

Research Article

Using CVIS to Improve Bus Schedule Adherence: A Predictive Control Strategy and Its Hardware-in-the-Loop Field Tests

Wei Yin, Jing Teng, Xiaoguang Yang, and H. Michael Zhang

Key Laboratory of Road and Traffic Engineering, Ministry of Education, Tongji University, 4800 Cao'an Road, Jiading, Shanghai 201804, China

Correspondence should be addressed to Jing Teng; tengjing@tongji.edu.cn

Received 22 August 2013; Accepted 22 September 2013

Academic Editor: Huimin Niu

Copyright © 2013 Wei Yin et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The ability of buses to adhere to their advertised schedule is vital to the bus operations. In this paper, an adaptive control strategy is proposed to dynamically adjust bus speed and traffic signal timings along the path of a running bus to improve its schedule adherence. The strategy relies on real-time location and speed information of buses provided by cooperative vehicle infrastructure system (CVIS) and uses key-time nodes calculated by back-stepping of planned arrival times to dynamically update signal timing plans to keep the bus running on time. A hardware-in-the-loop (HIL) field test was conducted to evaluate the developed strategy and the results are encouraging.

1. Introduction

Punctual service is vital to scheduled transit system operations but is often difficult to achieve in bus operations because its shared use of the road with other motor vehicles and the presence of traffic signals on its route. Traffic signals, bicycles and pedestrians, and uneven loading times at bus stops, to name a few, often cause buses to deviate from their advertised schedule.

Some work has been done in improving the reliability of bus arrival time by active signal control [1–3]. The primary method used in these studies is signal priority for buses [4, 5]. Bus signal priority, however, has two limitations. First, if there are many concurrent requests from different bus lines in each intersection, not all bus lines will get signal priority and their schedule cannot be guaranteed. Second, without prediction, the recognition of a bus being late may occur too late to have adequate time to restore the bus' schedule.

In order to proactively control traffic signals to support reliable bus schedules, it is desirable to predict bus arrival times in sufficient accuracy. Some complicated algorithms have been utilized to predict bus arrival times. The support vector machine (SVM) [6] and artificial neural network (ANN) [7] are two such algorithms. The data processing

mode of an ANN attempts to mimic that of a human brain, while that of the SVM is based on statistical learning theory. Both algorithms "learn" from examples provided in historical data to extract patterns for prediction. Such data were used to be obtained from inductive loops and/or other location based surveillance such as license plate readers.

With the development of vehicle-to-vehicle and vehicle-to-infrastructure technologies for improving traffic safety and traffic efficiency [8, 9], the implementation of cooperative vehicle infrastructure system (CVIS) would enable the location and velocity information of vehicles to be transmitted in real time to each other and to local traffic controllers [10, 11]. Coupled with more accurate traffic information provided by CVIS, the SVM or ANN algorithm can be used to predict bus arrival times more reliably.

In this paper, we develop a dynamic control strategy for the schedule adherence of buses under the CVIS environment and with the support of a reliable arrival time prediction algorithm such as SVC. Based on the real-time bus location and speed and the predicted arrival time under prevailing traffic conditions, it dynamically updates traffic signal timing plans at key intersections to keep the bus running on schedule. Since CVIS systems are not yet available, hardware-in-the-loop (HIL) test system is developed to evaluate the

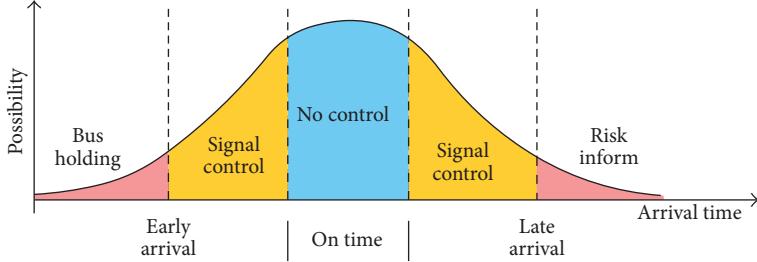


FIGURE 1: The conditions and corresponding actions.

effectiveness of the proposed control strategy. Together with field test and virtual simulator tests, HIL is often used to evaluate new ITS technologies [12–14].

The remainder of this paper is organized as follows. In Section 2, the basic concept and the logic of the control strategy are introduced. In Section 3, the HIL test platform is described and the test results are presented. Finally conclusions are given in Section 4.

2. The Basic Concept and Control Logic

2.1. The Basic Concept. Bus delays caused by the signal controls and random events can grow significantly as they travel along the path, akin to the Butterfly Effect shown in chaotic systems. For example, a small change in the system can develop into a catastrophic change over time. Similarly, a slight fluctuation in a bus' arrival time at an intersection may cause a long delay after it passes several intersections, which is one of the main causes of unpunctual bus service.

In Figure 1, a bus operating condition is divided into three categories with their possible frequencies of occurrences: early arrival, on time, and late arrival. In this figure, possible actions that help keep a bus' regular schedule are also shown. For the conditions that fall in the area marked by yellow, where bus unpunctuality is caused by random events like interference from bicycles/pedestrians and enlarged by the Butterfly Effect, using signal control alone may be sufficient to keep buses on schedule. It is in this range of conditions that our proposed bus schedule adherence control strategy applies.

The main idea of the proposed control strategy is recognizing the boundary at which the butterfly effect starts affecting the punctuality of buses and updating the signal timing plans in subsequent intersections to restore the bus' schedule. The strategy is also appropriate to the control of early bus [15, 16].

The planned arrival times are those times published in a bus schedule and the latest planned times are simply those of the planned time adding a tolerable amount of delay. Supposing that both of them have been known ahead of time, two back-stepping lines (green line and orange line) starting from the two arrival time points can then be drawn as shown in Figure 2 and the direction of the back-stepping lines is opposite to that of the bus trajectories. Those lines mark the boundaries of bus trajectories that would either adhere to schedule or be slightly late from the scheduled arrival times.

An intersection node is then defined as a key-time node if the back-stepping lines pass through its red phase. The control algorithm updates the signal timing plans of the key-time nodes to keep the bus trajectory within the bounds given by the two back-stepping lines. For example, a bus can take one of the three trajectories shown in Figure 2 (trajectories 1, 2, and 3) due to a small change of bus speed. The bus arrives at its stops either on time or within a tolerable amount of delay if it takes trajectories 1 and 2 but deviates further and further from its schedule if it takes trajectory 3. The proposed control would recognize the delay causing event at the second intersection and adjust, based on the computed back-stepping lines, the signal timing plans at subsequent intersections so as to steer the bus out of trajectory 3 and into trajectories 1 or 2.

CVIS plays an important role in the recognition of a late bus and the execution of corresponding measures because it establishes bidirectional communication between vehicles and signal controllers. Real-time bus location and speed are collected from vehicle and transmitted to control unit to predict the bus trajectory. It is then used to compare with the back-stepping lines to decide the corresponding measure. If it is speed adaption, the guide-related message is sent back to the vehicle from the control unit. And if it is signal adjustment, the message is sent to the signal controller.

The bidirectional communication based on CVIS makes the control fault-tolerant and precise. The actuated control only is unreliable because drivers' behavior is unpredictable without speed adaption. On the contrary, speed adaption only also lacks stable effect due to the interference from the surrounding vehicles, which can be solved by actuated control. The bidirectional communication integrates two types of measures and the estimation error of travel time or the fluctuation of driver behaviors is dynamically revised in the control.

2.2. The Operational Control Algorithm. Before we describe the control algorithm in detail, we first list the notations used in the rest of the paper in Table 1.

The process of executing the developed control strategy is shown in Figure 3. It consists of four steps: (1) no action, (2) speed adaption, (3) signal adjustment, and (4) delay notification. It takes no action unless the current speed of the bus may fail to keep the bus running within the stable region formed by the key nodes. If it is detected that the bus arrival time at a node falls outside the region, the speed adaption

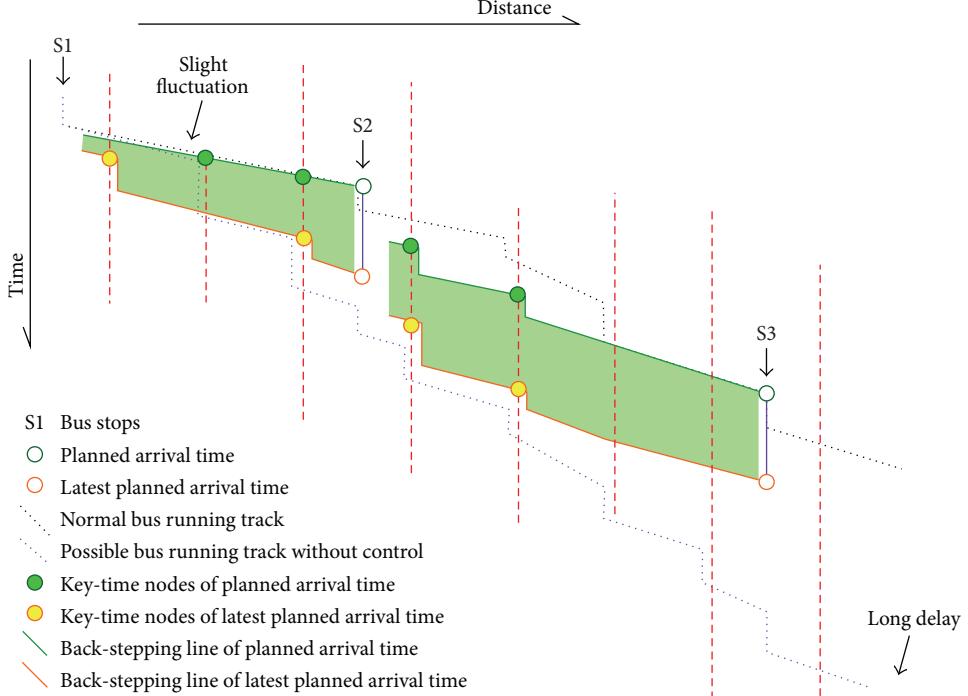


FIGURE 2: The basic concept of the proposed control.

TABLE 1: Notations.

Type	Symbol	Definition
Set	$i \in I$	The set of intersections
	$l \in L$	The set of links
	$bs \in BS$	The set of bus stops
	$tp \in TP$	The set of key-time node of planned arrival time
	$tl \in TL$	The set of key-time node of latest planned arrival time
Timing variable	K_i	Signal time when bus arrives at intersection i
	C	Common cycle
	t_i^a	Predicted arrival time point in one cycle when bus arrives at intersection i
	T_i^{te}	Red time between t_i^a and the ending time of the red phase at intersection i
	r_i^a	Back-stepping arrival time point in one cycle when bus arrives at intersection i
	T_i^{ts}	Red time between t_i^a and the starting time of the red phase at intersection i
	t_{bs}^{TP}	The planned arrival time point to bs downstream stop
	t_{bs}^{TL}	The latest planned arrival time point to bs downstream stop
Data	t_0	Current time
	V_c	Current speed of bus
	V_l	Empirical bus speed at link l
	L_l	Length of link l
	L_{bs}	Distance between bus and bs th downstream stop
	l_0	Current link
	l_{bs}^s	Link in which bs th downstream stop is located
	V_l	Empirical bus speed at link l

procedure is launched and the bus driver is notified. It is still possible that speed adaptation alone fails to bring the bus arrival times back within the stable region. Then the signal timing adjustment procedure kicks into the correct bus

trajectory. It checks the scope for improvement in the next several intersections downstream of the bus' current position, and if it is feasible to bring the bus trajectory inside the stable region, the control algorithm will update the signal timing

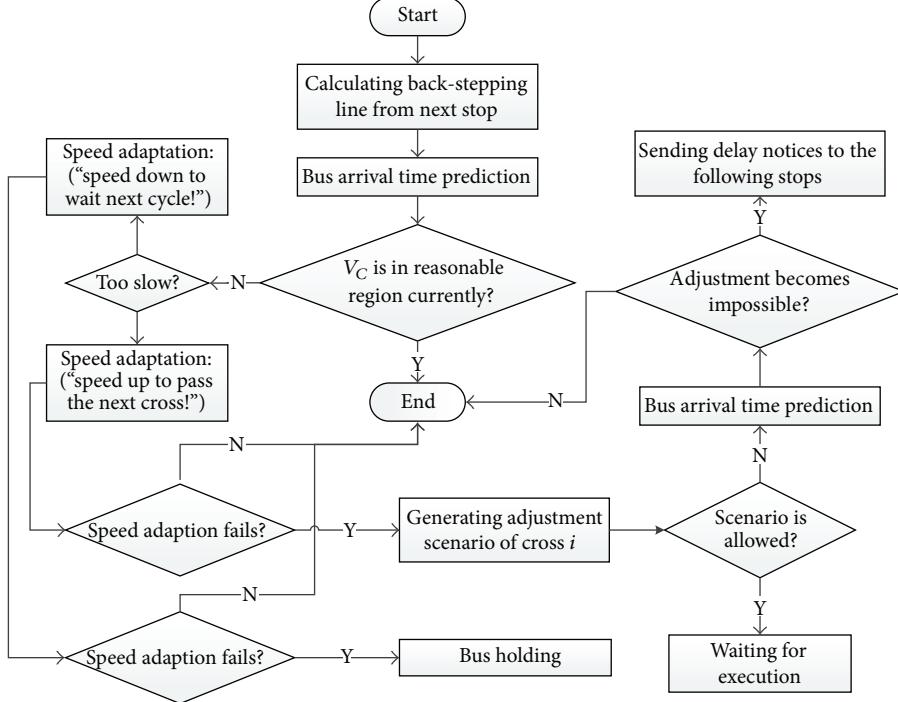


FIGURE 3: Flow chart of the control algorithm.

plans of the subsequent intersections accordingly, and if it is not possible to do so, a delay notice would be released to passengers waiting at the corresponding bus stops.

Next we describe the signal updating procedure in more detail. First, the back-stepping lines of planned arrival times are computed using the prevailing travel speeds. These are done in Steps 1 and 2. The key-time nodes are collected into the set $\{TP\}\{TL\}$. Next the predicted bus trajectory is obtained in Step 3 and checked against key-time nodes (Step 3-(3), (6)). If $t_i^a > tp$ or $t_i^a > tl$, the control algorithm starts generating signal adjustments (Step 4), which makes either a green extension in the green phase or a red truncation in the red phase. With signal timing adjustments made, the algorithm recalculates the predicted arrival times of the bus (Step 4-(3)). If the new bus arrival times fall within the green band in Figure 2, the signal adjustments are stored and wait to be executed. Otherwise, the control algorithm moves to Step 5 and estimates the delay risk of each downstream stop (Step 5-(2), (3)).

Step 1 (calculating the back-stepping line of planned arrival time). Consider the following:

- (1) $t_{\text{tmp}} \leftarrow t_0^{TP};$
 - (2) if $l = l_0$ then go to Step 2; else $r_i^a \leftarrow t_{\text{tmp}} - L_l/V_l; l \leftarrow l - 1;$
 - (3) if $r_i^a \in green\ phase$ then $t_{\text{tmp}} \leftarrow r_i^a; i \leftarrow i - 1;$ go to (2); else go to (4);
 - (4) $r_i^a \leftarrow r_i^a - T_i^{ts}; \{TP\} \leftarrow r_i^a; t_{\text{tmp}} \leftarrow r_i^a; i \leftarrow i - 1;$ go to (2).

Step 2 (calculating the back-stepping line of latest planned arrival time). Consider the following:

- (1) $t_{\text{tmp}} \leftarrow t_0^{TL}$;
 - (2) if $l = l_0$ then go to Step 3; else $r_i^a \leftarrow t_{\text{tmp}} - L_l/V_l$; $l \leftarrow l - 1$;
 - (3) if $r_i^a \in green\ phase$ then $t_{\text{tmp}} \leftarrow r_i^a$; $i \leftarrow i - 1$; go to (2); else go to (4);
 - (4) $r_i^a \leftarrow r_i^a - T_i^{ts}$; $\{TL\} \leftarrow r_i^a$; $t_{\text{tmp}} \leftarrow r_i^a$; $i \leftarrow i - 1$; go to (2).

Step 3 (bus arrival time prediction). Consider the following:

- (1) $t_{\text{tmp}} \leftarrow t_0$;
 - (2) if $l = l_0^s$ then go to Step 5; else go to (3) and $t_i^a \leftarrow t_{\text{tmp}} + L_l/V_l; l \leftarrow l + 1$;
 - (3) $tp \leftarrow \{TP\}$ or $tl \leftarrow \{TL\}$ //element with minimized i ;
 - (4) if $t_i^a \in green\ phase$ then $t_{\text{tmp}} \leftarrow t_i^a; i \leftarrow i + 1$; go to (6); else go to (5);
 - (5) $t_i^a \leftarrow t_i^a + T_i^{te}; t_{\text{tmp}} \leftarrow t_i^a; i \leftarrow i + 1$; go to (6);
 - (6) if $t_i^a < tp$ or $t_i^a < tl$ then go to End; else go to Step 4.

Step 4 (signal adjustment). Consider the following:

- (1) remove tp from $\{TP \text{ or } TL\}$;
 - (2) adjustment action at intersection i ;
 - (3) recalculate T_i^{te} and $i \leftarrow i + 1$; go to Step 3.

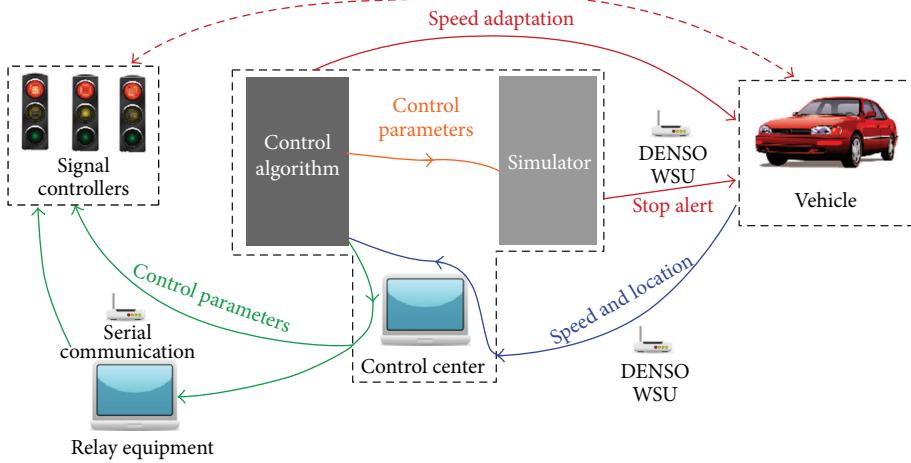


FIGURE 4: HIL simulator framework.

Step 5 (final processing). Consider the following:

- (1) if $t_i^a < tl$ then go to End; else go to (2);
- (2) declare delay at bs stop; go to (3);
- (3) $bs \leftarrow bs + 1$; if $t_{bs}^{TL} > t_i^a + \sum_K L_l/V_l K \in \{l_0, l_{bs}^s\}$ go to End; else go to (2).

3. Evaluation Using HIL Tests

3.1. The HIL Platform. A hardware-in-the-loop simulator is used to evaluate the proposed adaptive bus schedule adherence control strategy. As shown in Figure 4, the HIL simulator consists of two field hardware components (a specific vehicle (bus) and field signal controllers) and one simulation component (a control center which consists of control algorithms to generate signal timing plan updates and a simulator to generate and move virtual vehicles). Between the three components there are four sets of communication links labeled with different colors: the control center receives the real-time vehicle status via the blue link; the center controls the traffic lights in the field through the green links and releases the speed adaptation command to the field vehicle through the red link; signal timing plan updates are transmitted to the simulator via the yellow link; an alert message from the control center is sent to the bus driver via a red link if the bus has a high risk of “crash” with virtual vehicles. Among these communication links, the simulator-to-vehicle links are realized by DENSO wireless safety unit (WSU) and the simulator-to-controller links are realized by wireless serial communication.

The HIL simulator interface is shown in Figure 5. The simulator uses a mesoscopic model in which the red dot represents the real vehicle and the black dots represent the virtual ones. Except for the virtual vehicles, all the objects in this system are real.

3.2. Validation. A loop in the new Tongji University campus shown in Figure 6 was selected as the test site. The loop consists of four road segments with lengths of 150 m, 250 m,

TABLE 2: Fixed signal times of the four signals in the test.

	Green phase (s)	Red phase (s)	Yellow phase (s)
Signal1	20	20	3
Signal2	25	20	3
Signal3	30	25	3
Signal4	15	15	3

145 m, and 230 m, respectively. There are four two-phase signalized intersections in the loop, whose timing plans are shown in Table 2. Two real bus stop locations were set on the longer road segments, which represent the six virtual stops. As is shown in Figure 5, the same loop is created in the simulation with virtual vehicles traveling across the intersections and on the loop. In the test, a car is used to mimic a bus and the speed characteristic scales down due to the scale differences between the loop and the real roads.

For the simulation of random interference on the road, a small application running in the test vehicle would make a sound regarding a random value (negative is allowed) at a random moment to instruct the driver to speed up or slow down. To simulate possible conflicts generated from traffic in the opposite direction and from the cross streets, the system has an option to reject signal priority requests from the test vehicle (bus).

Two sets of experiments were carried out, one without adaptive signal control and one with adaptive signal control. In each experiment, the bus vehicle runs through the loop three times and hence makes six stops. The experiment is repeated 20 times for each test set. Seven typical bus trajectories without control (purple lines) and two trajectories with control (red and pink lines) are shown in Figure 7. In the plot showing trajectories with control, the trajectories without control were also shown in the background, and the planned arrival time (marked as blue points and lines), back-stepping lines (green lines and orange lines), and corresponding actions are labeled as well.

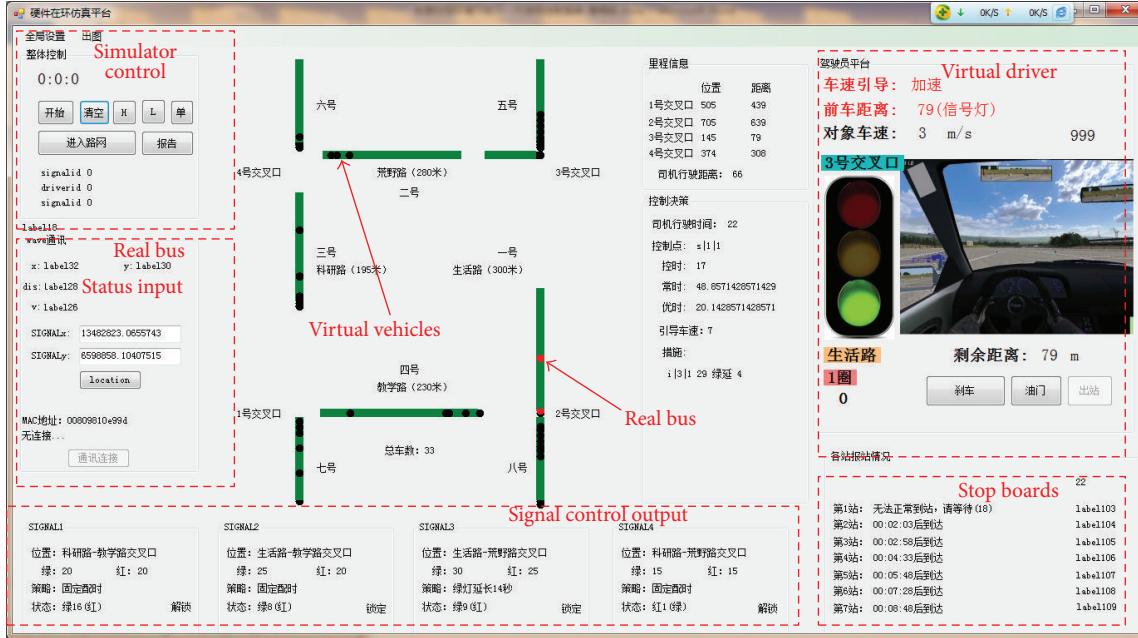


FIGURE 5: HIL simulator interface.

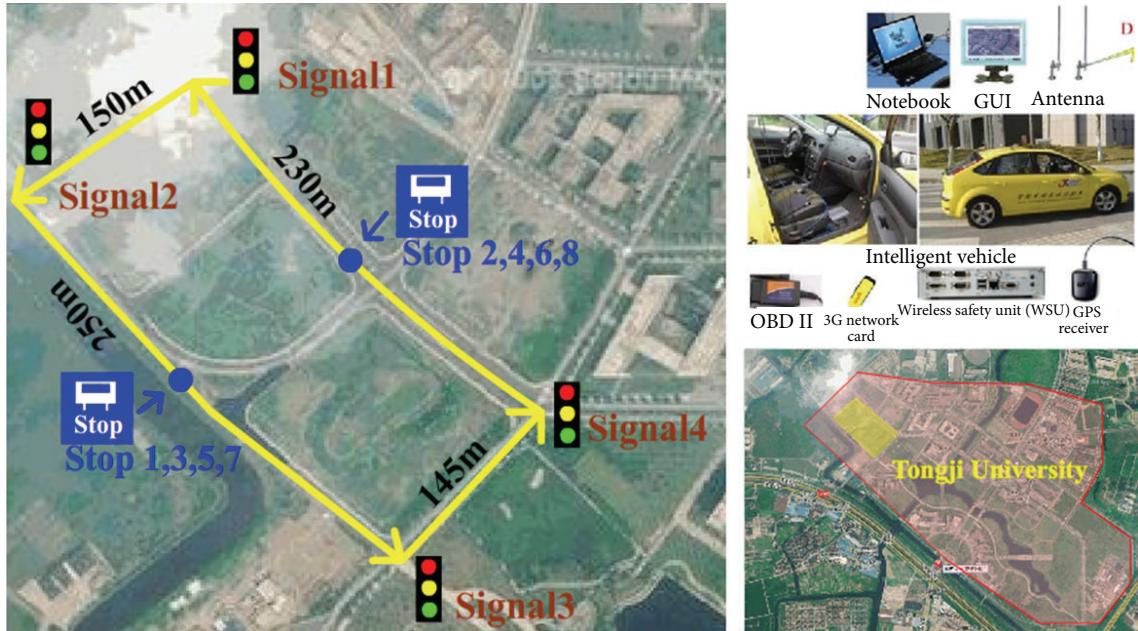


FIGURE 6: Field equipment and the test site.

To estimate the performance of the control strategy, the punctuality index PI(s) is created as follows:

$$PI = \frac{\sum_{i=0}^N | \text{planned arrival time } (i) - \text{actual arrival time } (i) |}{N} \quad (1)$$

where N is the number of samples in each bus stop. The PI results of the tests are shown in Figure 8.

As shown in Figure 7, when there is no control, the bus schedule drifts away from the planned schedule further and further as the bus travels along the route, which is reflected in the PI: it rises from 5 s to 47 s in the subsequent stops (the bus made six stops in each run). The change of PI (Group 1) has close relationship with the distance between the two physical stops, the number of intersections, and the offset between signals.

The two sample trajectories presented in Group 2 in Figure 7 demonstrate the robustness of the control algorithm: the

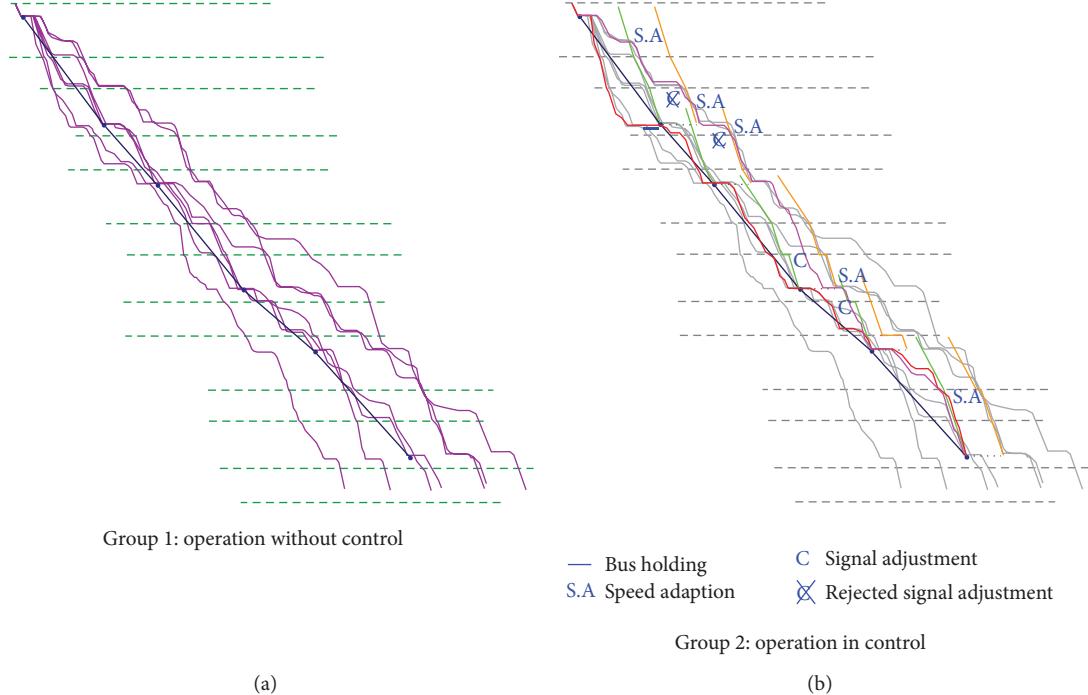


FIGURE 7: Experiment results.

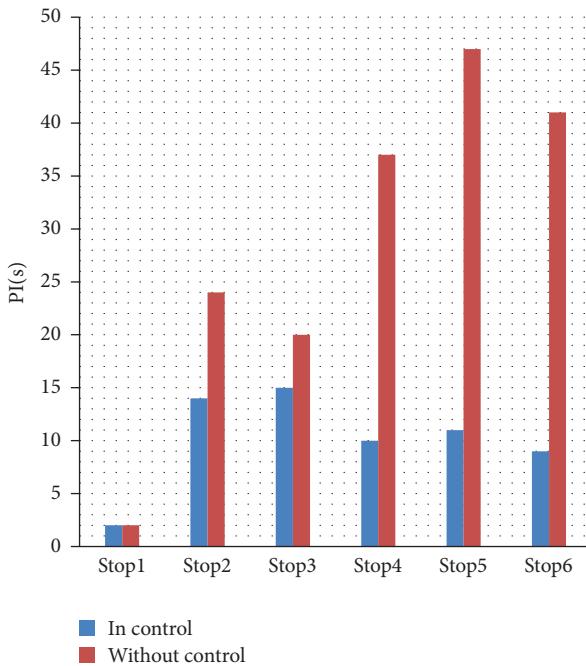


FIGURE 8: Punctuality index PI(s) in each stop.

bus received five speed adaptation messages but one of them failed to be implemented and the traffic signals were requested to change their timing plans four times but were only able to accommodate two such requests, yet the bus (pink lines) was able to run within 5 s to 15 s of the planned schedule despite these failings.

4. Conclusions

The proposed HIL simulation platform could reflect the performance of CVIS and the control strategy because all communication links and most objects are real. And given that there is no real crash between vehicles, the entire test is relatively more secure than the test on the real traffic system.

The HIL simulation results of the study have shown that the proposed strategy is feasible. The punctuality of the bus is effectively raised with the implement of the control strategy.

The strategy is fault-tolerant and precise due to the implement of the bidirectional communication based on CVIS. The reasonable estimation error of travel time is allowed and some failures of speed adaptation or signal adjustment have few fatal influences on the whole.

It should be noted that in the current work our primary objective is to keep buses running on schedule, and the control algorithm does not consider the potential delays to cross street traffic and traffic traveling opposite to the bus in detail. We will evaluate this potential adverse effect in our future work and refine our control strategy if such adverse effect is serious.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This research is supported by the National Natural Science Foundation of China (Project no. 61174185). The authors

wish to thank Dr. Wanjing Ma from Tongji University for his helpful suggestions and useful comments on an earlier version of the paper.

References

- [1] P. B. Mirchandani and D. E. Lucas, "Integrated transit priority and rail/emergency preemption in real-time traffic adaptive signal control," *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*, vol. 8, no. 2, pp. 101–115, 2004.
- [2] N. B. Hounsell, B. P. Shrestha, F. N. McLeod, S. Palmer, T. Bowen, and J. R. Head, "Using global positioning system for bus priority in London: traffic signals close to bus stops," *IET Intelligent Transport Systems*, vol. 1, no. 2, pp. 131–137, 2007.
- [3] Y. Wadjas and P. G. Furth, "Transit signal priority along arterials using advanced detection," *Transportation Research Record*, no. 1856, pp. 220–230, 2003.
- [4] F. Dion, H. Rakha, and Y. Zhang, "Evaluation of potential transit signal priority benefits along a fixed-time signalized arterial," *Journal of Transportation Engineering*, vol. 130, no. 3, pp. 294–303, 2004.
- [5] H. R. Smith and B. Hemily, *Signal Priority (TSP): A Planning and Implementation Handbook*, 2005.
- [6] B. Yu, W. H. K. Lam, and M. L. Tam, "Bus arrival time prediction at bus stop with multiple routes," *Transportation Research Part C*, vol. 19, no. 6, pp. 1157–1170, 2011.
- [7] R. Jeong and L. R. Rilett, "Bus arrival time prediction using artificial neural network model," in *Proceedings of the 7th International IEEE Conference on Intelligent Transportation Systems (ITSC '04)*, pp. 988–993, October 2004.
- [8] H. Hartenstein and K. P. Laberteaux, "A tutorial survey on vehicular ad hoc networks," *IEEE Communications Magazine*, vol. 46, no. 6, pp. 164–171, 2008.
- [9] J. Anda, J. LeBrun, D. Ghosal, C. N. Chuah, and M. Zhang, "VGrid: vehicular adhoc networking and computing grid for intelligent traffic control," in *Proceedings of the 2005 IEEE 61st Vehicular Technology Conference (VTC '05)*, pp. 2905–2909, June 2005.
- [10] S. Hallé and B. Chaib-draa, "A collaborative driving system based on multiagent modelling and simulations," *Transportation Research Part C*, vol. 13, no. 4, pp. 320–345, 2005.
- [11] S. K. C. Lo, "A collaborative multi-agent message transmission mechanism in intelligent transportation system—a smart free-way example," *Information Sciences*, vol. 184, no. 1, pp. 246–265, 2012.
- [12] S. Röglinger, "A methodology for testing intersection related Vehicle-2-X applications," *Computer Networks*, vol. 55, no. 14, pp. 3154–3168, 2011.
- [13] O. Gietelink, J. Ploeg, B. de Schutter, and M. Verhaegen, "Development of advanced driver assistance systems with vehicle hardware-in-the-loop simulations," *Vehicle System Dynamics*, vol. 44, no. 7, pp. 569–590, 2006.
- [14] J. Alonso, V. Milanés, J. Pérez, E. Onieva, C. González, and T. de Pedro, "Autonomous vehicle control systems for safe crossroads," *Transportation Research Part C*, vol. 19, no. 6, pp. 1095–1110, 2011.
- [15] W. Ma, X. Yang, and Y. Liu, "Development and evaluation of a coordinated and conditional bus priority approach," *Transportation Research Record*, no. 2145, pp. 49–58, 2010.
- [16] A. Skabardonis, "Control strategies for transit priority," *Transportation Research Record*, no. 1727, pp. 20–26, 2000.

