

Article

# A CAV Platoon Control Method for Isolated Intersections: Guaranteed Feasible Multi-Objective Approach with Priority

Chen Wang, Yulu Dai and Jingxin Xia \*

Intelligent Transportation Systems Research Center, Southeast University, Nanjing 211189, China; wkobec@hotmail.com (C.W.); 220183101@seu.edu.cn (Y.D.)

\* Correspondence: xiajingxin@seu.edu.cn

Received: 17 December 2019; Accepted: 23 January 2020; Published: 1 February 2020



**Abstract:** This paper proposed a multi-objective guaranteed feasible connected and autonomous vehicle (CAV) platoon control method for signalized isolated intersections with priorities. Specifically, we prioritized the intersection throughput and traffic efficiency under a pre-defined signal cycle, based on which we minimized fuel consumption and emissions for CAV platoons. Longitudinal safety was also considered as a necessary condition. To handle the aforementioned targets, we firstly designed a vehicular sub-platoon splitting algorithm based on Farkas lemma to accommodate a maximum number of vehicles for each signal green time phase. Secondly, the CAV optimal trajectories control algorithm was designed as a centralized cooperative model predictive control (MPC). Moreover, the optimal control problem was formulated as discrete linear quadratic control problems with constraints with receding predictive horizons, which can be efficiently solved by quadratic programming after reformulation. For rigor, the proofs of the recursive feasibility and asymptotic stability of our proposed predictive control model were provided. For evaluation, the performance of the control algorithm was compared against a non-cooperative distributed CAV control through simulation. It was found that the proposed method can significantly enhance both traffic efficiency and energy efficiency with ensured safety for CAV platoons at urban signalized intersections.

**Keywords:** isolated signalized intersection CAV platoon control; multi-objective optimal trajectory control; guaranteed feasibility; asymptotical stability; sub-platoons splitting algorithm; model predictive control (MPC)

## 1. Introduction

Connected and autonomous vehicles (CAVs) are expected to improve traffic efficiency, safety, and reduce fuel consumption and emissions. In a CAV environment, vehicles can communicate with each other (i.e., vehicle to vehicle, V2V) and with infrastructure (i.e., vehicle to infrastructure, V2I) [1–3] for critical real-time information exchange (e.g., location, speed, acceleration of CAVs, traffic signals, traffic flow conditions, etc.), which provides important elements for vehicles' trajectory optimization. In the past studies, various CAV trajectory control strategies had been proposed, with multiple objectives of maximizing traffic throughput, minimizing fuel consumption and emission, and ensuring driving safety.

Depending on roadway infrastructure type, previous CAV research can be categorized into two kinds: CAV trajectory control on freeways and CAV trajectory control on urban roads. The studies on freeway focused on optimizing vehicle trajectories [4–8] to deal with speed oscillation in stop-and-go conditions [9–15] and dangerous situations caused by lane change and merges [16–20]. Research on

urban roads mainly dealt with CAV trajectory control at intersections [21–24], where traffic flow can be interrupted by signals. Although a lot of achievements have been made in this field, most of them focused on single vehicle trajectory optimization, which could get local optimal solutions for a whole CAV platoon. Regarding CAV platoon control at urban intersections, at least two additional aspects need to be carefully considered: (1) A vehicular sub-platoon splitting algorithm may need to be proposed due to traffic signals; (2) when multiple objectives (e.g., efficiency, energy-efficiency, and safety) are considered for CAV platoon control, the optimization problems could become very complicated.

To be specific, it could be possible that a CAV platoon cannot cross an intersection within one green cycle. In this situation, if the control objective is maximizing the intersection throughput, some vehicles have to stop and wait for the next green, causing high fuel consumption and potential safety issues [25,26]. Thus, it is necessary to allow more vehicles pass through the intersection and keep each vehicle safe in the current platoon, while controlling other vehicles safety and minimizing their fuel consumption and emissions before the next green time. However, this raises a feasibility problem whether each vehicle in the platoon can pass the intersection or not within the current green, based on which the platoon could be split into two parts at the early stage. Moreover, if applying a centralized control strategy, the original platoon may need to be split into several sub-units, depending on how many cycles all vehicles in the original platoon can efficiently, safely, and energy-efficiently pass through the intersection. For a decentralized control strategy, the downstream CAV trajectory will be determined by leading vehicles so that the throughput of the signalized intersection could be affected.

Regarding control methods, linear control [27,28] and model predictive control (MPC) [5,7] have been utilized for CAV trajectory optimization. Linear control is fast computing but lacks hard constraints. On the other hand, MPC has clear multi-objectives and hard constraints, while it requires heavier computation loads and has no guaranteed solutions. In an urban environment with signal constraints and multiple optimization objectives [29], MPC appears to be a more feasible solution than linear controllers.

Considering these, the objective of this paper is to develop a CAV platoon control algorithm for an isolated signalized intersection under a fully connected and automated environment. Particularly, we took the efficiency that maximizes intersection throughput within the given green signal time as the main priority and let minimizing emission as the second priority. Longitudinal safety was also considered as a necessary condition. Based on those, we determined the number of vehicles that can pass through an intersection during each green time, and split them into multiple sub-platoons. In this study, we formulated the problem of splitting CAV platoon into a quadratic programming (QP) problem. We proposed a centralized cooperative model predictive control to optimize CAV trajectories to achieve the control objectives. The algorithm was compared to a decentralized CAV control method to evaluate its effectiveness and robustness, based on simulation experiments.

The remainder of the paper is organized as follows: Section 2 provides the assumptions and state-space formulation of CAV platoon; Section 3 introduces the MPC control method and quadratic programming; Section 4 presents the formulation of CAV platoon optimal control and solving algorithm. Section 5 compares the proposed control method with a decentralized control method, based on numerical experiments. Section 6 concludes the research.

## 2. Assumptions and State-Space Formulation

This section presents the assumptions and the state-space systems formulations for CAV platoon control. We present formulations for both single CAV and CAV platoons to reflect the difference between their control objectives.

In a CAV environment, all vehicles can access and share information from V2V and V2I technology. The model assumptions are as follows:

1. All vehicles are automated and the delay time of information transmission can be negligible. Thus, all CAV can be instantly controlled by a control algorithm;

2. All vehicles are equipped with sensors, allowing them to monitor signal timing and real-time position and velocity of other CAVs;
3. There is only one lane in one direction for the intersection, so that overtaking is not considered;
4. The sample time ( $\Delta t_s$ ) is short enough for the control input to be determined and all horizons for platoons can be divisible by  $\Delta t_s$ .

Based on the assumptions above, the state-space systems of single CAV control and CAV platoon control are formulated below.

According to the vehicle kinematics equation following uniform motion law, the speed of  $i$ th CAV at time  $t + 1$  is determined by its speed and acceleration at time  $t$ . Moreover, the position of  $i$ th CAV at time  $t + 1$  is related to its acceleration and speed at time  $t$ . Let  $P$ ,  $V$ , and  $a$  define the position, speed, and acceleration of CAV, respectively, and the relationship between them can be formulated as below:

$$V_{i,t+1} = V_{i,t} + a_{i,t} \times \Delta t_s \quad (1)$$

$$P_{i,t+1} = P_{i,t} + V_{i,t} \times \Delta t_s + \frac{1}{2} a_{i,t} \times \Delta t_s^2 \quad (2)$$

where  $\Delta t_s$  is the sample time. Therefore, the state-space formulation for a single CAV is:

$$X_{i,t+1} = \begin{bmatrix} 1 & \Delta t_s \\ 0 & 1 \end{bmatrix} X_{i,t} + \begin{bmatrix} \frac{1}{2} \Delta t_s^2 \\ \Delta t_s \end{bmatrix} \quad (3)$$

where

$$X_{i,t} = \begin{bmatrix} P_{i,t} \\ V_{i,t} \end{bmatrix} \quad u_{i,t} = a_{i,t} \quad (4)$$

For a CAV platoon, the state  $X$  contains the speed and position values of all CAVs at time  $t$ , and the input  $u$  contains the accelerate values of all CAVs at time  $t$ :

$$X_t = [ P_{1,t} \quad V_{1,t} \quad \cdots \quad P_{n,t} \quad V_{n,t} ]^T, X_t \in R^{2n} \quad (5)$$

$$u_t = [ a_{1,t} \quad \cdots \quad a_{n,t} ]^T, u_t \in R^n \quad (6)$$

Thus, the state-space formulation for a CAV platoon is:

$$X_{t+1} = \begin{bmatrix} 1 & \Delta t_s & \cdots & 0 \\ 0 & 1 & & \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 1 & \Delta t_s \\ & & 0 & 1 \end{bmatrix} X_t + \begin{bmatrix} \frac{1}{2} \Delta t_s^2 & \cdots & 0 \\ \Delta t_s & & \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \frac{1}{2} \Delta t_s^2 \\ & & & \Delta t_s \end{bmatrix} u_t \quad (7)$$

For leading vehicles, there are three constraints: Speed constraint, acceleration constraint, and green time constraint. The expression of speed constraint is:

$$0 \leq V_{n,t} \leq V_{free} \quad (8)$$

the expression of acceleration constraint is:

$$a_{min} \leq a_{n,t} \leq a_{max} \quad (9)$$

and green time constraint can be written as:

$$l \leq P_{n,g_{k,e}} \quad (10)$$

where

$l$  is the position of the intersection;

$P_{n,g_{k,e}}$  is the position of vehicle  $n$  at the end time  $e$  of cycle  $k$ .

For following vehicles, apart from the above constraints, they also have a distance constraint:

$$L_i + V_{n,t} \times \tau_{safe} \leq P_{n-1,t} - P_{n,t} \tag{11}$$

where  $\tau_{safe}$  is the target safe headway.

### 3. Model Predictive Control and Quadratic Programming

This section presents the model predictive control (MPC) method and quadratic programming, which were used to optimize CAV trajectories.

#### 3.1. Model Predictive Control for CAV Platoon

The model predictive control method was used to optimize CAV platoon trajectories in this study. MPC has been utilized for various dynamic systems, since it was firstly introduced for process control engineering by Cutler and Ramaker and Richalet et al. It is an online optimization control technique that uses continuous or discrete models to predict the future behavior of optimal control objects by solving constrained optimization problems in a finite time domain, which contain three basic elements including model predictive, rolling optimization, and feedback correction. By applying MPC, the control value can be dynamically adjusted to the desired range to obtain an optimal control result.

As shown in Figure 1, we used different predictive horizons of MPC for each platoon in this study. As such, the control and predict processes is changing. For example, using  $g_{1,e}$  to represent the end of the first green time, the horizon of platoon 1 is  $(t_0, t_{g_{1,e}})$  and  $(t_1, t_{g_{1,e}})$  at time  $t_0$  and  $t_1$ , respectively. Based on an objective function, MPC predicts the future output sequence of a platoon. By comparing it to the actual control output, MPC corrects the predicted output sequence of the previous moment  $t_1$ . The process is iteratively applied until the horizon is equal to zero.

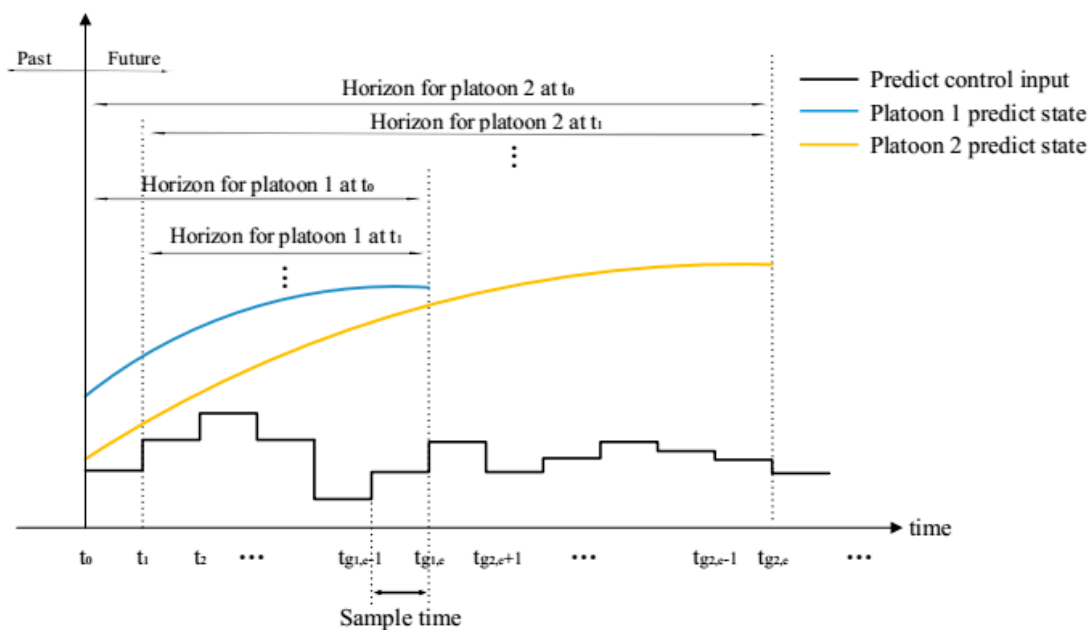


Figure 1. Model predictive control for connected and autonomous vehicle (CAV) platoons.

As for state-space formulation of MPC, we used the classical  $m$ -input,  $n$ -state, and  $p$ -output time-invariant system. The system can be approximated by difference equations as follows:

$$x(k+1) = f(x(k), u(k)) \quad (12)$$

$$y(k) = g(x(k)) \quad (13)$$

where  $x$  indicates the vector of states, while  $u$  represents the input vector. The current state and output are defined by function  $f$ , while the system outputs are denoted by function  $g$ , assuming there is no input feedforward.

Constraints on the input and state vectors are assumed to be in the form:

$$u(k) \in U \subseteq R^m \quad (14)$$

$$x(k) \in X \subseteq R^n \quad (15)$$

where

$U$  is normally a convex, compact subset of  $R^m$ ;

$X$  is normally a convex, closed subset of  $R^n$ .

In order to drive a system to an equilibrium state or to the origin, a controller can be defined with the following dynamics:

$$x(k|k) = x(k) \quad (16)$$

$$x(k+j+1|k) = f(x(k+j|k), u(k+j|k)) \quad (17)$$

$$x(k+j|k) = g(x(k+j|k)) \quad (18)$$

subject to the constraints:

$$u(k+j|k) \in U \quad (19)$$

$$x(k+j|k) \in X \quad (20)$$

$$x(k+N|k) \in T \quad (21)$$

for all  $j \in Z_{[0, N-1]}$ .  $x(k+j|k)$  indicates a predicted future state  $x$  that is  $j$  steps away from the current time  $k$ . Due to Equation (21), at the end of the predictive horizon of length  $N$ , a terminal constraint set  $T$  needs to be considered.

A finite horizon cost function for the predictive model can typically be written as:

$$J_N(x(k), u(k)) = \sum_{j=0}^{N-1} \ell(u(k+j|k), x(k+j|k)) + F(x(k+N|k)) \quad (22)$$

where

$x(k)$  is the current state,  $\ell(\cdot, \cdot)$  is the stage cost;

$F(\cdot)$  is the terminal cost function;

$u(k) = \{u(k|k), u(k+1|k), \dots, u(k+N-1|k)\} = u(k+j|k)_{j=0}^{N-1}$  is the input sequence applied over the predictive horizon.

### 3.2. Quadratic Programming

Before predicting the trajectories of a CAV platoon, checking the feasibility of each CAV is important which ensures the optimal number of CAVs to cross an intersection. Since it could consume a lot of computing resources, quadratic programming (QP) was introduced, which has a quadratic function to be optimized subject to linear equality/inequality constraints [30].

In this study, the quadratic objective function can be written as a nonlinear programming problem with two linear functions as constraints:

$$\begin{aligned} \min f(x) &= \frac{1}{2}x^T Gx + r^T x, & x \in R^n & \quad (23) \\ \text{s.t.} & \quad A_i^T x - b_i = 0, \quad i = 1, 2, \dots, m \\ & \quad A_i^T x - b_i \geq 0, \quad i = m + 1, \dots, m + n \end{aligned}$$

where  $G \in R^{n \times n}$ ;  $A_i, r, x \in R^n$ ;  $b \in R^1$ , and  $G$  is a positive definite matrix. Thus, the programming problem is a convex quadratic programming problem in which the global minimum value is unique.

In this study, the active-set method was used to solve the quadratic programming. First, the method constructs a set of sequences to approximate the effective set  $w(x^*)$ . Starting from the initial point, it calculates the effective set  $w(x_0)$  and solves the corresponding equality constraint sub-problem. By repeating this process, it obtains the effective set sequence  $\{w(x_k)\} (k = 0, 1, \dots)$  and makes  $w(x_0) \rightarrow w(x^*)$ . As such, the optimal solution of the original problem can be obtained. Table 1 below provides the steps of active-set algorithm.

**Table 1.** The steps of active-set algorithm.

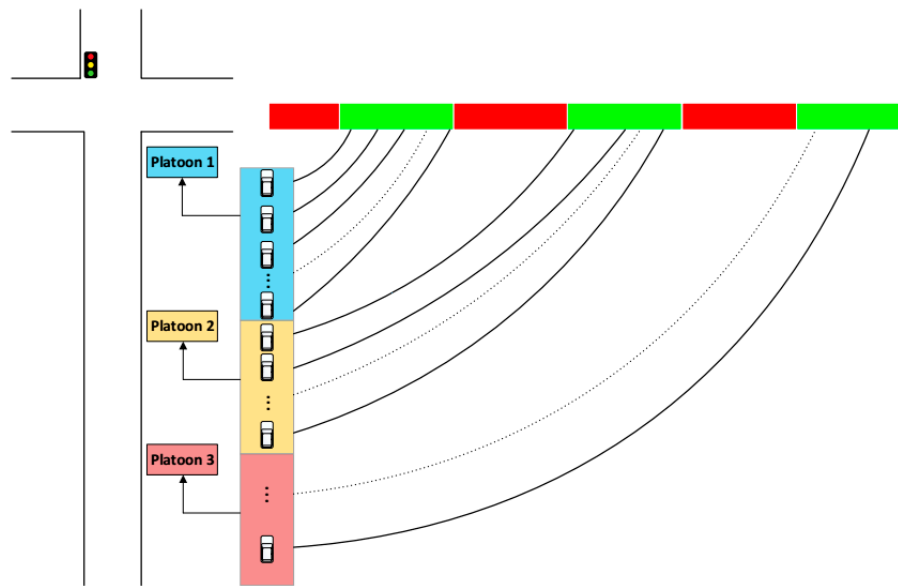
Active-Set Algorithm
1. Calculate the feasible initial point $x_0$ , let $w_0 = E \cup I(x_0), k:=0$ ;
2. Solve the equality-constrained quadratic programming problem and obtain $x_p$ , if $x_p$ is not equal to 0, go to step 3; if $\lambda_i^{(k)} \geq 0 (i \in w_k \cap I)$ ( $\lambda_i^{(k)}$ is the corresponding Lagrange multiplier), stop and get the solution $x^* = x_k$ ; otherwise, determine $i_k$ by the formulation $\lambda_{i_k}^{(k)} = \min\{\lambda_i^{(k)}   \lambda_i^{(k)} < 0, i \in w_k \cap I\}$ , then go to step 4;
3. Calculate $\alpha_k$ from formulation $\alpha_k = \min\left\{1, \min \frac{b_i - \alpha_i^T x_k}{\alpha_i^T p_k}\right\}$ , if $\alpha_k$ equal to 1, go to step 4; otherwise, find $j$ that $w_k := w_k \cup \{j\}$ and make $a_j^T (x_k + \alpha_k p_k)$
4. $w_{k+1} := w_k; k:=k+1$

## 4. Formulation of CAV Platoon Optimal Control and Solving Algorithms

### 4.1. Formulation of CAV Platoon Optimal Control

#### 4.1.1. CAV Platoon Optimal Control Strategy

In this study, the CAV platoon optimal control was implemented in a model predictive control fashion. As shown in Figure 2, when a long CAV platoon is approaching a green light, there could be a chance that it cannot cross the intersection as a single unit within this green. Thus, we dealt with this issue by considering two priorities: The first is to maximize intersection throughput given any green cycle; the second is to minimize fuel consumption and emissions. We took these priorities as the control objectives, and used the receding model predictive control (MPC) to predict and optimize CAV trajectories. Longitudinal safety was also considered as a necessary condition.



**Figure 2.** Distribute CAVs into platoons.

Due to the green time constraint, the feasibility of each vehicle passing through the intersection at a given green cycle needs to be verified. By doing that, a CAV platoon would be split into multiple sub-units in advance of approaching the intersection, in order to achieve the pre-determined two priority goals. Table 2 lists all parameters and indices used hereafter.

**Table 2.** Parameters and indices for model formulation.

Indices	Parameters
$\Delta t_s$	Sample time
$i$	The number of vehicles
$P$	Position of the vehicle
$V$	Speed of the vehicle
$g$	Green time
$a$	Accelerate of the vehicle
$V_{free}$	Free flow speed
$a_{min}$	Maximum accelerate
$L$	minimum standstill distance
$k$	The number of signal cycle
$b$	Beginning of the time
$e$	End of the time
$X$	Input control
$u$	Output control
$l$	Position of the intersection
$a_{min}$	Minimum acceleration
$H, f, Q, R$	positive definite matrix
$\tau_{safe}$	Safe following headway
$\alpha_1, \alpha_2, \alpha_3$	Parameters of objective function

#### 4.1.2. Objective Function

The objective function for CAV platoon control can be formulated into a quadratic programming problem as:

$$\min[\alpha_1(P_{n-1,t} - P_{n,t} - L - V_{n,t} * \tau_{safe})^2 + \alpha_2(V_{n-1,t} - V_{n,t})^2 + \alpha_3 a_{n,t}^2] \quad (24)$$

For the first part of the function, a following distance was defined by using the gap between two consecutive CAVs minus the multiplication of a certain safe following headway and the speed of the

following vehicle. Thus, minimizing the first part is to optimize the traffic efficiency of a CAV platoon, with a required safe headway. The second part is to optimize the traffic efficiency of a CAV platoon by minimizing the speed difference between two consecutive CAVs in the platoon. The third part is to minimize the acceleration of each following CAV, to optimize the energy efficiency of a CAV platoon.

For leading vehicles and following vehicles, we set different coefficient values. For the leading vehicles of a CAV platoon, we defined:

$$\alpha_1 = \alpha_2 = 0, \quad \alpha_3 > 0 \tag{25}$$

For following vehicles:

$$\alpha_1, \alpha_2, \alpha_3 > 0 \tag{26}$$

Thus, for a CAV platoon, the overall objective function can be written as:

$$\min f(x_t, u_t, t_t) = \sum_{t=t_0}^{T-1} (x_t^T Q x_t + u_t^T R u_t) + x_T^T Q x_T \tag{27}$$

$$\begin{aligned} \text{s.t. } & a_{\min} \leq a_t \leq a_{\max} \\ & L_i + V_{n,t} \times \tau_{\text{safe}} \leq P_{n-1,t} - P_{n,t} \\ & 0 \leq V_{n,t} \leq V_{\text{free}} \\ & l \leq P_{n,g_{k,e}} \end{aligned} \tag{28}$$

where

$$Q = \begin{bmatrix} q_1 & \dots & 0 \\ q_2 & \dots & \vdots \\ \vdots & \ddots & \vdots \\ 0 & \dots & q_2 & 0 \\ & & 0 & q_3 \end{bmatrix}, \quad Q \in R^{2n \times 2n}$$

$$T = \begin{bmatrix} \alpha_3 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \alpha_3 \end{bmatrix}, \quad T \in R^{n \times n}$$

and

$$q_1 = \begin{bmatrix} \alpha_1 & 0 & \alpha_1 & -\tau_{\text{safe}}\alpha_1 & 0 & 0 \\ 0 & \alpha_2 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$q_2 = \begin{bmatrix} -\alpha_1 0 & 2\alpha_1 & \tau_{\text{safe}}\alpha_1 & -\alpha_1 & -\tau_{\text{safe}}\alpha_1 \\ -\tau_{\text{safe}}\alpha_2 & -\alpha_2 & \tau_{\text{safe}}\alpha_1 & -\tau_{\text{safe}}^2\alpha_1 + 2\alpha_2 & 0 & -\alpha_2 \end{bmatrix}$$

$$q_3 = \begin{bmatrix} -\alpha_1 0 & \alpha_1 & \tau\alpha_1 \\ -\tau_{\text{safe}}\alpha_1 & \alpha_2 & \tau\alpha_1 & \tau_{\text{safe}}^2\alpha_1 + \alpha_2 \end{bmatrix}$$

### 4.1.3. Recursive Feasibility and Local Stability

The procession of MPC requires the solution of each predicted horizon for each sample time. Using the receding horizon MPC, the solution of first predictive horizon is only required, if the robust recursive feasibility and local stability of the receding horizon MPC can be proved. The recursive feasibility is true when the state and input of each horizon recursively belongs to the state and input of the first horizon. The local stability can hold when the cost function value of each horizon is no more than the cost function of the first horizon.



To prove recursive feasibility and local stability, the objective function can be rewritten as:

$$\min f(y_{n,t}, t) = \frac{1}{2} y^T H y + f^T y \quad (29)$$

$$\text{s.t. } y_{\min} \leq y \leq y_{\max} \quad Aeq \times y = beq \quad G \leq E y$$

where

$$j = (t_{g,e} - t_0) / \Delta t_s$$

$$H = 2 \begin{bmatrix} Q & \cdots & \cdots & 0 \\ \vdots & \ddots & \vdots & \\ \vdots & \cdots & Q & \cdots & 0 \\ 0 & \cdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & T \end{bmatrix}, \quad H \in \mathbb{R}^{(3nj+2n) \times (3nj+2n)}$$

$$f_1 = \begin{bmatrix} -2l & 0 & 0 & 2l\tau_{safe} & \cdots & 0 & 2l\tau_{safe} & 2l & 2l\tau_{safe} \end{bmatrix}^T, \quad f_1 \in \mathbb{R}^{2n}$$

$$f_2 = \begin{bmatrix} 0 & 0 & \cdots & 0 \end{bmatrix}^T, \quad f_2 \in \mathbb{R}^{nj}$$

$$f = \begin{bmatrix} f_1 & f_1 & \cdots & f_1 & f_2 \end{bmatrix}^T, \quad f \in \mathbb{R}^{3nj+2n}$$

$$y = [X_{t_0}, \cdots, X_{t_{gk,e}}, u_{t_0}, \cdots, u_{t_{gk,e-1}}]^T, \quad y \in \mathbb{R}^{3nj+2n}$$

$$y_{\max} = [yy_1, yy_2, \cdots, yy_2, a_{\max}, \cdots, a_{\max}]^T, \quad y_{\max} \in \mathbb{R}^{3nj+2n}$$

$$y_{\min} = [yy_4, \cdots, yy_4, yy_3, a_{\min}, \cdots, a_{\min}]^T, \quad y_{\min} \in \mathbb{R}^{3nj+2n}$$

$$yy_1 = \begin{bmatrix} 0 & V_{free} & \cdots & 0 & V_{free} \end{bmatrix}, \quad yy_1 \in \mathbb{R}^{1 \times 2n}$$

$$yy_2 = \begin{bmatrix} NA & V_{free} & \cdots & NA & V_{free} \end{bmatrix}, \quad yy_2 \in \mathbb{R}^{1 \times 2n}$$

$$yy_3 = \begin{bmatrix} 0 & 0 & \cdots & 0 \end{bmatrix}, \quad yy_3 \in \mathbb{R}^{1 \times 2n}$$

$$yy_4 = \begin{bmatrix} P_{1,t_0} & 0 & \cdots & P_{n,t_0} & 0 \end{bmatrix}, \quad yy_4 \in \mathbb{R}^{1 \times 2n}$$

$$A = \begin{bmatrix} 1 & \Delta t_s & \cdots & 0 \\ 0 & 1 & \cdots & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 1 & \Delta t_s \\ & & 0 & 1 \end{bmatrix}, \quad A \in \mathbb{R}^{2n \times 2n}$$

$$B = \begin{bmatrix} -1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & -1 \end{bmatrix}, \quad B \in \mathbb{R}^{2n \times 2n}$$

$$C = \begin{bmatrix} \frac{1}{2} \Delta t_s^2 & \cdots & 0 \\ \Delta t_s & \cdots & \vdots \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \frac{1}{2} \Delta t_s^2 \\ & & & \Delta t_s \end{bmatrix}, \quad C \in \mathbb{R}^{2n \times n}$$

$$\begin{aligned}
 Aeq &= \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & A & B & \cdots & 0 & C & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & A & B & 0 & \cdots & C \end{bmatrix}, \quad Aeq \in \mathbb{R}^{(2nj+2n) \times (3nj+2n)} \\
 aa &= \begin{bmatrix} 0 & 0 & \cdots & 0 \end{bmatrix}, \quad e_2 \in \mathbb{R}^{2n} \\
 bb &= \begin{bmatrix} P_{1,t_0} & V_{1,t_0} & \cdots & P_{1,t_{gk,e}} & V_{1,t_{gk,e}} \end{bmatrix}, \quad bb \in \mathbb{R}^{2n} \\
 beq &= \begin{bmatrix} bb, & aa, & \cdots & ,aa \end{bmatrix}, \quad beq \in \mathbb{R}^{2nj+2n} \\
 G &= \begin{bmatrix} 0 & 0 & \cdots & 0 \end{bmatrix}, \quad G \in \mathbb{R}^{3nj+2n} \\
 ee_1 &= \begin{bmatrix} -1 & 0 & 1 & \tau_{safe} \\ 0 & 0 & 0 & 0 \end{bmatrix} \\
 e_1 &= \begin{bmatrix} ee_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & ee_1 \end{bmatrix}, \quad e_1 \in \mathbb{R}^{(2jn+2n) \times (2jn+2n)} \\
 e_2 &= \begin{bmatrix} 0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0 \end{bmatrix}, \quad e_2 \in \mathbb{R}^{(2jn+2n) \times jn} \\
 E &= \begin{bmatrix} e_1 & e_2 \end{bmatrix}, \quad E \in \mathbb{R}^{(2jn+2n) \times (3jn+2n)}
 \end{aligned}$$

Based on the constraints and objective functions, the following propositions related to robust recursive feasibility and local stability can be obtained. The details are shown as follows.

**Proposition 1.** For a CAV platoon, the serial distributed algorithm as described in the algorithm is recursively feasible if the MPC is initially feasible.

**Proof.** If the MPC is initially feasible, there exists an optimal control sequence that  $U = [u_{t_0} \ u_{t_0+1} \ \cdots \ u_{t_{g,1,e}}]$  with the predicted trajectory  $X = [x_{t_0} \ x_{t_0+1} \ \cdots \ x_{t_{g,1,e}}]$ . Therefore, at the time interval  $t_0+1$ , the optimal control sequence is given as:  $U' = [u_{t_0+1} \ u_{t_0+2} \ \cdots \ u_{t_{g,1,e}}] \in U$  and the predicted trajectory is  $X' = [x_{t_0+1} \ x_{t_0+2} \ \cdots \ x_{t_{g,1,e}}] \in X$ . In this logic, it can be proved that the feasibility can be satisfied at all time intervals using induction.

**Proposition 2.** The CAV platoon leader is asymptotically locally stable if the MPC for platoon 1 is initially feasible.

**Proof.** At the time  $t_0$ , the cost function is:  $f(x_{t_0}, u_{t_0}, t_0) = \sum_{t=t_0}^{t_{g,e}-1} (x_t^T Q x_t + u_t^T R u_t) + x_t^T Q x_t$ . Therefore, at time interval  $t_0+1$ , the cost function is given as:  $f(x_{t_0+1}, u_{t_0+1}, t_0+1) = \sum_{t=t_0+1}^{t_{g,e}-1} (x_t^T Q x_t + u_t^T R u_t) + x_t^T Q x_t$ .  $f(x_{t_0+1}, u_{t_0+1}, t_0+1) - f(x_{t_0}, u_{t_0}, t_0) < 0$ . Hence, the CAV platoon leader is asymptotically locally stable if the MPC for platoon 1 is initially feasible.

#### 4.2. Sub-Platoon Splitting Algorithms

As the methodologies described above, we can summarize the control algorithm for sub-platoon splitting algorithm. The process will stop when all vehicles in the initial platoon have been planned. Table 3 presents the algorithm.

**Table 3.** Sub-platoon splitting algorithm.

<b>Algorithm:</b>	
1.	Get the initial condition: The number of CAVs: $M$ ; the signal timing; the position, and position of all CAVs;
2.	Set the $k$ th signal timing: $k = 1$ , the $i$ th platoon: $i = 1$ ;
3.	Set the number of CAVs in platoon $i$ : $N$ ; the beginning and the end of the $k$ th signal green timing: $g_{k,b}$ and $g_{k,e}$ ;
4.	Set the horizon of platoon $i$ ;
5.	Check the feasibility of constraints, if yes, go to step 6, otherwise, go to step 7;
6.	$M = M - N$ , $I = I + 1$ , $k = k + 1$ , go to step 8;
7.	$N = N - 1$ ; go to step 4;
8.	Judge whether $M = 0$ , if yes, end; otherwise, go to step 3.

## 5. Experiment and Results

To evaluate the performance of the proposed model, a simulation experiment was conducted due to the high cost of field testing. Particularly, the proposed model was compared to a decentralized control model, in terms of objective function values, optimal trajectory outputs, control delay, and emissions. Note that the decentralized method was developed based on the same concept of the proposed method, while it optimized each CAV instead. Table 4 lists the simulation parameters used in the experiments.

**Table 4.** Attributes in the simulation.

Parameters	Value
Maximum acceleration ( $\text{m/s}^2$ )	3.5
Minimum acceleration ( $\text{m/s}^2$ )	-4
Maximum speed (m/s)	0
Minimum speed (m/s)	21
Free flow speed (m/s)	21
Duration of red light (s)	30
Duration of green light (s)	30
Number of CAV	100
Time step (s)	1
Factor $\alpha_1$	1
Factor $\alpha_2$	1
Factor $\alpha_3$	1
Safety distance $L$ (m)	3

In our experiments, four target safe headways for the CAV platoon control were examined, including: 1, 0.8, 0.6, and 0.4 s. To note, this target safe headway is equivalent to the safety headway in the objective function (i.e.,  $\tau_{safe}$ ). That is, each CAV in a platoon is expected to follow its leader with a headway that is close to but no less than the target safe headway. As such, the CAV platoon can be expected to maximize its efficiency with a pre-determined safety headway.

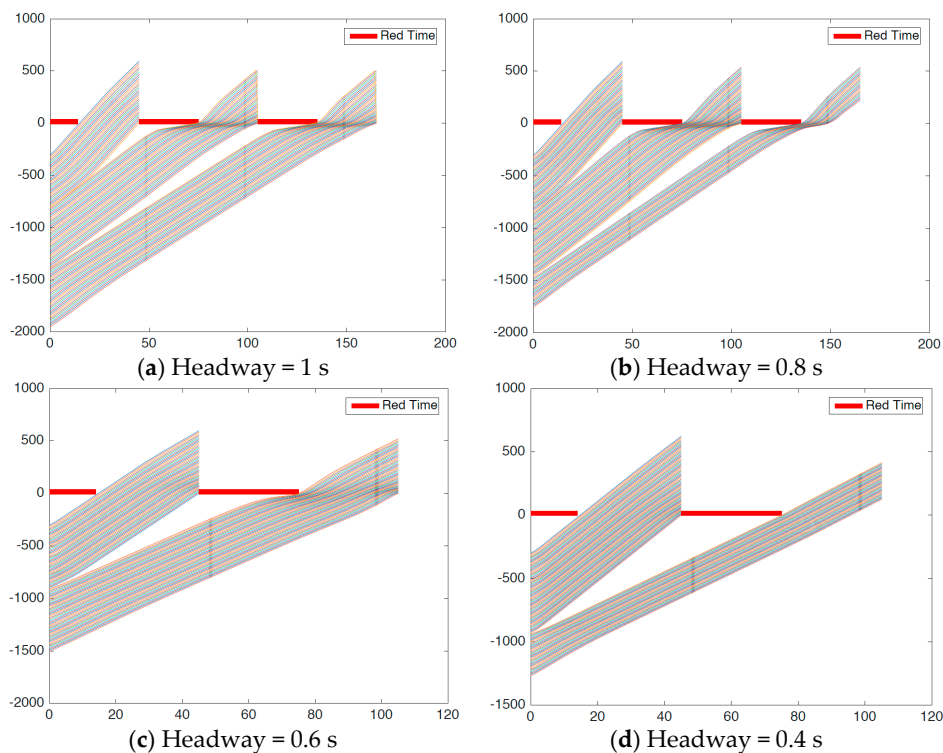
Two control algorithms were applied for the CAV platoon control, one is the proposed centralized algorithm and the other is decentralized algorithm. Regarding the decentralized algorithm, the problem formulation (i.e., objective function and constrains) and solving methods (i.e., MPC and platoon split algorithm) were the same as the centralized algorithm. However, when checking the feasibility, quadratic programming was applied on each CAV instead of the whole platoon. As such, if a CAV was determined to be unable to cross the intersection within the current green time, it was treated as the leading vehicle of the next platoon.

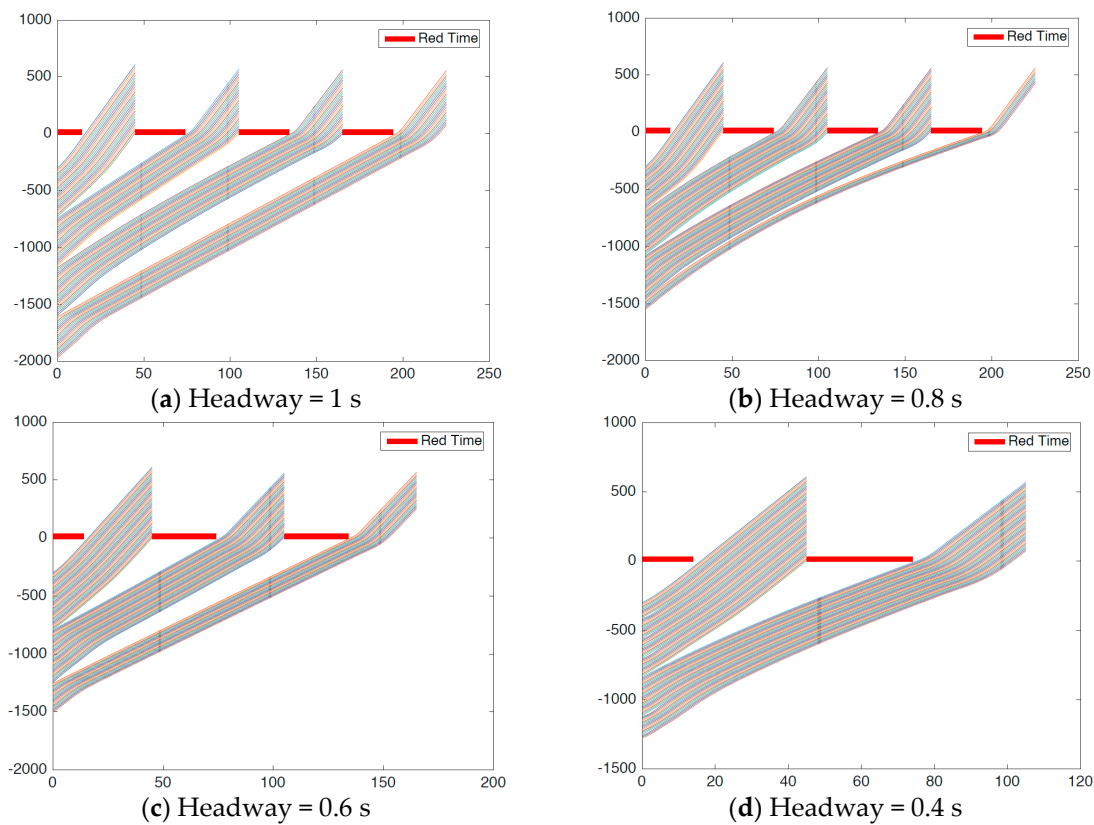
According to the simulation results, the proposed centralized algorithm generally outperformed the decentralized algorithm, in terms of resulting in lower objective function values for all target safe headways. The objective function values and CAV platoon splits for the two algorithms are shown in Table 5.

**Table 5.** Objective function value and CAV platoon splits for the proposed control algorithm under different initial following headways.

Headway		1	0.8	0.6	0.4
Centralized	Objectives	6101.4	5341.4	4469.5	3922.6
	CAV sub-platoons	Platoon 1:31 Platoon 2:35 Platoon 3:34	Platoon 1:38 Platoon 2:42 Platoon 3:20	Platoon 1:50 Platoon 2:50	Platoon 1:65 Platoon 2:35
	Objectives	9433.6	6244.5	5893.0	4563.3
Decentralized	CAV sub-platoons	Platoon 1:26 Platoon 2:26 Platoon 3:26 Platoon 4:22	Platoon 1:31 Platoon 2:31 Platoon 3:30 Platoon 4: 8	Platoon 1:40 Platoon 2:38 Platoon 2:22	Platoon 1:54 Platoon 2:46

As for the CAV platoon split, the centralized algorithm resulted in fewer sub-platoons than the decentralized algorithm did, under certain target safe headways (i.e., 1, 0.8, and 0.6 s). The optimal control results for the two control algorithms are shown in Figures 3 and 4, respectively. To note, when the target safe headway is relatively large (e.g., 1 and 0.8 s), there is an obvious deceleration for CAVs under the centralized control near the intersection. The underlying reason could be that when the target headway is relatively large, vehicles have to decelerate and keep a relatively low speed to cross the intersection. In doing so, a tighter formation of the CAV platoon could be formed. Otherwise, if all CAVs keep a high speed for a large safe headway, the distance between two consecutive CAVs would be very large. As such, fewer CAVs could pass through the intersection within one green cycle. This phenomenon was not observed for the decentralized cases. Some may argue that a considerable large deceleration could cause uncomfortableness of CAV passengers. In this study, the maximum deceleration rate was carefully set to  $4 \text{ m/s}^2$  in order to prevent extreme cases. However, a trade-off between comfortableness and safe headway could be further explored.

**Figure 3.** Optimal CAV trajectories for the proposed control algorithm under different initial following headways.



**Figure 4.** Optimal CAV trajectories for the decentralized control algorithm under different initial following headways.

Control delay was used to compare the performance of the two algorithms in terms of traffic efficiency. Control delay of a signalized intersection is generally defined as the delay caused by the traffic signal operation, which includes acceleration/deceleration delay and stopped delay [31]. It can be calculated as [32]:

$$C_d = \frac{\sum_i ta_i - tv_i}{n} \tag{30}$$

where

$C_d$  is the average control delay;

$ta_i$  is the time stamp that a CAV crosses the intersection;

$tv_i$  is the time stamp that a CAV is supposed to cross the intersection under free flow speed.

Table 6 shows the average control delay of all CAVs in the platoon, based on the proposed centralized and the decentralized algorithm. The delay time for the centralized algorithm is much less than that for the decentralized algorithm, by allowing more vehicles to pass through the intersection within one green cycle.

**Table 6.** Average delay for two control algorithms.

Control Delay (s)	Headway = 1 s	Headway = 0.8 s	Headway = 0.6 s	Headway = 0.4 s
Centralized	41.8739	39.2108	19.3483	16.9789
Decentralized	63.2619	55.3255	35.8247	20.0893

According to Figures 3 and 4, the decentralized method seemed to be more energy efficient, since the centralized method resulted in CAV platoons with relatively low speed near intersections.

Thus, to explore the energy efficiency of the two methods, the emission was calculated by the fuel consumption and emissions model developed by Akcelik is adopted [33]:

$$e_{t,n}(t) = \sum_{j=0}^3 \rho_j x_{2,n}(t)^j + \sigma_1 x_{2,n}(t) a_n(t) + \sigma_2 x_{2,n}(t) a_n(t)^2 \varphi(v_n(t)) \quad (31)$$

where

$x_{2,n}(t)$  is the speed difference between vehicle  $n$  and vehicle  $n-1$ ;

$a_n(t)$  is the acceleration of vehicle  $n$  at time  $t$ ;

$\rho_j, \sigma_1, \sigma_2$  are the parameters of fuel consumption and emissions model;

$\varphi(v_n(t))$  is a Heaviside function of acceleration:

$$\varphi(v_n(t)) = \begin{cases} 1 & a_n(t) \geq 0 \\ 0 & a_n(t) < 0 \end{cases}$$

According to Table 7, the centralized method outperformed the decentralized method. The primary reason could be that the centralized method allowed most vehicles to cross the intersection in a shorter time. As such, the overall emission of the centralized method was still lower than that of the decentralized method.

**Table 7.** Emissions for centralized and decentralized control algorithm.

	Headway = 1 s	Headway = 0.8 s	Headway = 0.6 s	Headway = 0.4 s
Centralized	10768.30	5808.05	3289.96	3356.82
Decentralized	26,704.30	24,834.03	30,009.98	33,161.46

From the mathematical perspective, the proposed centralized control algorithm focused on the whole platoon optimization, while the decentralized control algorithm attempted to optimize the trajectory of every single CAV. Namely, the proposed centralized control algorithm was calculating the minimization of summations, while the decentralized control algorithm was calculating the summation of minimization. Mathematically, the minimization of summations is smaller or equal to the summation of minimization. Thus, the results could be considered as reasonable and consistent.

## 6. Conclusions and Future Directions

This paper proposed a multi-objective guaranteed feasible CAV platoon trajectory control method with priorities for isolated signalized intersections. Specifically, we prioritized the intersection throughput and traffic efficiency under a pre-defined signal cycle, based on which we minimized fuel consumption and emissions for the CAV platoons. To this end, a vehicular sub-platoon splitting algorithm based on Farkas lemma was introduced to accommodate a maximum number of vehicles for each signal green time phase. Moreover, the optimal control problem was formulated as discrete linear quadratic control problems and was solved by quadratic programming efficiently. For rigor, the recursive feasibility and asymptotic stability of the proposed predictive control model was proved.

The performance of the control algorithm was compared against a non-cooperative distributed CAV control through simulation. The proposed centralized algorithm outperformed the decentralized algorithm, in terms of providing lower objective function values, smaller average control delay, and lower emission. When a relatively large target safe headway was required for the CAV platoon control, the centralized algorithm tended to drive vehicles through the intersection under a relatively low speed. As such, a tighter CAV platoon can be formed so that more vehicles can cross the intersection within one cycle. In general, the research provides an effective and efficient control method to enhance

both traffic throughput and energy efficiency with ensured longitudinal safety for long CAV platoons at signalized intersections.

This research also has some limitations. First, some more complicated scenarios could be considered, such as lane-change behaviors. Second, the development of controllers can be extended to mixed vehicle platoons consisting of both CAVs and human-driven vehicles. Third, the combined use of signal control and CAV trajectory optimization may be more effective to improve intersection performance. Last but not the least, human factors (both on board and outside) need to be carefully considered if the proposed method will be tested in the real-world. For instance, the proposed centralized method could result in considerable decelerations of CAVs at intersections, when the target safe headway is relatively large. As such, passengers could feel uncomfortable. The authors recommend that future studies could focus on these topics.

**Author Contributions:** Conceptualization, C.W. and J.X.; methodology, C.W. and Y.D.; software, Y.D.; validation, C.W. and Y.D.; resources, C.W.; data curation, C.W. and Y.D.; writing—original draft preparation, C.W. and Y.D.; writing—review and editing, C.W. and J.X. All authors have read and agree to the published version of the manuscript.

**Funding:** This research has supported by the National Natural Science Foundation of China (71971061) and the National Key R&D Program of China (2018YFB1601003).

**Acknowledgments:** This research has supported by the National Natural Science Foundation of China (71971061).

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Guo, Y.; Ma, J.; Xiong, C.F.; Li, X.; Zhou, F.; Hao, W. Joint optimization of vehicle trajectories and intersection controllers with connected automated vehicles: Combined dynamic programming and shooting heuristic approach. *Transp. Res. Part C Emerg. Technol.* **2019**, *98*, 54–72. [[CrossRef](#)]
2. Lokhandwala, M.; Cai, H. Dynamic ride sharing using traditional taxis and shared autonomous taxis: A case study of NYC. *Transp. Res. Part C Emerg. Technol.* **2018**, *97*, 45–60. [[CrossRef](#)]
3. Fagnant, D.J.; Kockelman, K. Preparing a nation for autonomous vehicles: Opportunities, barriers and policy recommendations. *Transp. Res. Part A Policy Pract.* **2015**, *77*, 167–181. [[CrossRef](#)]
4. Wang, M.; Daamen, W.; Hoogendoorn, S.P.; Arem, B.V. Rolling horizon control framework for driver assistance systems. Part I: Mathematical formulation and non-cooperative systems. *Transp. Res. Part* **2014**, *40*, 271–289. [[CrossRef](#)]
5. Wang, M.; Daamen, W.; Hoogendoorn, S.P.; Arem, B.V. Rolling horizon control framework for driver assistance systems. Part II: Mathematical formulation and non-cooperative systems. *Transp. Res. Part C* **2014**, *40*, 290–311. [[CrossRef](#)]
6. Li, X.P.; Yanfeng, O.Y. Characterization of Traffic Oscillation Propagation under Nonlinear Car-Following Laws. *Transp. Res. Part B Methodol.* **2015**, *45*, 1346–1361. [[CrossRef](#)]
7. Zhou, Y.; Ahn, S.; Chitturi, M.; Noyce, D.A. Rolling horizon stochastic optimal control strategy for ACC and CACC under uncertainty. *Transp. Res. Part C Emerg. Technol.* **2017**, *83*, 61–76. [[CrossRef](#)]
8. Zhou, Y.; Ahn, S.; Wang, M.; Hoogendoorn, S. Stabilizing mixed vehicular platoons with connected automated vehicles: An H-infinity approach. *Transp. Res. Part B Methodol.* **2019**. [[CrossRef](#)]
9. Li, Y.; Wang, H.; Wang, W.; Lu, X.; Shanwen, L. Evaluating the safety impact of adaptive cruise control in traffic oscillations on freeways. *Accid. Anal. Prev.* **2017**, *104*, 137–145. [[CrossRef](#)]
10. Li, Y.; Li, Z.B.; Wang, H.; Wang, W.; Lu, X. Evaluation of the impacts of cooperative adaptive cruise control on reducing rear-end collision risks on freeways. *Accid. Anal. Prev.* **2017**, *98*, 87–95. [[CrossRef](#)]
11. Li, L.; Chen, X.Q.; Zhang, L. A global optimization algorithm for trajectory data based car-following model calibration. *Transp. Res. Part C Emerg. Technol.* **2016**, *68*, 311–332. [[CrossRef](#)]
12. Li, X.P.; Cui, J.; An, S.; Parsafard, M. Stop-and-go traffic analysis: Theoretical properties, environmental impacts and oscillation mitigation. *Transp. Res. Part B* **2014**, *70*, 319–339. [[CrossRef](#)]
13. Treiber, M.; Kesting, A. *Traffic Flow Dynamics: Data, Models and Simulation*; Springer: Berlin/Heidelberg, Germany, 2013.

14. Hao, S.; Yang, L.; Shi, Y. Data-Driven Car-Following Model Based on Rough Set Theory. *IET Intell. Transp. Syst.* **2018**, *12*, 49–57. [[CrossRef](#)]
15. Wang, X.; Jiang, R.; Li, L.; Lin, Y.; Zheng, X.; Wang, F.Y. Capturing car-following behaviors by deep learning. *IEEE Trans. Intell. Transp. Syst.* **2018**, *19*, 910–920. [[CrossRef](#)]
16. Cassidy, M. Increasing the capacity of an isolated merge by metering its on-ramp. *Transp. Res. Part B* **2005**, *39*, 896–913. [[CrossRef](#)]
17. Liu, P.; Li, Z.; Xu, C.; Duan, H.; Wang, W. Reinforcement Learning-Based Variable Speed Limits Control to Reduce Crash Risks near Traffic Oscillations on Freeways. *IET Intell. Transp. Syst.* **2019**, *18*, 3204–3217. [[CrossRef](#)]
18. Xu, C.C.; Wang, Y.; Liu, P.; Wang, W.; Bao, J. Quantitative risk assessment of freeway crash casualty using high-resolution traffic data. *Reliab. Eng. Syst. Saf.* **2018**, *169*, 299–311. [[CrossRef](#)]
19. Huang, X.; Sun, J.; Sun, J. A car-following model considering asymmetric driving behavior based on long short-term memory neural networks. *Transp. Res. Part C Emerg. Technol.* **2018**, *95*, 346–362. [[CrossRef](#)]
20. Feng, Y.; Yu, C.; Liu, H. Spatiotemporal intersection control in a connected and automated vehicle environment. *Transp. Res. C Emerg. Technol.* **2018**, *89*, 364–383. [[CrossRef](#)]
21. Zhou, F.; Li, X.; Ma, J. Parsimonious shooting heuristic for trajectory design of connected automated traffic part I: Theoretical analysis with generalized time geography. *Transp. Res. Part B Methodol.* **2017**, *95*, 394–420. [[CrossRef](#)]
22. Ma, J.; Li, X.P.; Zhou, F.; Hu, J.; Park, B. Parsimonious shooting heuristic for trajectory design of connected automated traffic part II: Computational issues and optimization. *Transp. Res. B Methodol.* **2017**, *95*, 421–441. [[CrossRef](#)]
23. Guo, Y.; Liu, P.; Wu, Y.; Chen, J. Evaluating How Right-turn Treatments Affect Right-Turn-on-Red Conflicts at Signalized Intersections. *J. Transp. Saf. Secur.* **2018**. [[CrossRef](#)]
24. Lee, J.; Park, B. Development and evaluation of a cooperative vehicle intersection control algorithm under the connected vehicles environment. *IEEE Trans. Intell. Transp.* **2012**, *13*, 81–90. [[CrossRef](#)]
25. Wang, C.; Xu, C.C.; Xia, J.X.; Qian, Z.D. A combined use of microscopic traffic simulation and extreme value methods for traffic safety evaluation. *Transp. Res. Part C Emerg. Technol.* **2018**, *90*, 281–291. [[CrossRef](#)]
26. Wang, C.; Xu, C.C.; Dai, Y. A crash prediction method based on bivariate extreme value theory and video-based vehicle trajectory data. *Accid. Anal. Prev.* **2018**, *123*, 365–373. [[CrossRef](#)]
27. Montanino, M.; Punzo, V. Trajectory data reconstruction and simulation-based validation against macroscopic traffic patterns. *Transp. Res. Part B Methodol.* **2015**, *80*, 82–106. [[CrossRef](#)]
28. Raffel, C.; Ellis, D.P.W. Feed-Forward Networks with Attention Can Solve Some Long-Term Memory Problems. *arXiv* **2015**. Available online: <https://arxiv.org/abs/1512.08756> (accessed on 2 October 2019).
29. Liu, P.; Wu, J.; Qin, X.; Zhou, H.; Bao, J.; Yang, Z. Estimating Queue Length for Contraflow Left-turn Lane Design at Signalized Intersections. *J. Transp. Eng.* **2019**, *145*, 04019020. [[CrossRef](#)]
30. Liu, Y.; Liu, Z.; Jia, R. DeepPF: A Deep Learning Based Prediction Architecture for Metro Passenger Flow. *Transp. Res. Part C Emerg. Technol.* **2019**, *101*, 18–34. [[CrossRef](#)]
31. Transportation Research Board. *Highway Capacity Manual*; Transportation Research Board: Washington, DC, USA, 2000.
32. Ko, J.; Hunter, M.; Guensler, R. Measuring Control Delay Components Using Second-by-Second GPS Speed Data. *J. Transp. Eng.* **2008**, *134*, 338–346. [[CrossRef](#)]
33. Akcelik, R. Efficiency and drag in the power-based model of fuel consumption. *Transport. Res. Part B Methodol.* **1989**, *23*, 376–385. [[CrossRef](#)]

