Article:
Zhao, W, Liu, R and Ngoduy, D (2018) A bilevel programming model for autonomous intersection control and trajectory planning. Transportmetrica A: Transport Science. ISSN: 2324-9943

https://doi.org/10.1080/23249935.2018.1563921

A bilevel programming model for autonomous intersection control and trajectory planning

Weiming Zhao[a,*], Ronghui Liu [a] and Dong Ngoduy [b]

[a] *Insitute for Transport Studies, University of Leeds, UK*;
[b] *Civil & Natural Resources Engineering, University of Canterbury, New Zealand*

ARTICLE HISTORY
Compiled January 4, 2019

ABSTRACT
Advances in autonomous and connected vehicles bring new opportunities for intelligent intersection control strategies. In this paper, we propose a centralised way to jointly integrate an intersection control problem with vehicle trajectory planning. It is formulated as a bilevel optimisation problem in which the upper level is designed to minimise the total travel time by a mixed integer linear programming (MILP) model. In contrast, the lower level is a linear programming (LP) model with an objective function to maximise the total speed entering the intersection. The two levels are coupled by the arrival time and terminal speed. By using the relationship between the safe time headway and the process time, a novel platoon based method is developed to reduce the computational burden. Finally, simulation tests are carried out to investigate the control performance under different demands, intersection lengths, communication ranges and traffic compositions.

KEYWORDS
Autonomous vehicle; intersection control; trajectory planning; bilevel programming

1. Introduction

Traffic control is one of the most important methods to organise vehicle movements in urban networks. The purposes of traffic control may vary according to the traffic demand and vehicle composition. Nevertheless, most of them still share a common objective to create safe and efficient traffic operations at urban intersections. Various traffic control methods are proposed, for example for a local intersection or a network, using a fixed, actuated or adapted control strategy. Basically, the more advanced the traffic control method is, the more precise the traffic information it requires. This usually indicates that new traffic detectors, such as loop detector, Bluetooth, Electronic Vehicle Identification, need to be installed.

It is a common vision that many vehicles will be equipped with some kinds of Vehicle Autonomous and Connected System (VACS) in the near future (Diakaki et al. 2015). The autonomous vehicle is considered not just a sensor but also an actuator. From the aspect of sensor, it is possible to access much more detailed data about the vehicle and the traffic flow than traditional traffic detectors. For example, such

*Corresponding author. Email: w.zhao14@leeds.ac.uk, joyfig07@gmail.com

data include the acceleration limit of each vehicle and the driver's desired speed and value of time (Isukapati and Smith 2017). From the aspect of actuator, it is possible to control some or all vehicles' trajectories along the link for a pre-designed purpose such as eco-driving (Ma et al. 2017; Zhao et al. 2018). These distinct properties of the autonomous vehicle help to create a vehicle-based traffic control method rather than the traditional flow-based method.

For an urban traffic control problem, the autonomous vehicle can receive and send system dynamic information to the intersection controller or road-side infrastructure and react precisely according to the control strategy. This provides the intersection controller with more capabilities to optimise the traffic flow around the intersection, which is the so-called autonomous intersection control strategy. In the autonomous intersection control, the signal timing problem can be seen as a machine scheduling problem. The conflict area in the intersection acts as a "server" or "machine" while every approaching vehicle around the intersection is a "job". The vehicles can negotiate either with each other or with the intersection controller to allocate the priority of passing sequence or the time window to avoid collisions and to achieve more efficient operations at the intersection. In this control method, collision avoidance can be obtained by either a pure signal timing or a pure trajectory planning. However, the efficiency is an issue that is much more difficult to achieve at the same time. To this end, our main motivation in this paper is to consider simultaneously signal timing and trajectory planning in the autonomous intersection control method.

In the literature, little research considered the vehicle dynamics in the centralised signal control. In the scheduling method, it is usually assumed that the time required for a vehicle to pass the intersection (i.e. process time) is a constant. However, in reality, the process time strongly depends on the speed crossing the stop line and the travel time from the current position to the stop line. In this paper, a novel bilevel programming method is proposed to optimise the vehicle passing sequence and the vehicle trajectory. In our bilevel optimisation model, the upper level determines the passing sequence which is seen as a job shop scheduling problem and modelled as a mixed integer linear programming (MILP), whereas the lower level determines the corresponding process time which is modelled as a linear programming (LP). Though the model introduces many variables, due to the linear structure, it can be solved very efficiently using existing commercial solvers such as CPLEX (IBM 2017) and Gurobi (Gurobi Optimization 2017).

In summary, the main contributions of this paper are:

(1) To combine the intersection control optimisation and trajectory optimisation which makes sure that the result of intersection control optimisation is achievable and optimal.
(2) To consider explicitly the vehicle's dynamics in the model in which there is no need to assume the vehicle's speed entering the intersection.
(3) To achieve a quick convergence in the optimisation problem via a linear programming structure in each level of the proposed bilevel programming model.
(4) To propose a new platoon-based method to dynamically reduce the computation burden and increase the efficiency of the intersection control problem.

We briefly review the state-of-the-art in intersection control problems with autonomous vehicles in Section 2. Section 3 describes the notations used throughout this paper and the structure of the proposed model. Furthermore, the formulation of the model and a novel platoon split approach is proposed. Section 4 illustrates several simulation tests in various scenarios. Finally, we conclude the paper in Section 5.

2. Literature review

The traffic control method can be classified as the centralised method (Diakaki, Papageorgiou, and Aboudolas 2002) and the distributed method (Bazzan 2005). Diakaki, Papageorgiou, and Aboudolas (2002) developed a traffic-responsive network-wide signal control using store-and-forward modelling and linear-quadratic regulator theory. Bazzan (2005) developed a decentralised coordinated traffic control in an arterial using evolutionary game theory. Every intersection is modelled as an intelligent agent and considers both local goal and global gain. In the ensuing paper, we will discuss the three main approaches in autonomous intersection control: 1) centralised control method; 2) vehicle trajectory-based signal control; and 3) combined intersection control and vehicle trajectory optimisation.

The first main approach is the centralised control method which usually needs to collect every vehicle's information within a certain range before the optimisation is executed. Xie et al. (2012) developed a scheduling-based intersection control method in which the vehicles are first aggregated into clusters and then the intersection control problem is formulated as a scheduling model. A forward recursive algorithm was proposed to solve this intersection control problem efficiently with some state elimination criterion. Sun, Zheng, and Liu (2017) proposed an intersection operation method to maximise the intersection capacity. This method considered all the through and turning movements, but the lane management is not fixed and can be dynamically changed. The green duration and lane management are optimised together by a multi-objective mixed-integer nonlinear programming model. Zhu and Ukkusuri (2015) proposed a linear programming model to optimise traffic flows in the network accounting for the dynamic departure time and dynamic route choice. Li and Zhou (2017) proposed a novel phase-time-traffic hyper-network model to represent the heterogeneous traffic of autonomous vehicles and traditional vehicles and used a mixed integer programming model to minimise the total delay at an intersection. Guler, Menendez, and Meier (2014) added a penalty term to the departure time calculation and took into account the different passing time for connected vehicles because of the different passing sequences. However, this model still did not consider the passing velocity. In a similar line, some other research also considered the intersection control with fully connected vehicles (Goodall, Smith, and Park 2013) or partially connected vehicles (Feng et al. 2015).

Dresner and Stone (2004, 2005, 2006, 2008) studied extensively a reservation-based autonomous intersection management. In this approach, every vehicle is an autonomous agent and there is another intersection manager agent at each intersection. The vehicle agent sends a request to reserve a specific space and time in the intersection, and the intersection agent will accept or reject the reservation based on the intersection control policy. Therefore, in this method, the control policy is a black box to the vehicle agent, and the vehicle agent cannot know when and what kind of reservation will be accepted. The most widely used policy in the reservation-based model is the first come first serve (FCFS) policy. Different levels of priority in the reservation method were considered by Zhang et al. (2015). The main drawback of the reservation-based rule is that it only accepts or rejects the request of the vehicle, but does not provide the exact time of passing. So it cannot be employed to optimise the vehicle trajectory. Levin, Boyles, and Patel (2016) showed that such reservation-based intersection control methods may increase congestion or vehicle delay in some situations.

The second approach is to coordinate the trajectories of vehicles around the intersection to avoid conflicts. Li and Wang (2006) developed a cooperative driving model at blind crossings. The grouped vehicles used a decision tree generation model to get a safe and efficient driving pattern. Then a virtual vehicle mapping technique was used to get the safe vehicle trajectory. Lee and Park (2012) developed an optimisation model to minimise the overlapping trajectory of vehicles from conflicting roads, thus to ensure a safe manoeuvre for every vehicle. The main drawback of this method is that the complex nonlinear constrained model makes it difficult to find a feasible solution, and then it will fallback to a rule-based control method. Even though the method can reduce the average delay, the main purpose of the model is to generate a safe trajectory, while efficiency is not considered. Kamal et al. (2015) proposed a risk function to quantify the risk of collision around the intersection and then used a model predictive control method to avoid collision by considering all the vehicles' states. Yang and Monterola (2016) developed a decentralised intersection traffic control with no traffic lights where some vehicles are equipped with a simple driver assistance system. The method only controls vehicles to brake in a specific condition, so it is suitable for level 1 or above autonomous vehicles. This simple method can be adopted to mixed traffic with both conventional and autonomous vehicles.

Besides the two approaches, some work has been conducted to combine the intersection control and trajectory planning. Li, Elefteriadou, and Ranka (2014) discussed the vehicle's trajectory under different travel times, however they assumed that all vehicles except the first vehicle on the road can achieve the maximum velocity when they enter the intersection and the model is strongly nonlinear. An enumeration method is used to obtain best signal timing instead of an optimisation model. Müller, Carlson, and Junior (2016); Yang, Guler, and Menendez (2016); Xu et al. (2017) and Yu et al. (2018) formulated the problem as a two-stages problem: solve intersection optimisation first, and then optimise the trajectory. Müller, Carlson, and Junior (2016) and Yu et al. (2018) assumed that every vehicle can achieve the maximum speed or desire speed when entering the intersection which may not be possible for the vehicle close to the intersection, or in high demand situation. Yang, Guler, and Menendez (2016) assumed a piece-wise linear trajectory in the trajectory optimisation model which may not be realistic, but unconnected vehicles are also considered in their model. Xu et al. (2017) and Yu et al. (2018) combined traffic signal optimisation with phases and an optimal control model for vehicle trajectory optimisation. However, they still use the conventional duel ring signal structure for the case of autonomous vehicles and there is no feedback between signal optimisation and vehicle trajectory optimisation. Besides, Yu et al. (2018) also optimised the lane changing behaviour for autonomous vehicles.

3. Methodology

3.1. *Problem formation*

To simplify the problem, we only consider a simple intersection with two approaches as shown in Figure 1 in which only one conflict zone exists (red shadowed area). This simple intersection is widely used in the development of autonomous intersection control, such as in Li, Elefteriadou, and Ranka (2014); Guler, Menendez, and Meier (2014); Yang, Guler, and Menendez (2016) and Yang and Monterola (2016). It is a good start of developing complex vehicle-based intersection control methods. All the vehicles in the system are autonomous vehicles that are able to communicate with the intersection controller via Vehicle-to-Infrastructure (V2I) technology. No communication delay or package loss is considered in this paper. The intersection control problem is formulated in a centralised way, which is described as follows:

(1) When a vehicle enters a certain control range, it will send some necessary information to the intersection controller in order to request a proper time to pass the intersection. The information includes its current position, speed and other characteristics such as the desired speed and acceleration limit.
(2) The intersection controller collects all the information from vehicles, then attempts to optimise both intersection performance and vehicle trajectories.
(3) The vehicles receive the arrival time and trajectories from the intersection controller and behave accordingly.

Another assumption is that the conflict zone can only be used by the vehicles coming from the same direction simultaneously. This means that if the intersection is already occupied by a vehicle from one approach, the vehicles coming from another approach cannot enter the intersection anymore no matter how large the intersection is. Consequently, they have to decelerate towards the stop line. However, it is possible that multiple vehicles coming from the same approach appear in the conflict area simultaneously if they can keep a safe time headway.

All the settings and assumptions in this problem are ideal in our proof of concept study. Nevertheless, it is still beneficial to see how autonomous vehicles can improve the performance of an urban traffic control strategy.

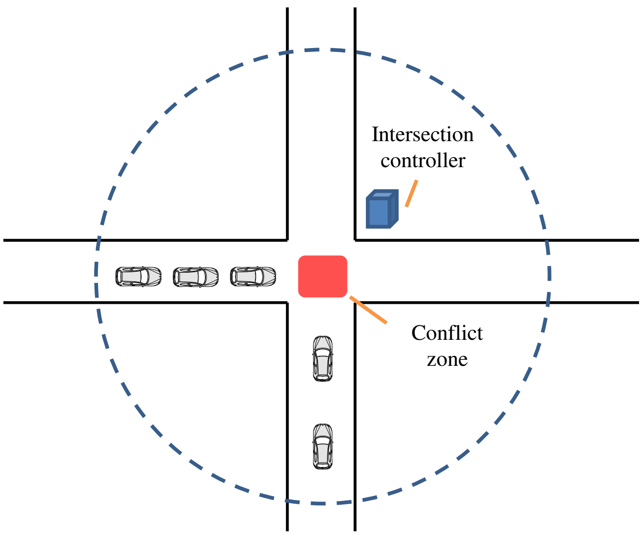


Figure 1.: A simple intersection with two approaches

3.2. *Notations*

All the major symbols used throughout this paper are described in Table 1.

Table 1.: Notations

i	Index for roads, in this paper $i = 1, 2$
j	Index for vehicles
f_u, f_l	The objective function of the upper level (lower level) optimisation
n_i	Number of vehicles within the communication range on road i
$T_{i,j}$	The arrival time to the stop line for vehicle j on road i
$T_{i,j}^{min}, T_{i,j}^{max}$	The minimum (maximum) arrival time to the stop line for vehicle j on road i
$v_{i,j}^{k}$	Speed for vehicle j on road i at time step k
$v_{i,j}^{r}$	Terminal speed to enter the intersection for vehicle j on road i
$v_{i,j}^{0}$	The initial speed at the beginning of optimisation for vehicle j on road i
$s_{i,j}^{k}, s_{i,j}^{0}$	The distance to the stop line for vehicle j on road i at time step k (at the initial time step)
$l_{i,j}$	Vehicle length of vehicle j on road i
h	Saturation headway
$p_{i,j}$	Process time (the travel time from the stop line to exit the intersection)
$c_{j,j'}$	Interchange time (the time difference of vehicle j entering the intersection before vehicle j' from a conflict road)
θ	Redundant time for safety in interchange time
$\hat{h}_{i,j}$	Time headway for vehicle j on road i entering the intersection
$o_{j,j'}$	Binary variable to denote the passing order for conflicting vehicle j and j'
$S_{i,j}^{k}$	Safe distance headway for vehicles j on road i with the preceding vehicle
s_0	Minimum safe distance (including vehicle length) at jam density
L_s	Intersection length
L_c	Communication range
a_{max}, a_{min}	Maximum acceleration (deceleration)
v_{max}, v_{min}	Maximum (Minimum) speed
$x_{i,j}$	The distance from the entrance of the road for vehicle j on road i

3.3. *Model structure*

Before formulating the model, a major issue needs to be addressed, why the intersection control method should consider the trajectories of autonomous vehicles and what benefits it can bring to the system?

(1) Vehicles' arrival information is important for intersection control from both local and global points of view. If there are no traffic detectors on the roads, the

intersection can only be operated in a fixed timing strategy. If there are loop detectors located before the stop line, the arrival time of vehicles can be estimated and the intersection will be operated in an actuated or even adaptive timing strategy. If the vehicle position and speed information are available in real time, a better green phase can be allocated and the wasted green time can be reduced.

(2) Usually, in a conventional intersection signal control, there should be some yellow time or even all-red time between conflicting green phases to clear the remaining vehicles in the intersection. For a large intersection, the yellow time and all-red time are usually lengthy to enable the slowest vehicle to pass the intersection safely. For example, a stationary vehicle needs 3.87 s to cross the intersection with a constant acceleration of $2\,\mathrm{m/s^2}$ when the length of intersection is 10 m and the vehicle length is 5 m. But a vehicle running with 55 km/h only needs 0.98 s. This does not even count the start-up loss time for human-driven vehicles. In the environment of autonomous vehicles, every vehicle's clearance time and the intersection clearance time can be obtained accurately. This will greatly reduce the inter-green time. Unfortunately, this issue has been ignored in the previous research.

(3) The other reason for including yellow time in the conventional signal timing is to reduce rear-end crashes in the dilemma zone. In most cases, the yellow time is set to 4.2 s on a 72 km/h road (Rakha, Amer, and El-Shawarby 2008). The time is even longer for higher-speed roads. With V2X communication capability, the intersection controller can detect whether there are vehicles in the dilemma zone and give priority to that road. So there is no need to set the yellow time anymore. Actually, points (2) and (3) are also the main reasons why the phase in conventional signal timing cannot be switched frequently.

The connection between intersection control and trajectory planning is the vehicle state at the stop line including arrival time $T_{i,j}$ and terminal speed $v^r_{i,j}$. As there is no need to stop at the stop line for an autonomous vehicle, the arrival time is equal to the time entering the intersection. The terminal speed determines the time required to pass the intersection which is called *process time* in the ensuing paper. Then the autonomous intersection control problem is modelled as a bilevel programming problem. Its main structure is shown in Figure 2 and the model can be seen as a master-slave scheme (Lamorgese, Mannino, and Piacentini 2016; Sharon et al. 2015). The master problem is the intersection control which minimises the total travel time for all vehicles. It is then formulated as a mixed integer linear programming (MILP) problem in this paper. The slave problem is the trajectory optimisation for every vehicle which maximises its speed to enter the intersection. This problem is formulated as a linear programming (LP) problem in this paper. The whole bilevel programming problem is decoupled by a well-known Benders' reformulation (Nemhauser and Wolsey 1999). Please note that in this framework the lower level optimisation is not just following the upper level intersection control. In addition, the optimised terminal speed $v^r_{i,j}$ which is the output of the lower level optimisation will be the input for the upper level optimisation in the next iteration. To this end, the feedback structure and interaction between the upper level and lower level can describe the relationship between the intersection control and the trajectory planning more accurately.

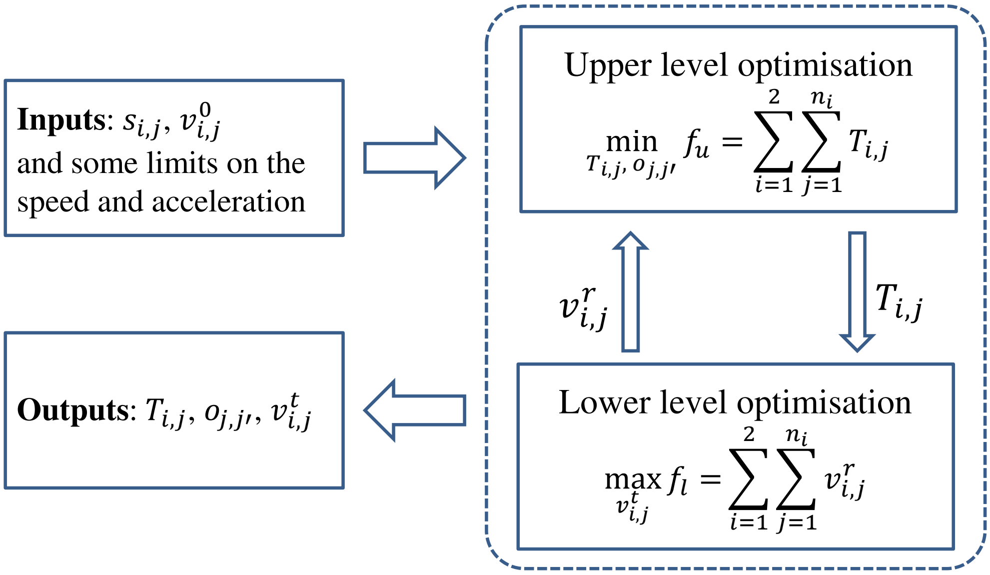


Figure 2.: Schematic of the bilevel programming structure

3.4. *Upper level optimisation*

The objective function for the upper level optimisation is minimising the total travel time for all the vehicles in the communication range around the intersection. As the minimum travel time can be pre-calculated, minimising the total travel time is equivalent to minimising the average delay. That is:

$$\min_{T_{i,j}, o_{j,j'}} f_u = \sum_{i=1}^{2}\sum_{j=1}^{n_i} T_{i,j} \tag{1}$$

where every vehicle's arrival time is bounded by the earliest arrival time $T_{i,j}^{min}$ and the latest arrival time $T_{i,j}^{max}$, i.e.

$$T_{i,j}^{min} \leq T_{i,j} \leq T_{i,j}^{max} \tag{2}$$

To achieve the earliest arrival time, the vehicle has to accelerate to the maximum velocity as early as possible. In order to be consistent with the lower level optimisation, the acceleration is also updated every 0.5 s, which means that the velocity change only happens at the beginning of an interval. Although the obtained minimum travel time under such 0.5s-update-interval can be higher than the theoretical minimum travel time without such assumption, usually the difference is less than 0.01 s and can be ignored. The earliest time to arrive at the stop line is fixed and can be calculated as follows:

(1) If the vehicle cannot achieve the maximum speed because of the distance restriction, i.e.

$$s_{i,j}^0 \leq \frac{(v_{max})^2 - (v_{i,j}^0)^2}{2a_{max}} \tag{3}$$

where $s_{i,j}^0$ denotes the current distance to the stop line, then $T_{i,j}^{min}$ can be calculated as

$$T_{i,j}^{min} = \frac{\sqrt{(v_{i,j}^0)^2 + 2a_{max}s_{i,j}^0} - v_{i,j}^0}{a_{max}} \tag{4}$$

(2) If it can achieve the maximum speed, the minimum travel time includes three parts as shown in Figure 3, where $T_{i,j}^{min}$ can be computed as:

$$t_1 = \left\lfloor \frac{2(v_{max} - v_{i,j}^0)}{a_{max}} \right\rfloor \tag{5}$$

$$s_1 = v_{i,j}^0 t_1 + \frac{1}{2}a_{max}t_1^2 \tag{6}$$

$$s_2 = \frac{1}{2} \times 0.5 \times \left(v_{i,j}^0 + t_1 a_{max} + v_{max}\right) \tag{7}$$

$$t_3 = \frac{s_{i,j}^0 - s_1 - s_2}{v_{max}} \tag{8}$$

$$T_{i,j}^{min} = t_1 + 0.5 + t_3 \tag{9}$$

where $\lfloor x \rfloor$ means the largest integer less than or equal to x.

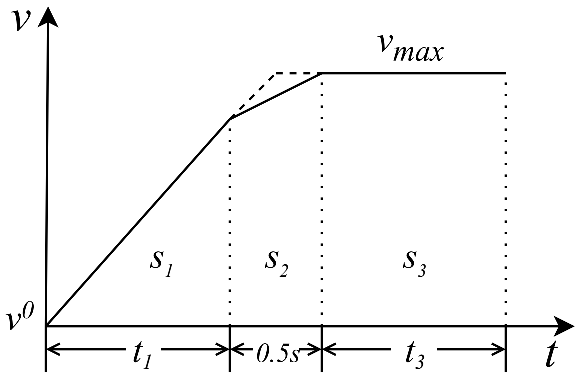


Figure 3.: Calculation of minimum travel time $T_{i,j}^{min}$

To calculate the maximum arrival time, there are two situations:

(1) If the vehicle cannot stop before the stop line, then the maximum arrival time is calculated by applying the maximum deceleration.

$$T_{i,j}^{max} = \frac{\sqrt{(v_{i,j}^0)^2 + 2a_{min}s_{i,j}^0} - v_{i,j}^0}{a_{min}} \tag{10}$$

(2) If the vehicle can stop before the stop line, then the maximum arrival time is calculated by setting a maximum delay t_{delay}. The maximum allowed delay can be vehicle specific. For simplicity, we use a constant for every vehicle.

$$T_{i,j}^{max} = T_{i,j}^{min} + t_{delay} \tag{11}$$

For the vehicles running on the same road, the safe time headway should be maintained not only at the entering time but also at the exiting time due to the difference in the process time. Constraint (12) below indicates the safety constraint at the entering time while constraint (13) at the exiting time.

$$T_{i,j} + h \leq T_{i,j+1} \quad i = 1,2; j = 1,2,\ldots,n_i - 1 \tag{12}$$

$$T_{i,j} + p_{i,j} + h \leq T_{i,j+1} + p_{i,j+1} \tag{13}$$

where the safety headway h is set to 1.5 s. Various safety headways for autonomous vehicles are used in the literature around $1 \sim 2$ s (Yang, Guler, and Menendez 2016; Ghiasi et al. 2017; Jia and Ngoduy 2016b).

In many studies, the process time for every vehicle is usually set to be a constant value by assuming that the vehicle passes the intersection with the maximum velocity or desired velocity (Müller, Carlson, and Junior 2016; Yu et al. 2018), but this may not be possible if the vehicle is too close to the intersection or the traffic volume is high. So we define the process time as a function of the entering velocity. It can be easily calculated by replacing $v^0_{i,j}$ to $v^r_{i,j}$ and $s^0_{i,j}$ to $L_s + l_{i,j}$ in equations (3) - (9).

For the vehicles running on different roads, the safety constraints are much more difficult to obtain due to the unknown passing sequence. We introduce a binary variable $o_{j,j'} \in \{0,1\}$ to indicate the passing sequence: $o_{j,j'} = 1$ if vehicle j in approach 1 crosses the conflict area before vehicle j' in approach 2, otherwise $o_{j,j'} = 0$. Then the safety constrains are

$$T_{i,j} + c_{j,j'} \leq T_{i',j'} + M(1 - o_{j,j'}) \tag{14a}$$
$$T_{i',j'} + c_{j',j} \leq T_{i,j} + M o_{j,j'} \tag{14b}$$

where M is a large enough constant.

The interchange time is calculated by

$$c_{j,j'} = p_{i,j} + \theta \tag{15a}$$
$$c_{j',j} = p_{i',j'} + \theta \tag{15b}$$

where θ denotes the redundant safe time because of the sensor delay. We will use $\theta = 0.2\,s$ in the following simulations as it is likely ranging from 0.1 s to 0.3 s (Xiao and Gao 2011; Wang et al. 2018).

3.5. *Lower level optimisation*

After each run of the upper level optimisation, every vehicle will have an allocated arrival time. The objective of the lower level optimisation is to optimise every vehicle's trajectory to follow the allocated arrival time and maximise the terminal speed. The necessity of the lower level optimisation is that the assumed process time in the upper level may be overestimated or underestimated, which will make the signal timing plan less efficient or infeasible. So the feedback structure is one of the key contributions of this paper.

The objective function is to maximise the total vehicle's speed entering the intersection.

$$\max_{v_{i,j}^k} f_l = \sum_{i=1}^{2} \sum_{j=1}^{n_i} v_{i,j}^r \quad k = 1, 2, \cdots, r \tag{16}$$

where $v_{i,j}^r$ denotes the velocity of vehicle j on road i at the stop line.

To reduce the complexity in the trajectory planning, we assume that the vehicle updates its acceleration at an interval of 0.5 s. So there are $r = \lceil T_{i,j}/0.5 \rceil$ velocity variables before arriving at the intersection where $\lceil x \rceil$ indicates the smallest integer value which is larger than or equal to x. $r = 0$ is not considered as it indicates the vehicle already arrives at the stop line and there are no decision variables in this case. The last time interval is $t_{i,j}^r = \text{mod}(T_{i,j}, 0.5)$ where $\text{mod}(x, y)$ denotes the remainder of x divided by y. Then the total travel distance can be formulated easily as:

$$s_{i,j}^0 = \begin{cases} \frac{t_{i,j}^r}{2} \left(v_{i,j}^0 + v_{i,j}^r \right) & \text{if } r = 1 \\ \frac{0.5}{2} \left(v_{i,j}^0 + v_{i,j}^{r-1} \right) + \frac{t_{i,j}^r}{2} \left(v_{i,j}^{r-1} + v_{i,j}^r \right) & \text{if } r = 2 \\ \frac{0.5}{2} \left(v_{i,j}^0 + 2\sum_{t=1}^{r-2} v_{i,j}^t + v_{i,j}^{r-1} \right) + \frac{t_{i,j}^r}{2} \left(v_{i,j}^{r-1} + v_{i,j}^r \right) & \text{if } r > 2 \end{cases} \tag{17}$$

At every time step, the speed change is bounded by the acceleration restrictions, which is:

$$-0.5a_{min} \leq v_{i,j}^{k+1} - v_{i,j}^k \leq 0.5a_{max} \quad k = 0, 1, \ldots, r-1 \tag{18}$$

At every time step, the vehicle should keep a safe distance from the preceding vehicle as expressed in (19). These constraints are often ignored in the intersection signal optimisation, even though they impact on both the speed profile and the intersection access time.

$$s_{i,j}^k - s_{i,j-1}^k \geq S_{i,j}^k \quad k = 1, 2, \ldots, r \tag{19}$$

where $S_{i,j}^k$ denotes the safe distance at time step k which is calculated as:

$$S_{i,j}^k = \max(s_0, v_{i,j}^k h) \tag{20}$$

where s_0 denotes the minimum space headway between adjacent vehicles, which is set to 7 m.

The distance to the stop line $s_{i,j}^k$ at time step k is calculated by

$$s_{i,j}^k = \begin{cases} s_{i,j}^0 & \text{if } k = 0 \\ s_{i,j}^0 - \frac{0.5}{2} \left(v_{i,j}^0 + v_{i,j}^1 \right) & \text{if } k = 1 \\ s_{i,j}^0 - \frac{0.5}{2} \left(v_{i,j}^0 + 2\sum_{t=1}^{k-1} v_{i,j}^t + v_{i,j}^k \right) & \text{if } 1 < k < r \\ 0 & \text{if } k = r \end{cases} \tag{21}$$

3.6. *Heuristic algorithm*

Bilevel optimisation problems are $\mathcal{NP}$-Hard problems and even the simplest bilevel linear problems are nonconvex and nondifferentiable optimisation problems (Dempe 2002, 2003). For the proposed method, as the travel time is discretised in the lower level optimisation and it is also the decision variable of the upper level optimisation, the number of decision variables and constraints in the lower level optimisation are determined by the upper level optimisation. This also causes difficulties in applying KKT (Karush–Kuhn–Tucker) conditions on the lower level optimisation which is the most popular and efficient method for solving bilevel problems (Bard 1998; Lu et al. 2016).

In the proposed method, the bilevel problem has a cooperative relationship that the decision variables in the lower level help the upper level to achieve optimal goals (Zhang, Lu, and Gao 2015). A heuristic pseudo code below is proposed to solve the proposed master-slave problem:

- Step 1: In the first iteration of the upper level optimisation, the velocity entering the intersection $v_{i,j}^r$ is assumed to be the same as the initial velocity $v_{i,j}^0$.
- Step 2: Fix the terminal velocity $v_{i,j}^r$ and solve the upper level optimisation. The optimised travel time to enter the intersection T_{ij} from the upper level optimisation is passed to the constraints in the lower level optimisation.
- Step 3: Fix the travel time T_{ij} and solve the lower level optimisation. The optimised velocity entering the intersection $v_{i,j}^r$ from the lower level optimisation then becomes the constraints of the upper level optimisation in the next iteration.
- Step 4: When the difference of the total travel time between two iterations becomes less than a threshold value, which is set to be 0.5 s, the algorithm will stop. Otherwise, continue the iteration between step 2 and 3.

3.7. *Platoon-based scheduling*

This section presents a novel platoon-based method to reduce the computation burden of the proposed bilevel optimisation problem. Research on the platoon-based traffic operations has been conducted widely in the literature (see Jia and Ngoduy (2016a); Ngoduy (2013); Zhao et al. (2018) and references there-in). Allowing platoon-based operations under the connected environment may increase the roadway capacity (Jia and Ngoduy 2016a). Nevertheless, this is not always the case for the intersection. It all depends on the relationship between the time headway $\hat{h}_{i,j}$ for the vehicles on the same road and the process time $p_{i,j}$.

Considering the following example in Figure 4, the earliest time for the vehicle B to enter the intersection is the time headway h_B, while the earliest time for the vehicle C to enter the intersection is the interchange time c_C which is a function of the process time of vehicle A according to Equation (15), denoted as $c_{C,A} = g(p_A)$. Whether vehicle B or vehicle C goes first depends on the relationship between h_B and $c_{C,A}$ and can also be seen as the relationship between h_B and p_A because θ is a constant in Equation (15). In other words, we only need to calculate $h_{i,j}$ and $p_{i,j}$ for the vehicles on the same road in order to determine the passing sequence. Then we have the following scenarios as shown in Figure 4:

- If $h_B < g(p_A)$, there is no doubt that vehicle B enters the intersection before vehicle C.

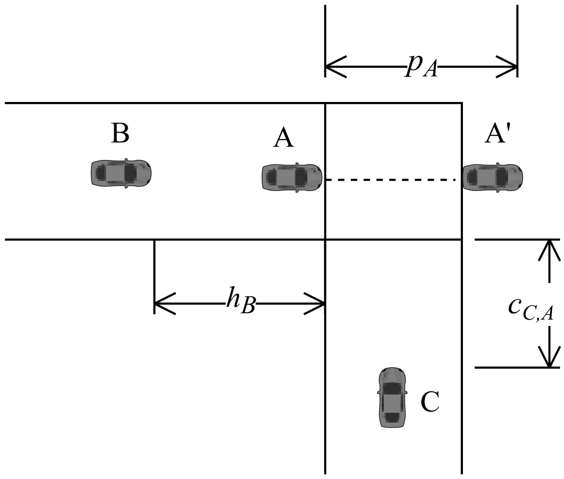


Figure 4.: The relationships among headway, process time and interchange time

- If $h_B > g(p_A)$, then vehicle C enters the intersection earlier than vehicle B to reduce the total delay.
- If $h_B = g(p_A)$, then either vehicle B or C can go first as both cases have the same total arrival time.

More generally,

- If $h_B < g(p_A)$, the intersection control strategy should give priority to the platoon on the same road to reduce the overall travel time.
- If $h_B > g(p_A)$, vehicles close to the intersection have higher priority to pass the intersection, which is similar to the first in first out (FIFO) principle.
- If $h_B = g(p_A)$, which vehicle goes first depends on who gets closer to the stop line. It is also the same as the FIFO principle.

It is clear that the platoon will only be active and beneficial to the entire traffic flow when $h_B < g(p_A)$. Please note that both h_B and p_A are closely related to vehicles' state at the stop line. When the vehicles are away from the stop line, it is essential to determine the relationship between h_B and p_A. In order to reduce the delay, the vehicle always attempts to keep a minimum time headway and achieve high speed when entering the intersection.

Proposition 3.1. *When the difference of the minimum arrival time to the stop line of two adjacent vehicles is less than or equal to the minimum time headway, these two vehicles will also keep the minimum time headway at the stop line no matter what the passing sequence is. In a mathematical form: if $T_{i,j}^{min} - T_{i,j-1}^{min} \leq h$, then $\hat{h}_{i,j} = h$.*

Proof. If the difference of the minimum arrival time of two adjacent vehicle is less than or equal to the minimum time headway, there always exists a time instant that the following vehicle has a minimum time headway to the preceding vehicle and keeps the minimum time headway until it reach the stop line. □

Proposition 3.2. *If $s_{i,j}^0 \geq \tilde{s}_{i,j}$ where $\tilde{s}_{i,j} = (v_{i,j}^0)^2/(2a_{min}) + (v_{max})^2/(2a_{max})$, then no matter what the passing sequence is, the vehicle can always achieve the maximum speed at the stop line.*

Proof. When the distance to stop line $s_{i,j}^0$ equals to $\tilde{s}_{i,j}$, then

- If $T_{i,j} = v^0_{i,j}/a_{min} + v_{max}/a_{max}$, the vehicle can reach the maximum speed as shown in Figure 5;
- If $T_{i,j} > v^0_{i,j}/a_{min} + v_{max}/a_{max}$, the vehicle can stop for a period after decelerating to zero and then start to accelerate to the maximum speed;
- If $T_{i,j} < v^0_{i,j}/a_{min} + v_{max}/a_{max}$, the vehicle can decelerate for a shorter time than $v^0_{i,j}/a_{min}$ or even do not decelerate and start to accelerate to the maximum speed.

When the distance to the stop line $s^0_{i,j}$ is larger than $\tilde{s}_{i,j}$, it can keep the same speed pattern as shown in the Figure 5 but may use a different deceleration and acceleration, and run with v_{max} finally.

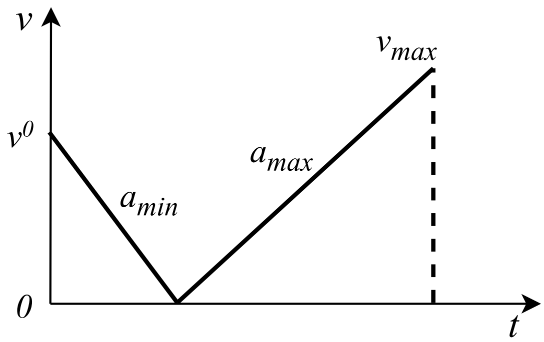


Figure 5.: The speed pattern when $s^0_{i,j} = \tilde{s}_{i,j}$ and $T_{i,j} = v^0_{i,j}/a_{min} + v_{max}/a_{max}$

$\square$

Proposition 3.3. *If the intersection length $L_s > (h-\theta)v_{max} - l$, then for the vehicles satisfying the condition in Proposition 3.1, no matter what the passing sequence is, the platoon-based operations are always preferred to the vehicle-based operations in terms of reduced delay.*

Proof. According to the Proposition 3.1, $\hat{h}_{i,j} = h$. When $L_s > (h-\theta)v_{max} - l$, then $p_{i,j} \geq \frac{L_s + l}{v_{max}} > h - \theta$ and $c_{j,j'} = p_{i,j} + \theta$, so $c_{j,j'} > h$. According to the previous discussions about headway and interchange time in this section, the platoon-based operations are preferred. Using the parameters in Table 2, $(h-\theta)v_{max} - l \approx 14.9\, m$ $\square$

Based on Propositions 3.1, 3.2 and 3.3, we can obtain the vehicle's time headway $\hat{h}_{i,j}$ and process time $p_{i,j}$ at the stop line before the upper level optimisation is executed. This can also be used as the criteria to split the traffic flow. When the following vehicle's headway is less than the process time of the preceding vehicle plus a small non-negative safety tolerance, they will form a platoon. A platoon including n vehicles will be seen as one "big vehicle" so that traffic flow is now considered to consist of many platoons moving together rather than many vehicles (or particles) moving together (Ngoduy 2013). All the vehicles in a platoon will pass the intersection sequentially without disturbances from other roads. Therefore, the number of binary variables and the total calculation time can be greatly reduced.

4. Numerical studies

We developed a simulation environment using Matlab to test the performance of the proposed method and compare it with other intersection control strategies. All simulations are carried out in a simple intersection as shown in Figure 1. The algorithm is solved using Gurobi 7.5.1 in Matlab. Typically, if there are 9 vehicles in one approach and 15 vehicles in another, every run of the upper level optimisation usually takes 0.2 s and every run of the lower level optimisation takes 0.08 s. The whole algorithm converges after two iterations, taking less than 0.5 s.

Table 2.: The basic parameters used in simulations

Parameter	Description	Value	Unit
L_r	Simulation road length	600	m
L_c	Communication range	500	m
L_s	Intersection length	10	m
l	Vehicle length	5	m
h	Saturation headway	1.5	s
h_g	safety time gap	0.8	s
θ	Safety headway tolerance	0.2	s
s_0	Minimum space headway at standstill	7	m
a_{max}	Maximum acceleration	2	m/s^2
a_{min}	Maximum deceleration	5	m/s^2
b	Comfortable braking deceleration	3	m/s^2
v_{max}	Maximum speed	55	km/h
v_{min}	Minimum speed	0	km/h
t_{delay}	Maximum allowed delay for each vehicle	30	s

The basic simulation parameters are shown in Table 2, and some of them may be changed in the simulations later when studying the impact of particular parameters on the system performance. Meanwhile, a scenario is defined as a particular set of parameters. In every scenario, the simulation will be run five times and the simulation period is 20 min in every run. The time headway of vehicles entering the studied road follows a truncated exponential distribution with minimum headway 1.5s. Four control methods are considered in the following simulation tests.

- Bilevel model (Bilevel): the method proposed in Section 3.
- Conservative scheduling model (conservative): It is a two-stage model that is similar to the bilevel model, but there is no feedback between the upper level and the lower level, and the process time is calculated in a conservative way by assuming that every vehicle may stop at the stop line. This is to ensure that every vehicle has sufficient time to pass the intersection without conflicts when the intersection controller does not know the entering speed.
- First in First out (FIFO): when a vehicle enters a predefined distance from the stop line earlier than the others, it will also have a higher priority to pass the intersection. A trajectory mapping method, which is the same as Li and Wang (2006), is also used.
- Actuated control (Actuated): traditional actuated control method is considered in the simulation. In this method, a loop detector is located at 40 m away from the stop line on every road. The green time extension interval is 3 s and the

minimum and maximum green time is chosen by running multiple simulations and selecting the best with the minimum average delay.

For simplicity, in the bilevel and conservative scheduling models, the Intelligent Driver Model (IDM) (Treiber, Hennecke, and Helbing 2000) is used to update the vehicles' speed and location before they enter the communication range. This model is chosen as it is widely used to model adaptive cruise control vehicles with a constant time headway policy (Kesting et al. 2007). Nevertheless, the proposed framework should work with other more advanced models for the connected environment, i.e. the model of Jia and Ngoduy (2016a,b). After vehicles enter the communication range, the optimisation is performed in a constant frequency which is once every 10 s if there is no special instruction and then every vehicle updates its trajectory according to the optimisation results. We assume that every vehicle is autonomous and can follow the optimisation results precisely. The IDM is also used in the FIFO and actuated control method. Briefly, the IDM describes the acceleration of the follower via the following equations:

$$a_{i,j} = a_{max} \left(1 - \left(\frac{v_{i,j}}{v_{max}} \right)^4 - \left(\frac{s^*(v_{i,j}, \Delta v_{i,j})}{\Delta s_{i,j}} \right)^2 \right) \tag{22}$$

$$s^*(v_{i,j}, \Delta v_{i,j}) = g_{i,j}^0 + v_{i,j} h_g + \frac{v_{i,j} \Delta v_{i,j}}{2\sqrt{a_{max} b}} \tag{23}$$

where the distance gap between vehicles is calculated as $\Delta s_{i,j} = x_{i,j-1} - x_{i,j} - l_{i,j-1}$, the speed difference $\Delta v_{i,j} = v_{i,j} - v_{i,j-1}$ and the minimum gap $g_{i,j}^0 = s_0 - l_{i,j-1}$.

The most important performance index in the simulation is the average delay. It is defined as the difference between the actual travel time from entering the studied road to exiting the intersection area and the minimum travel time $T_{i,j}^{min}$ calculated by equation (3) - (9). So the conflicting distance for every vehicle is the sum of the intersection length and the vehicle length. Another performance index used is the percentage of the platoon. It is defined as the ratio of number of vehicles passing the intersection in platoon to the total number of vehicles which passed the intersection during the simulation period. Here, the platoon is defined as two or more vehicles passing the intersection successively without the disturbance from the conflict roads. So the percentage of platoon indicates the frequency of changes in the right of way.

4.1. *Different demands*

Demand is one of the most important factors affecting the performance of the control strategy. Different demand levels varying from 400 to 1000 veh/h under the aforementioned four control methods are tested. The results are shown in Figure 6 and the simulated trajectories are shown in Figure 7.

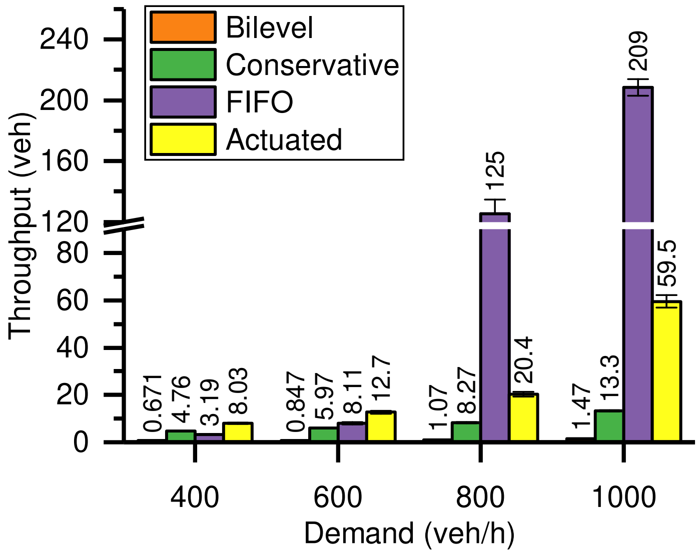
It is clear that the bilevel model has the best performance in terms of the average delay compared to all the other control schemes under all demand levels. The proposed bilevel model maintains a very low average delay and high capacity in all simulated demands. It is mainly because the time headway of two vehicles from both the same approach and conflict approaches are minimised by the model. The lower level optimisation minimised the time for vehicles passing the intersection in the same approach and the upper level optimised the passing sequence to minimise the overall delay. The main drawback of the conservative scheduling model is that it overestimates the time required by the vehicle to pass the intersection because it does not use the maximum possible speed arriving at the stop line. As a result, the conservative scheduling model has more than five times average delay than the proposed bilevel method, but it still works quite well in high demands compared to the FIFO and actuated control methods. In particular, the FIFO works well only in low demands which is also found by Li and Zhou (2017); when the demand increases, the control performance deteriorates quickly in both average delay and throughput. This is because even though the priority of passing is determined by the FIFO, the trajectory is not optimised according to such priority. Even though the trajectory mapping technique is applied, and the vehicle can avoid a complete stop at the stop line in most situations, it sometimes has a low speed when crossing the stop line because it needs to wait for the vehicle with a higher priority in the other approach. Such low speed causes negative impact on the capacity and defers all the vehicles with lower priority. This can be clearly seen in Figure 7c. The actuated control is the worst in the demand levels of 400 veh/h and 600 veh/h but works better than the FIFO method in higher demands. The reason is that although some vehicles have to stop in front of the stop line in high demands, the following vehicles can still accelerate and have higher speeds when crossing the intersection area. This implies that when there is no trajectory optimisation, the platoon can increase the capacity in high demands.

The four control methods show different throughout in Figure 6b because of different capacities. The throughput of the intersection in FIFO does not change when the demand keeps increasing from 600 veh/h. This indicates that it already reaches its capacity in the demand of 600 veh/h from each approach. However, the intersection in all the other three methods are in unsaturated state in all the simulated traffic demand levels. The throughput in the actuated control method is about 7% less than that in other two optimisation-based methods in the high demand of 1000 veh/h because of the loss time of stop-and-go behaviour. Though the two optimisation-based methods have similar throughout, they have quite different passing behaviours. Vehicles in the conservative scheduling model tend to pass in platoons, and the percentage of platoon in the conservative scheduling model is much higher than that in the bilevel programming model. This is mainly because of the longer process time in the conservative scheduling model. However, the percentage of platoon has the same trend of increasing with respect to the demand in both models.

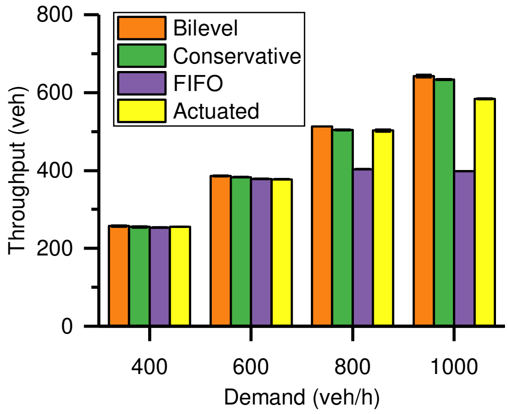
4.2. *Different intersection lengths*

Another important factor that is often ignored is the relationship between the minimum headway on the same road and the minimum process time in the conflict approach. Usually, the maximum allowed speed in the intersection area is the same, so the factor that needs to be investigated is the intersection length. Simulation results of intersection lengths varying from 10 m to 30 m are shown in Figure 8.

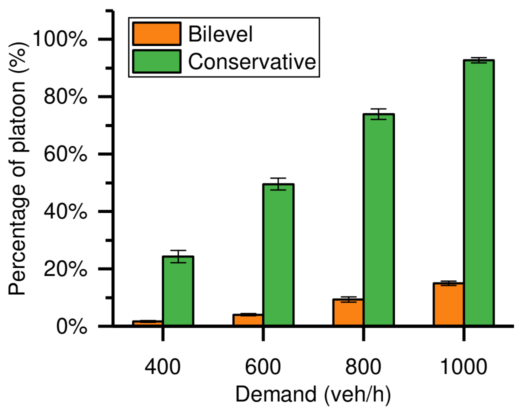
In these simulations, the demand is kept to 600 veh/h on each road. It is shown that the longer the intersection, the larger the average delay in all four studied methods. Nevertheless, the sensitivity varies greatly. Figure 8b shows the rate of change in the average delay for the four methods with respect to different intersection lengths. In this case, we use the result when the intersection length is 10 m as the baseline. It appears that the FIFO method is the most sensitive to the intersection length. A



(a) Average delay



(b) Throughput



(c) The percentage of platoon

Figure 6.: Simulation results under different demands

possible reason is that the demand is already close to the intersection capacity and the additional process time due to the longer intersection causes serious congestion on the road and increases the average delay rapidly. The actuated control does not show obvious sensitivity to the intersection length since the intersection length does not affect the signal timing in the actuated control and some vehicles can achieve a much higher speed entering the intersection compared to the FIFO method. The bilevel model is more sensitive to the intersection length than the conservative scheduling method because of their different ways in calculating the process time. It is worth noticing that a longer intersection also prompts the vehicles to pass in platoons in both bilevel method and conservative scheduling method, where the former is more sensitive but the delay is significantly lower. This is due to the fact that the bilevel method makes better use of the intersection conflict area through the feedback between the signal optimisation and the trajectory optimisation. We can also observe a great increase in the percentage of platoon when the intersection length increases from 10m to 15m. This can be explained by the proposition 3.3. When the intersection length is greater than a value (14.9 m using the data in this paper), more vehicles tend to pass the intersection in platoons.

4.3. *Different communication ranges*

Different communication ranges varying from 100 m to 500 m are tested for the proposed model. The optimisation frequency increases from every 10 s to 5 s to avoid unoptimised vehicles being too close to the stop line and cannot find a feasible solution. The demand is kept to 600 veh/h in every simulation. The simulation results are shown in Figure 9.

We can see that the average delay keeps decreasing with the increasing communication range when the communication range is greater than 200 m. The average delay when the communication range is 500m is more than 36% less than that when the communication range is 100m. Nevertheless, the average delay is still quite small in every case. When the communication range is 100 m, vehicles are too close to the intersection and have little room to be optimised. The percentage of platoon are extremely higher than that in longer communication range. When the communication range is 200 m, it becomes another extreme case that the priority of passing keeps changing between two approaches. There are few number of vehicles available to be optimised and the proposed method degenerates to a control scheme that is similar to the reservation method. The throughput keeps the same among all scenarios. So the suggested communication range should be at least 300 m to ensure the traffic flow operates stably and the communication range of 500 m is preferred in terms of improving the control performance.

4.4. *Different traffic compositions*

To model the performance of the proposed method under heterogeneous traffic composition, we assume the traffic consists of different shares of buses. More stochastic vehicle characteristics are modelled in both cars and buses. As even in ACC or autonomous vehicles, the drivers still have some preferences for the ACC or autonomous vehicles that can be personalised, their characteristics may not be the same (Butakov and Ioannou 2016; Ghiasi et al. 2017). Instead of using identical maximum speed as the desired speed for every vehicle, the desired speed of cars follows a truncated normal distribution $N(50, 2^2)$ within the interval $[40, 55]$, while that of buses follows a truncated normal distribution $N(45, 2^2)$ within the interval $[40, 55]$. We use the same bound for the desired speed of cars and buses to ensure that they are not too low or too high that exceed the maximum allowed speed. Other properties of buses are length (10 m), maximum acceleration ($1.5\,\mathrm{m/s^2}$), minimum deceleration ($3\,\mathrm{m/s^2}$). The safety time headway of a car or bus following a bus is 2s. The total demand is kept to 600 veh/h on each road. Every vehicle's distinct characteristics are considered in the vehicle-based intersection control rather than to be assumed the same in the traditional flow-based intersection control.

Remark 1. There is no bus stop near the intersection in our study in this proof of concept case study.

The simulation results are shown in Figure 10. We can see that the average delay increases with respect to the percentage of buses. That is rather obvious due to the lower desired speed and lower maximum acceleration of the bus. Comparing to the simulation in section 4.1 at the same demand level, the average delay with 0% bus increases by 71.8% from 0.85 s/veh to 1.46 s/veh which implies that the traffic heterogeneity has a negative impact on the performance. In another aspect, even with 20% of buses, the average delay is still quite small, and much lower than that given by the other methods in Figure 6 at the same total demand level. On the other hand, the vehicles are less likely to pass in a platoon with the increasing percentage of buses because of the higher time headway of buses. Moreover, the percentages of platoons under different percentages of buses are all greater than 20%, which implies that the platoon-based operations help reduce the delay in the heterogeneous traffic flow. No clear throughput changes are observed with the increasing percentage of buses.

5. Concluding remarks

This paper developed a novel way to jointly optimise the intersection control and the trajectory. To relax the commonly used assumption of constant process time for the vehicle to pass the intersection, the vehicle's detailed trajectory has been considered in the intersection control. This approach can fully utilise the available information from the intersection controller and vehicles to increase the efficiency of the intersection. The problem was formulated as a bilevel programming model in which the upper level optimisation is a mixed integer linear programming model to minimise the total travel time. In contrast, the lower level optimisation is a linear programming model to maximise the total speed entering the intersection. The coupled structure between the upper level and the lower level is a major contribution of this paper. This proposed method can achieve better intersection control performance, and the results are safe and feasible for every vehicle. Moreover, after a discussion on the relationship between the safe time headway and the process time, a novel platoon split method has been developed to improve the performance of the model. We showed that integrating the trajectory planning into the signal optimisation can significantly improve the intersection control performance. Platoon is not always beneficial to the intersection. We discussed the condition how the platoon-based approach can be applied to reduce the number of binary variables.

The models and findings established in this paper can be extended in the following aspects in the future. First, although the number of binary variables is greatly reduced due to the dynamic platoon formation, a more efficient heuristic method is still needed to enable a real-time application in the future. One possible way is to apply signal phases to reduce the number of binary variables (Xu et al. 2017; Yu et al. 2018), but this also reduces the flexibility of the intersection control. Platoon-based approach is not always beneficial for reducing the delay. Second, in this paper all vehicles were assumed to be autonomous, so the obtained results provide the upper bounds of the benefit of the autonomous vehicles to the urban intersection operations. However, this assumption is not practical in the near future. A more realistic situation would be the mixed traffic consisting of vehicles with different levels of automation and connectivity. In this case, how to develop a robust and efficient intersection control method will be an interesting research question (Yang, Guler, and Menendez 2016; Li and Zhou 2017), which should be left in our future research. At last, we only applied the bilevel model to a simplified intersection. If the intersection is complex, more conflict zones will exist and it will require more binary variables. Note that the solver time also increases with respect to the number of binary variables. Some simplified ways may be applied to reduce the complexity of the problem, such as the aforementioned platoon-based approach or reducing the communication range.

Acknowledgement

The first author would like to acknowledge the financial support of the Leeds International Research Scholarship (LIRS) in University of Leeds for his PhD research. An earlier version of this paper was presented at the Transportation Research Board conference in January 2018. The authors wish to thank the anonymous peer reviewers.

References

Bard, Jonathan F. 1998. *Practical Bilevel Optimization: Algorithms and Applications*. Vol. 30 of *Nonconvex Optimization and Its Applications*. Boston, MA: Springer US.

Bazzan, Ana L. C. 2005. "A Distributed Approach for Coordination of Traffic Signal Agents." *Autonomous Agents and Multi-Agent Systems* 10 (1): 131–164.

Butakov, V. A., and P. Ioannou. 2016. "Personalized Driver Assistance for Signalized Intersections Using V2I Communication." *IEEE Transactions on Intelligent Transportation Systems* 17 (7): 1910–1919.

Dempe, Stephan. 2002. *Foundations of Bilevel Programming*. Vol. 61 of *Nonconvex Optimization and Its Applications*. Boston, MA: Kluwer Academic Publishers.

Dempe, Stephan. 2003. "Annotated Bibliography on Bilevel Programming and Mathematical Programs with Equilibrium Constraints." *Optimization* 52 (3): 333–359.

Diakaki, Christina, Markos Papageorgiou, and Kostas Aboudolas. 2002. "A multivariable regulator approach to traffic-responsive network-wide signal control." *Control Engineering Practice* 10 (2): 183–195.

Diakaki, Christina, Markos Papageorgiou, Ioannis Papamichail, and Ioannis Nikolos. 2015. "Overview and analysis of Vehicle Automation and Communication Systems from a motorway traffic management perspective." *Transportation Research Part A: Policy and Practice* 75: 147–165.

Dresner, Kurt, and Peter Stone. 2004. "Multiagent Traffic Management: A Reservation-Based Intersection Control Mechanism." In *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems - Volume 2*, AAMAS '04, New York, USA, 530–537. IEEE Computer Society.

Dresner, Kurt, and Peter Stone. 2005. "Multiagent Traffic Management: An Improved Intersection Control Mechanism." In *Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems - AAMAS '05*, Utrecht, The Netherlands, 471. ACM Press.

Dresner, Kurt, and Peter Stone. 2006. "Human-usable and emergency vehicle-aware control policies for autonomous intersection management." In *Fourth International Workshop on Agents in Traffic and Transportation (ATT), Hakodate, Japan*, .

Dresner, Kurt, and Peter Stone. 2008. "A multiagent approach to autonomous intersection management." *Journal of artificial intelligence research* 591–656.

Feng, Yiheng, K. Larry Head, Shayan Khoshmagham, and Mehdi Zamanipour. 2015. "A real-time adaptive signal control in a connected vehicle environment." *Transportation Research Part C: Emerging Technologies* 55: 460–473.

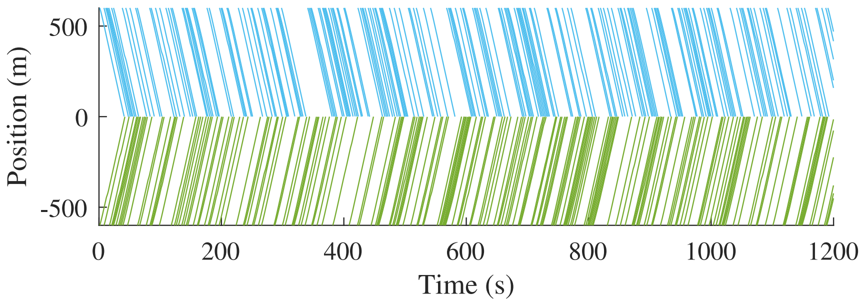
Ghiasi, Amir, Omar Hussain, Zhen (Sean) Qian, and Xiaopeng Li. 2017. "A Mixed Traffic Capacity Analysis and Lane Management Model for Connected Automated Vehicles: A Markov Chain Method." *Transportation Research Part B: Methodological* 106 (Supplement C): 266–292.

Goodall, Noah, Brian Smith, and Byungkyu Park. 2013. "Traffic Signal Control with Connected Vehicles." *Transportation Research Record: Journal of the Transportation Research Board* 2381: 65–72.

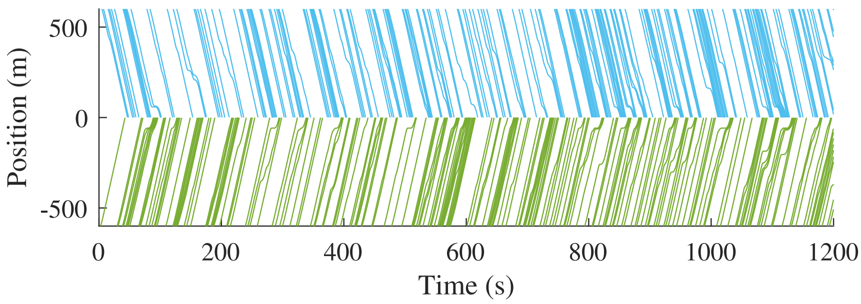
Guler, S. Ilgin, Monica Menendez, and Linus Meier. 2014. "Using connected vehicle technology to improve the efficiency of intersections." *Transportation Research Part C: Emerging*

Technologies 46: 121–131.
Gurobi Optimization, Inc. 2017. “Gurobi Optimizer Reference Manual.” http://www.gurobi.com.
IBM, Corp. 2017. “IBM ILOG CPLEX Optimization Studio.” https://www-01.ibm.com/software/commerce/optimization/cplex-optimizer/.
Isukapati, I. K., and S. F. Smith. 2017. “Accommodating High Value-of-Time Drivers in Market-Driven Traffic Signal Control.” In *2017 IEEE Intelligent Vehicles Symposium (IV)*, Jun., 1280–1286.
Jia, D., and D. Ngoduy. 2016a. “Platoon based Cooperative Driving Model with Consideration of Realistic Inter-vehicle Communication.” *Transportation Research Part C* 68: 245—-264.
Jia, D., and D. Ngoduy. 2016b. “Enhanced cooperative car-following traffic model with the combination of V2V and V2I communication.” *Transportation Research Part B* 90: 172—-191.
Kamal, M.A.S., J.-I. Imura, T. Hayakawa, A. Ohata, and K. Aihara. 2015. “A Vehicle-Intersection Coordination Scheme for Smooth Flows of Traffic Without Using Traffic Lights.” *IEEE Transactions on Intelligent Transportation Systems* 16 (3): 1136–1147.
Kesting, Arne, Martin Treiber, Martin Schönhof, and Dirk Helbing. 2007. “Extending Adaptive Cruise Control to Adaptive Driving Strategies.” *Transportation Research Record: Journal of the Transportation Research Board* 2000: 16–24.
Lamorgese, Leonardo, Carlo Mannino, and Mauro Piacentini. 2016. “Optimal train dispatching by Benders’-like reformulation.” *Transportation Science* 50 (3): 910–925.
Lee, Joyoung, and Byungkyu Park. 2012. “Development and Evaluation of a Cooperative Vehicle Intersection Control Algorithm Under the Connected Vehicles Environment.” *IEEE Transactions on Intelligent Transportation Systems* 13 (1): 81–90.
Levin, Michael W., Stephen D. Boyles, and Rahul Patel. 2016. “Paradoxes of reservation-based intersection controls in traffic networks.” *Transportation Research Part A: Policy and Practice* 90: 14–25.
Li, Li, and Fei-Yue Wang. 2006. “Cooperative Driving at Blind Crossings Using Intervehicle Communication.” *IEEE Transactions on Vehicular Technology* 55 (6): 1712–1724.
Li, Pengfei (Taylor), and Xuesong Zhou. 2017. “Recasting and Optimizing Intersection Automation as a Connected-and-Automated-Vehicle (CAV) Scheduling Problem: A Sequential Branch-and-Bound Search Approach in Phase-Time-Traffic Hypernetwork.” *Transportation Research Part B: Methodological* 105 (Supplement C): 479–506.
Li, Zhuofei, Lily Elefteriadou, and Sanjay Ranka. 2014. “Signal control optimization for automated vehicles at isolated signalized intersections.” *Transportation Research Part C: Emerging Technologies* 49: 1–18.
Lu, Jie, Jialin Han, Yaoguang Hu, and Guangquan Zhang. 2016. “Multilevel Decision-Making: A Survey.” *Information Sciences* 346–347: 463–487.
Ma, Jiaqi, Xiaopeng Li, Fang Zhou, Jia Hu, and B. Brian Park. 2017. “Parsimonious Shooting Heuristic for Trajectory Design of Connected Automated Traffic Part II: Computational Issues and Optimization.” *Transportation Research Part B: Methodological* 95: 421–441.
Müller, Eduardo Rauh, Rodrigo Castelan Carlson, and Werner Kraus Junior. 2016. “Intersection control for automated vehicles with MILP.” *IFAC-PapersOnLine* 49 (3): 37–42.
Nemhauser, George L., and Laurence A. Wolsey. 1999. *Integer and Combinatorial Optimization.* 1st ed. New York, NY: Wiley-Blackwell.
Ngoduy, D. 2013. “Platoon-based macroscopic model for intelligent traffic flow.” *Transportmetrica B: Transport Dynamics* 1 (2): 153–169.
Rakha, Hesham, Ahmed Amer, and Ihab El-Shawarby. 2008. “Modeling Driver Behavior Within a Signalized Intersection Approach Decision-Dilemma Zone.” *Transportation Research Record: Journal of the Transportation Research Board* 2069: 16–25.
Sharon, Guni, Roni Stern, Ariel Felner, and Nathan R. Sturtevant. 2015. “Conflict-based search for optimal multi-agent pathfinding.” *Artificial Intelligence* 219: 40–66.
Sun, Weili, Jianfeng Zheng, and Henry X. Liu. 2017. “A capacity maximization scheme for intersection management with automated vehicles.” *Transportation Research Procedia* 23:

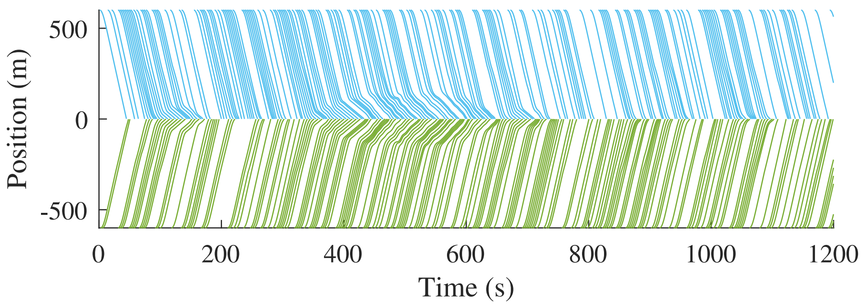
121–136.
Treiber, Martin, Ansgar Hennecke, and Dirk Helbing. 2000. “Congested traffic states in empirical observations and microscopic simulations.” *Physical Review E* 62 (2): 1805–1824.
Wang, M., S. P. Hoogendoorn, W. Daamen, B. van Arem, B. Shyrokau, and R. Happee. 2018. “Delay-Compensating Strategy to Enhance String Stability of Adaptive Cruise Controlled Vehicles.” *Transportmetrica B: Transport Dynamics* 6 (3): 211–229.
Xiao, L., and F. Gao. 2011. “Practical String Stability of Platoon of Adaptive Cruise Control Vehicles.” *IEEE Transactions on Intelligent Transportation Systems* 12 (4): 1184–1194.
Xie, Xiao-Feng, Stephen F. Smith, Liang Lu, and Gregory J. Barlow. 2012. “Schedule-driven intersection control.” *Transportation Research Part C: Emerging Technologies* 24: 168–189.
Xu, Biao, Xuegang (Jeff) Ban, Yougang Bian, Jianqiang Wang, and Keqiang Li. 2017. “V2I Based Cooperation between Traffic Signal and Approaching Automated Vehicles.” In *2017 IEEE Intelligent Vehicles Symposium (IV)*, Los Angeles, CA, USA, Jun., 1658–1664. IEEE.
Yang, Bo, and Christopher Monterola. 2016. “Efficient intersection control for minimally guided vehicles: A self-organised and decentralised approach.” *Transportation Research Part C: Emerging Technologies* 72: 283–305.
Yang, Kaidi, S. Ilgin Guler, and Monica Menendez. 2016. “Isolated intersection control for various levels of vehicle technology: Conventional, connected, and automated vehicles.” *Transportation Research Part C: Emerging Technologies* 72: 109–129.
Yu, Chunhui, Yiheng Feng, Henry X. Liu, Wanjing Ma, and Xiaoguang Yang. 2018. “Integrated Optimization of Traffic Signals and Vehicle Trajectories at Isolated Urban Intersections.” *Transportation Research Part B: Methodological* 112: 89–112.
Zhang, Guangquan, Jie Lu, and Ya Gao. 2015. *Multi-Level Decision Making: Models, Methods and Applications*. Intelligent Systems Reference Library. Berlin Heidelberg: Springer-Verlag.
Zhang, Kailong, Dafang Zhang, A. de La Fortelle, Xiao Wu, and J. Gregoire. 2015. “State-Driven Priority Scheduling Mechanisms for Driverless Vehicles Approaching Intersections.” *IEEE Transactions on Intelligent Transportation Systems* 16 (5): 2487–2500.
Zhao, Weiming, Dong Ngoduy, Simon Shepherd, Ronghui Liu, and Markos Papageorgiou. 2018. “A Platoon Based Cooperative Eco-Driving Model for Mixed Automated and Human-Driven Vehicles at a Signalised Intersection.” *Transportation Research Part C: Emerging Technologies* 95: 802–821.
Zhu, Feng, and Satish V. Ukkusuri. 2015. “A linear programming formulation for autonomous intersection control within a dynamic traffic assignment and connected vehicle environment.” *Transportation Research Part C: Emerging Technologies* 55: 363–378.



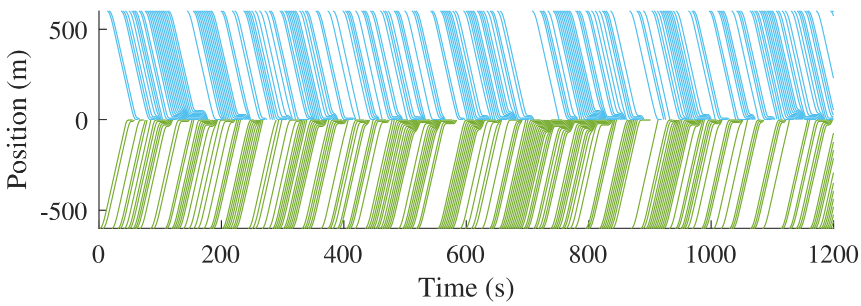
(a) Bilevel



(b) Conservative

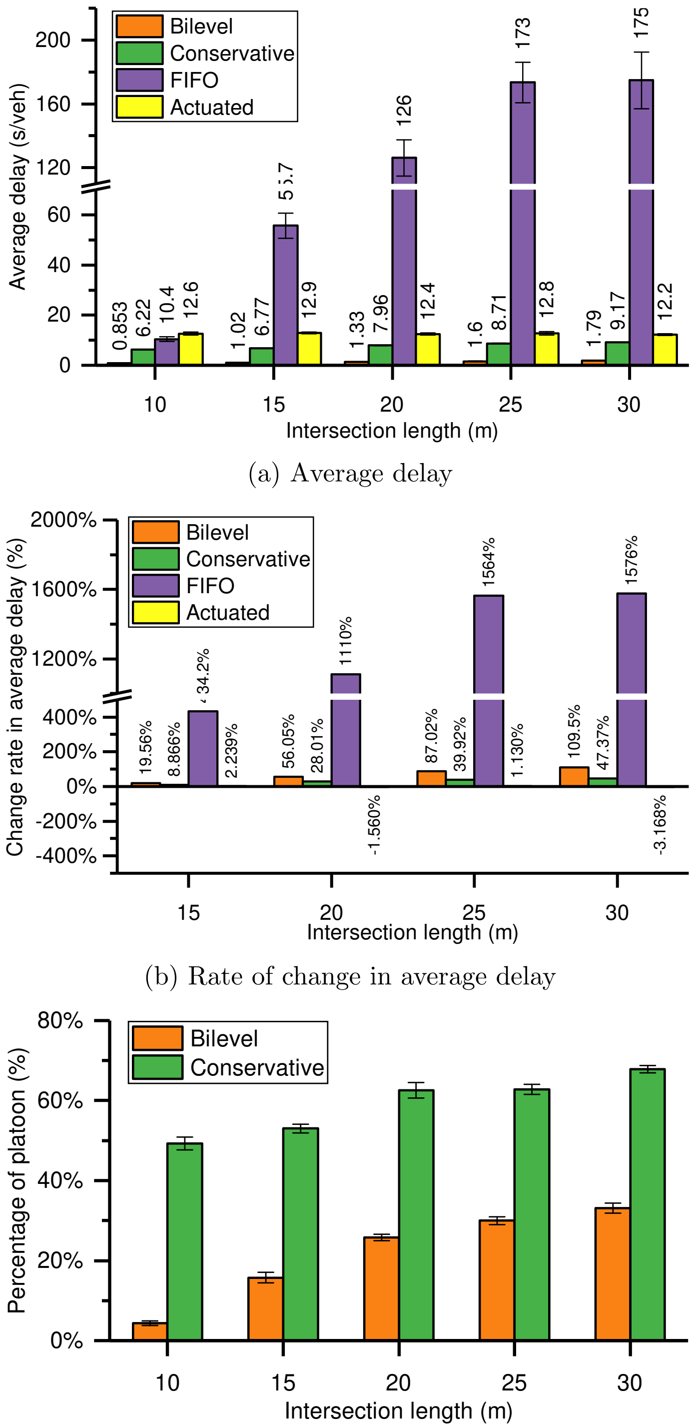


(c) FIFO



(d) Actuated

Figure 7.: Simulated trajectories under different control methods

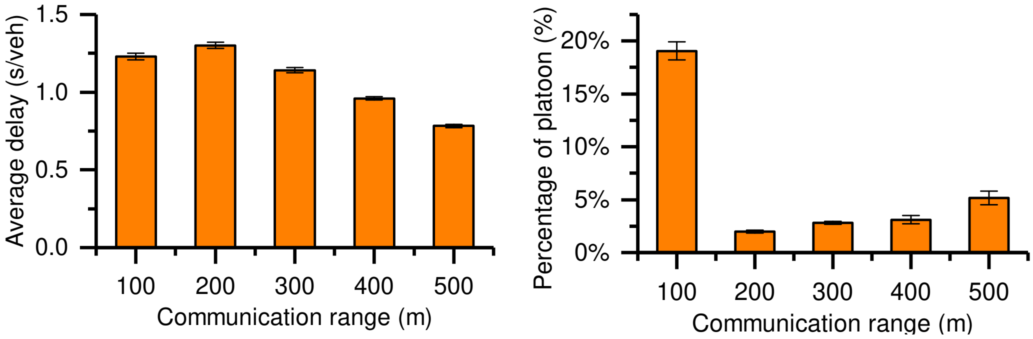


(a) Average delay

(b) Rate of change in average delay

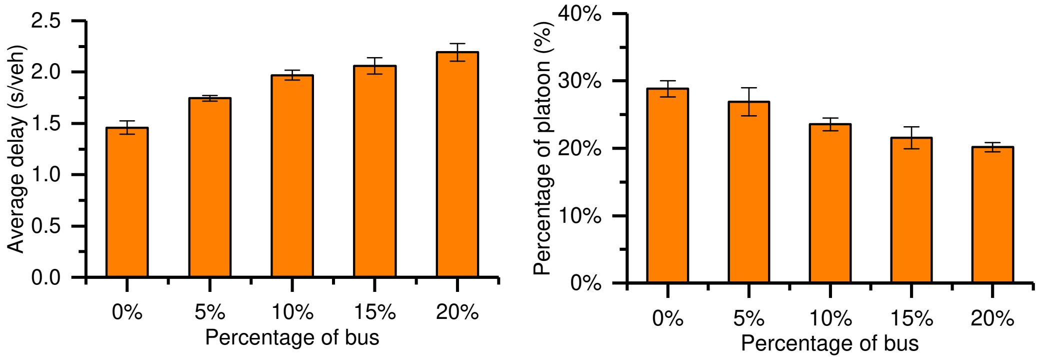
(c) The percentage of platoon

Figure 8.: Simulation results under different intersection lengths



(a) Average delay (b) The percentage of platoon

Figure 9.: Simulation results under different communication ranges



(a) Average delay (b) The percentage of platoon

Figure 10.: Simulation result under different traffic compositions