V2X-Based Decentralized Cooperative Adaptive Cruise Control in the Vicinity of Intersections

Bing Liu and Abdelkader El Kamel, *Senior Member, IEEE*

***Abstract*—Cooperative driving with V2X communication is widely researched due to its considerable potential to improve the safety and efficiency of road transportation systems. In this paper, a decentralized cooperative adaptive cruise control algorithm using V2X for vehicles in the vicinity of intersections (CACC-VI) is proposed. This algorithm is designed to improve the throughput of intersection by reorganizing the vehicle platoons around it, in consideration of safety, fuel consumption, speed limit, heterogeneous features of vehicles, and passenger comfort. Within a platoon, vehicles try to find the optimal control input by a distributed particle swarm optimization (PSO) algorithm, in order to reduce tracking errors, while respecting different constraints. A concept of opportunity space is proposed to facilitate platoon reorganization, in which a subplatoon or an individual vehicle can choose to accelerate to join in the preceding platoon or to decelerate to depart from the current one. The main idea is to make full use of the road capacity and to distribute it to most vehicles that are capable to find an accelerating trajectory to get through the intersection within a limited period. The originality of our algorithm is the introduction of a novel application of V2X communication to make the traffic more intelligent, in terms of safety, time saving, and environment friendly.**

***Index Terms*—V2X communication, cooperative adaptive cruise control (CACC), PSO with constraints, eco-driving.**

I. Introduction

ARRIVING at one place safely in a certain period is the basic requirement of transportation. However, today's road transportation is far from perfect. Incorrect driving behaviors like drunken driving, fatigue driving and speeding are thought to be the main reasons for road accidents which on one hand cause injury, death, and property damage, on the other hand make vehicles keep larger distance from each other, thus the road capacity is not made full use of. Moreover, congestions caused by incorrect driving behaviors, accidents, improper signal timing have become a global phenomenon which has economically and ecologically negative effects, people have to spend more time on road and more fuel is consumed, which leads to more pollution.

More efficient, better space utilization and elimination of human error, self-driving or semi self-driving car developed by Google and automobile manufacturers all over the world is a potentially revolutionizing technology to solve these problems [1]. However, the intelligence of individual vehicles does not represent the intelligence of the whole transportation system.

V2X technology, including Vehicle to Vehicle (V2V) communication via wireless ad hoc network and Vehicle to Infrastructure (V2I) via Dedicated Short-Range Communication (DSRC), is considered to be the next step to construct the intelligent transportation system (ITS). Instead of detecting the environment all by the vehicle itself, this technology enables the communication between vehicles themselves and with infrastructure along the road. Possible applications of V2X in terms of safety, information dissemination and efficiency are illustrated in [2].

For instance, the cooperative collision warning systems proposed in [3] provides warnings or situation awareness displays to drivers based on information about the motions of surrounding vehicles obtained via V2V communications from those vehicles, without use of any ranging sensors. In [4], a traffic congestion recognition and avoidance algorithm is proposed which utilizes V2X technology to share information about the current local traffic situation and to use this information to optimize the routes. A velocity planning algorithm is designed in [5] based on traffic signal information obtained by V2I communication, the idea is to minimize the acceleration/deceleration rates while ensuring the vehicle get through the intersection without coming to a full stop. Researches shows that vehicle fuel consumption and emissions can be reduced by avoiding sharp acceleration/deceleration patterns and decreasing idling period [5]–[7]. Similarly, different optimal speed advising algorithms based on V2I have been proposed by other researchers [8]–[10]. In these algorithms, the maneuver responsibility is left to the driver, and no direct inter-vehicle communication is considered.

Furthermore, V2X is also the key technology to make some current systems more intelligent. For example, the adaptive cruise control (ACC) system, which is commercially available in the market, aims at automatically maintaining a constant speed or a safe distance to the preceding vehicle. Currently, the ACC system is mainly based on ranging sensors, e.g., radars and lasers, which are capable to measure the range or its changing rate. Generally, ACC system can relieve drivers from performing repetitive and boring jobs like adjusting speed and distance to the preceding vehicle in highway scenarios. However, current ACC systems still have disadvantages, like incapability to more complicated urban scenarios and limit anticipatory capabilities to sudden maneuver of the preceding vehicle [11]. Moreover, ACC system cannot deal with

Manuscript received December 15, 2014; revised June 14, 2015; accepted September 20, 2015. Date of publication November 4, 2015; date of current version February 26, 2016. The Associate Editor for this paper was L. Li.

The authors are with the Centre de Recherche en Informatique, Signal et Automatique de Lille (CRIStAL), École Centrale de Lille, 59651 Lille, France (e-mail: bing.liu@ec-lille.fr; abdelkader.elkamel@ec-lille.fr).

Digital Object Identifier 10.1109/TITS.2015.2486140

shock-wave effect. Due to delays in sensor and actuator system of the ACC host vehicle, amplifications may appear, especially when constant spacing policy is used. Thus passenger comfort cannot be guaranteed and more fuel is consumed.

With V2X technology, ACC can be extended into cooperative ACC (CACC), which is considered to be highly potential to improve traffic flow capacity and smoothness and to reduce congestions [12]. V2X integrated vehicle can not only get information from the preceding vehicle as in ACC system, but also the vehicles before the preceding one, even beyond line of sight [13]. Research shows that with CACC, vehicles can have smaller spacing distance compared to ACC, and the shock-wave effect can be greatly mitigated [14], [15]. A safety spacing policy (SSP) is introduced in [16] which utilizes the state of the preceding vehicle combined with the braking capacity to adjust the position and velocity of the controlled vehicle. The SSP can ensure vehicle string stability and it can also yield stable traffic flow and higher traffic capacity than the traditional Constant Time Gap (CTG) policy.

String stability is an important goal to be achieved in CACC system design [13]. A platoon of vehicles is called string stable only if disturbances propagated from the leading vehicle to the rest of the platoon can be attenuated [17]. Researchers have tried different methodologies in designing platoon controllers. Most of the projects are relied on classic control theory like proportional-derivative feedback/feedforward controllers [13], [18]–[20]. The authors in [21] summarize the block diagram to design a CACC system using classic control theory and suggest the four techniques to be used in the system. The concepts of "expected velocity" and "expected acceleration" are introduced in [22] and the CACC problem is turned into a tracking problem of spacing error, velocity and acceleration error in the platoon, where the control input of each vehicle is calculated by using an optimal control method. Model predictive control (MPC) technique can also be found in CACC research and implementation [23], [24]. A multi-objective CACC controller is designed in [25], which takes the tracking capacity, fuel economy and driver desired response into account.

By using V2X communication, revolutionary improvements can also be achieved in traffic control systems. A survey of different design philosophy for traffic control at intersections using V2X communication is given in [26]. Generally, there are two ways to improve the current systems. One way is platoon-based control strategy [27]–[31], in which time plan of traffic lights is set referring to the information of platoons comings from different directions so that they can get through the intersection without being interrupted. If V2X device is integrated in the platoon [31], the information can be sent directly to the intersection manager instead of identifying the platoon by inductive loops or other sensors.

Another way is cooperative driving, in which the intersection controller receives the vehicles' requests to get through the intersection and gives each vehicle the advice of an optimal course in consideration of their dynamic capacity of action ensuring no crashes occur, at the same time minimizing the intersection delay [32]–[35]. In cooperative driving, a platoon becomes flexible, a vehicle can join in or depart from an already formed platoon; and several vehicles can also choose to form a new one [32]. There are two different ways to decide a feasible schedule: negotiation-based control [33] and planning-based control [32], [34]. The former consumes less computation costs because only the a few vehicles nearby the intersection are taken into account, whereas the latter sometimes can achieve higher performance. Other methods to find a feasible schedule are also investigated, like game theory [36] and heuristic algorithms [37].

In this paper, we propose a CACC algorithm based on V2X technology which can be used in the vicinity of intersections. The main idea is to reorganize the already existed platoons according to the traffic signal timing and the heterogeneous characteristics and capability of vehicles to take advantage of the unused road capacity in order to improve the throughput of intersection. Meanwhile, many other features and limits are also taken into account, such as safety, fuel consumption, speed limit, passenger comfort, etc. Each vehicle in the platoon can obtain the position, velocity, and acceleration information from the platoon leader and the preceding vehicle. A particular swarm optimization (PSO) algorithm with constraints is used to find the optimal inputs at every step to eliminate the tracking errors. When approaching the intersection, an "opportunity space" is calculated according to the traffic signal timing and platoons' future positions. An optimal trajectory is calculated for each platoon leader in consideration of fuel economy within the constrained domain. A space arrangement approach is designed in order to maximize the number of vehicles which could get through the intersection during the current green phase and minimize the distance to be covered for each vehicle. If the opportunity space is larger than a platoon's length and the all the platoon members are capable to produce the desired trajectory, this platoon can get through the intersection within the limited period by accelerating. If not, the platoon splits into two parts, the first one gets through the intersection by accelerating, while the second one decelerates to a lower speed to get through when the lights turns green. If the opportunity space has not been fully distributed, a vehicle can choose to join in the preceding platoon to take advantage of this space after verifying its own capability.

The rest of the paper is organized as follows. In Section II, the scenario of our research is described, the control objectives are established, and the vehicle kinematic model and the platoon model that we use in the algorithm design are introduced. In Section III, a platoon control algorithm using constrained PSO is given, and a platoon reorganization method, including our space arrangement approach and trajectory planning algorithm, is introduced in details. In Section IV, simulation results are presented to prove the validation of our algorithm. Finally, some concluding remarks and perspectives are given in Section V.

II. Problem Statement

A. Scenario Description and Objectives

The object of our study is the group of vehicles approaching an intersection controlled by a signal timing plan which is produced by the intersection manager using some kind of superior signal control algorithm to coordinate the vehicles

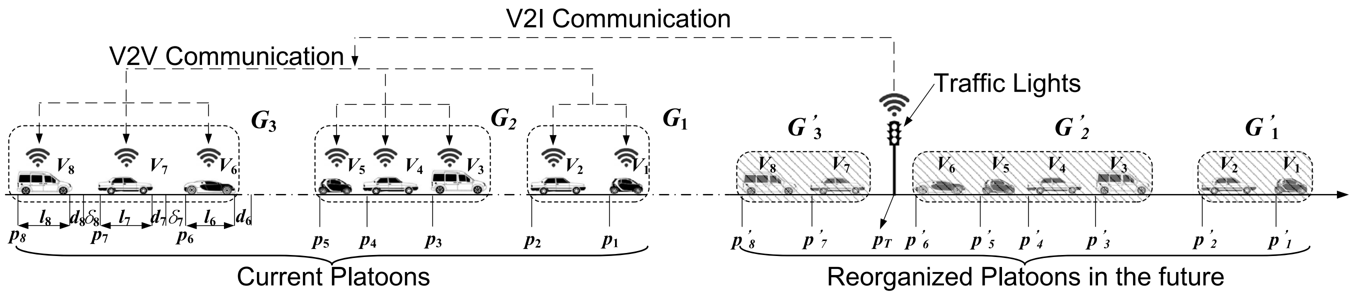


Fig. 1. Scenario illustration.

coming from different directions. The traffic signal control systems have been widely investigated [27], [29], [38] and will not be covered in the scoop of this paper. In our research, the intersection manager does not need to plan the trajectory of each vehicle because its computational capability is usually limited, it produces only near optimal schedule including the "safe time window" and the "danger time window" to permit or prohibit the vehicles in a certain lane to pass. In order to facilitate the expression, we consider the "safe time window" as a "green light," while the "danger time window" as a "red light." The plan can be expressed directly by physical traffic lights or be sent to vehicles via V2I communication.

Each vehicle is supposed to be equipped with V2V device which is capable to send/receive the information such as position, velocity and acceleration to/from the corresponding vehicle. The intersection manager can broadcast the signal timing to the vehicles in the corresponding lane. We assume that lane-shifting maneuver has been completed before arriving at the effect area of our algorithm, which is defined as the area can be covered by the V2I communication. So all the lanes with green light can be considered as independent. Therefore, we can extract a single lane from the intersection as our study object in which the vehicles' trajectories should be planned according to the signal time plan already designed by the intersection manager.

To briefly explain the idea of our method, we take the scenario in Fig. 1 as an instance. All the vehicles circulate from the left to the right. The vehicles which have passed the intersection are not displayed in the figure and they are not considered in our algorithm. Platoons have been formed already when entering the effect area of our algorithm. Instead of a homogeneous vehicle group in which all the vehicles are of the same kind, a heterogeneous group is discussed in this paper where vehicles can be different in type, size, engine power, and so on. Besides, the difference in driver settings can also lead to the heterogeneity of vehicles.

In Fig. 1, there are three platoons formed by eight vehicles approaching the intersection in the same lane. The ith vehicle is denoted as V_i, while the ith platoon is denoted as G_i, and the traffic lights are installed at p_T which indicates the position of intersection entrance. The traffic signal timing is designed by the intersection manager and is broadcasted to the vehicles. The first platoon G_1 can get through the intersection during the current green light by maintaining its current speed, while the rest platoons G_2 and G_3 cannot. A ameliorated solution can be found by our algorithm: the vehicle V_6 can choose to join in the platoon G_2, and the new platoon G'_2 accelerates to pass during the limited period. At the same time, the new platoon G'_3 decelerates to a smaller velocity so that it can get through the intersection when the light turns green again while full stop is avoided.

According to the vehicles' behaviors in this scenario, the design objectives of our algorithm are established as followings:

- Instead of a centralized algorithm, a distributed one should be proposed in order to reduce the computational cost at the intersection manager by distributing the cost to vehicles;
- In consideration of the heterogeneity of vehicles, it is necessary to design an algorithm, in which each vehicle must make decisions individually based on its own features, the driver's desired response and real-time traffic information from other vehicles and the intersection manager;
- The number of vehicles which can get through the intersection during the current green phase should be maximized;
- The platoons should be flexible, a vehicle can choose to join in a platoon or depart from it;
- Trajectories need to be planned for vehicles in different cases, meanwhile their fuel consumption should be minimized;
- Constraints like speed limit and power limit of the vehicle must be respected;
- The safety of the whole process must be guaranteed, thus a safety space must be maintained to avoid rear-end collision;
- The velocity of a platoon should be the same to its preceding platoon to prevent collision after getting through the intersection;
- The passengers' comfort should be taken into account, in other words, sharp change of the acceleration should be averted.

B. Vehicle Kinematic Model

As we assume that no lane shifting maneuver is taken during the process, a longitudinal vehicle model is sufficient. Hence, a simple three-state space linear model is used [22]. For the ith vehicle V_i, we have:

$$\dot{x}_i(t) = \Phi_i x_i(t) + \Pi_i u_i(t)$$
$$x_i(t) = \begin{bmatrix} p_i(t) & v_i(t) & a_i(t) \end{bmatrix}^T$$
$$\Phi_i = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -1/\tau_i \end{bmatrix}, \quad \Pi_i = \begin{bmatrix} 0 \\ 0 \\ 1/\tau_i \end{bmatrix} \tag{1}$$

where $x_i(t) \in R^3$ is the ith vehicle's system state at time t, and $p_i(t)$, $v_i(t)$ and $a_i(t)$ are separately the position, velocity and acceleration of the vehicle; $u_i(t) \in R$ represents the control input like thrust and brake; τ_i is the time constant coming from the vehicle's dynamical system which is heterogeneous for different vehicles. The continuous-time system above should be turned into discrete-time domain considering the implementation issue.

$$x_i(k+1) = A_i x_i(k) + B_i u_i(k). \tag{2}$$

Moreover, constraints should be added on the vehicle system,

$$v_i(k) \le v_{\text{limit}} \tag{3}$$
$$u_{\min,i} \le u_i(k) \le u_{\max,i} \tag{4}$$
$$|\Delta a_i(k)| \le |\Delta a|_{\max,i} \tag{5}$$
$$\frac{P_{\text{tractive},i}(k)}{\eta_i} \le P_{\text{engine},i}$$
$$P_{\text{tractive},i}(k) = \frac{M_i}{1000} \cdot v_i(k) \cdot a_i(k) + \left(M_i \cdot g \cdot C_{r,i} + \frac{\rho}{2} \cdot v_i^2(k) \cdot A_i \cdot C_{a,i}\right) \cdot \frac{v_i(k)}{1000}. \tag{6}$$

Equation (3) represents the road speed limit. Equation (6) is the power constraint of the ith vehicle, and it means that the tractive power demanded by the vehicle cannot exceed its engine power, where η_i is the transmission efficiency of the vehicle. The tractive power is composed of two parts, the first part comes from the acceleration of the vehicle, while the second part comes from the rolling resistance and the aerial resistance, where g is the gravitational constant, C_r is the rolling resistance coefficient, ρ is the mass density of air, A_i is the vehicle cross sectional area, and C_a is the aerodynamic drag coefficient. In consideration of reality, we add equation (4) as the constraint on the control input because the thrust and the brake ability are usually limited. As the passenger's comfort should be guaranteed, the acceleration cannot change sharply, so the changing rate of acceleration is also constrained by equation (5).

C. Platoon Model

As shown in Fig. 1, we define the vehicle's position p_i as the position of its rear bumper. l_i denotes the length of the ith vehicle V_i, while d_i denotes the safety spacing which is defined as:

$$d_i(k) = \gamma_i \cdot d_{\min,i} + h_i \cdot v_i(k) \tag{7}$$

where γ_i is a safety coefficient which can be selected by the driver in accordance with the road condition and driver's preference. For instance, a bigger γ_i should be set to get a larger safety spacing when the road is wet. $d_{\min,i}$ is the minimal constant inter-distance, and h_i represents a time delay to recognize a hard brake of the preceding vehicle. l_i, $d_{\min,i}$ and h_i are all the specific features of V_i. Hence, the safety spacing d_i is composed of two parts: $\gamma_i \cdot d_{\min,i}$ is the constant part and $h_i \cdot v_i(k)$ is the velocity dependent part. An instance can be referred to the platoon G_3 in Fig. 1.

The spacing error of V_i to its preceding vehicle V_{i-1} can be defined as:

$$\delta_i(k) = p_{i-1}(k) - p_i(k) - d_i - l_i \tag{8}$$

with

$$\delta_i(k) > -d_i. \tag{9}$$

Equation (9) is a strict constraint to be respected in the control process, which aims at avoiding collision in the platoon. As introduced in the control objectives, the platoon should be compact to take full use of the space, while a safety spacing should be maintained to avoid rear-end collision. This objective can be explained as a zero spacing error that we should achieve during the control process.

For the velocity and acceleration coordination of the platoon, conceptions of "expected velocity" and "expected acceleration" are proposed in [22]. In this platoon model, the host vehicle V_i can not only receive the information from its nearest preceding vehicle but also the leading vehicle V_l of the platoon, so that it can make the decision from a larger perspective and the control can be more intelligent. The advantage of this setting is that the host vehicle has the ability to predict the behavior of the platoon, thus a smaller delay of movement transmission can be achieved, and the platoon can converge faster to the steady state. The influence from the platoon leader V_l to the host vehicle V_i is related to the position of the host vehicle in the platoon. The greater the distance to the platoon leader, the host vehicle get less influence from the leader and more influence from the nearest preceding vehicle. The expected velocity and acceleration of the host vehicle V_i are defined as follows:

$$v_{r,i}(k) = (1 - w_i) \cdot v_{i-1}(k) + w_i \cdot v_l(k)$$
$$a_{r,i}(k) = (1 - w_i) \cdot a_{i-1}(k) + w_i \cdot a_l(k) \tag{10}$$

where w_i is the influence weight from the platoon leader, which corresponds to the vehicle's position in the platoon, and it decreases with the distance to the leader. We can notice that the vehicle right behind the leader is completely influenced by the latter.

Therefore, we can define the velocity error and the acceleration error as:

$$\Delta v_{e,i}(k) = v_i(k) - v_{r,i}(k)$$
$$\Delta a_{e,i}(k) = a_i(k) - a_{r,i}(k). \tag{11}$$

The objectives of platoon control can be partially expressed as to make the spacing, velocity, and acceleration error as small as possible. However, these objectives are contradictory in some cases. For example, when the vehicles in the platoon are at the same constant velocity while the spacing error between two vehicles is not zero, the posterior vehicle have to accelerate to eliminate the spacing error, whereas the velocity and acceleration error have to be increased. Thus, a priority must be defined for these three different control objectives.

In order to make decisions individually, each vehicle in the platoon should be capable to measure the three errors based on its own situation and the traffic information coming from related vehicles via V2V communication. Therefore, equations (8) and (10) should be integrated into the discrete vehicle model (2). The measurement output can be divided into two parts according to the information source: the measurement based on its own situation, and the measurement based on the information of the platoon leader and the nearest preceding vehicle. The former can be defined as y_i:

$$\begin{aligned} x_i(k+1) &= A_i x_i(k) + B_i u_i(k) \\ y_i(k) &= C_i x_i(k) \end{aligned} \tag{12}$$

where

$$C_i = \begin{bmatrix} -1 & -h_i & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

And the latter can be defined as z_i:

$$z_i(k) = H_i \xi_i(k) \tag{13}$$

where

$$H_i = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & w_i - 1 & 0 & -w_i & 0 \\ 0 & 0 & w_i - 1 & 0 & -w_i \end{bmatrix}$$

$$\xi_i = \begin{bmatrix} x_{i-1}(k) - l_i - \gamma_i \cdot v_i(k) \\ v_{i-1}(k) \\ a_{i-1}(k) \\ v_l(k) \\ a_l(k) \end{bmatrix}.$$

Therefore, the tracking error of vehicle V_i can be expressed as follows:

$$e_i(k) = y_i(k) + z_i(k) \tag{14}$$

where

$$e_i(k) = [\delta_i(k) \quad \Delta v_{e,i}(k) \quad \Delta a_{e,i}(k)]^T.$$

III. Methodology

In this section, based on the control objectives that we proposed and the models of vehicle and platoon, our CACC-VI algorithm is introduced in details.

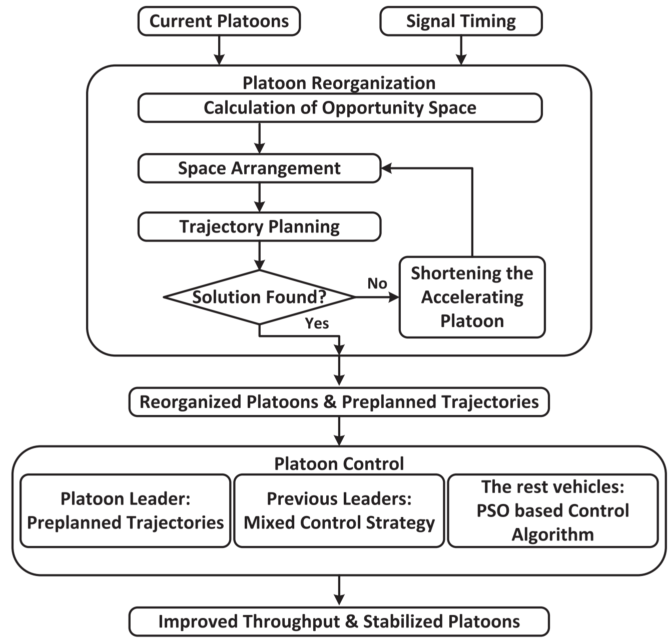


Fig. 2. Overview of CACC-VI.

A. Overview of CACC-VI

First of all, an overview of CACC-VI is given as shown in Fig. 2. The conception of "Opportunity Space" is proposed to evaluate the redundant road capacity, and three sub-algorithms are designed to establish the integrated CACC-VI algorithm. The first sub-algorithm is PSO based platoon control algorithm (Section III-B), which is applied by the vehicles in a platoon except the leader to eliminate the tracking errors that we defined in the previous section; the second one is the trajectory planning algorithm (Section III-C1) which aims at finding an optimal input profile for a vehicle V_i to arrive at the decided future position p_i' at time t with a specific velocity v; the third one is space arrangement (Section III-C2), which is utilized in coordination with the trajectory planning to distribute the opportunity space to as many vehicles as possible, while minimize the distance to be covered by each vehicle.

At first, the intersection manager broadcasts the signal timing to the vehicles in the lane, while the platoons send their current positions and velocities to the manager; then the opportunity space is calculated and distributed to vehicles using space arrangement approach and trajectory planning algorithm; if there is a vehicle in the accelerating platoon which cannot find an input profile, the last vehicle in the platoon is reorganized into the decelerating platoon; then the process is lunched again until all the vehicles in the accelerating platoon can find a solution. After that, the platoon reorganization process is finished. The newly reorganized platoons are controlled by different strategies: the platoon leaders are controlled by following the preplanned trajectories; the previous platoon leaders, which are merged into a new platoon, are controlled by a mixed control strategy that we define in the Section III-D; the rest vehicles are controlled by the PSO based platoon control algorithm. By using the CACC-VI algorithm, the intersection throughput can be improved by making use of the redundant road capacity;

and platoons' motion is controlled with stabilized behaviors in consideration of fuel consumption, heterogeneous vehicular features and passenger comfort.

B. PSO Based Platoon Control

1) Cost Function Design: In order to find an optimal control input for every vehicle at each time step with an integrated consideration of the contradictory between different objectives, a cost function needs to be defined. In this paper, we use a quadratic function of the tracking errors and the control input to find a trade-off between the tracking objectives and the control cost as well.

$$J(V_i) = e_i^T(k)Q_i e_i(k) + u_i^T(k)R_i u_i(k) \tag{15}$$

where Q_i and R_i are respectively the weights of the tracking errors and the control input,

$$Q_i = \begin{bmatrix} q_{i,1} & 0 & 0 \\ 0 & q_{i,2} & 0 \\ 0 & 0 & q_{i,3} \end{bmatrix}, \quad R_i = \text{cont}_i.$$

In fact, the cost function of vehicle V_i is defined as the weighted sum of its tracking errors and the control input at step k, where the weights represent the priorities of the different objectives. In the function, the greater the weight is, the more important the objective will be, so the controller will tend to achieve this objective in priority. For example, if the weight of spacing error is greater than that of velocity error, the controller will adjust the spacing at first, when the spacing error is small enough, the vehicle will begin to adjust its velocity.

In this paper, constraints like velocity limit, power limit, collision avoidance, and passenger comfort should also be considered. The feasible domain of control input is constrained and varying at each time step. Therefore, the studied problem is a constrained optimization problem. We utilize a penalty function technique [39] to solve this constrained optimization problem through a sequence of unconstrained optimization problems.

The penalty function is defined using a trail-and-error method, and the idea is quite straightforward: a higher penalty value is given when a solution goes outside the feasible domain. In fact, the penalty value must be given properly: if it is too high, the optimization algorithm usually get trapped in local optima, while if it is too low, the feasible optimal solutions may not be found easily. The penalty value can be either stationary or non-stationary: for the former, the penalty value is fixed during the optimization process; and for the later, the penalty value is variant with the iteration number. In this paper, we use a non-stationary penalty value.

Another term for the penalty is added on the original cost function (15):

$$J_{\text{constraint}}(V_i) = J(V_i) + h(n) \cdot H(x_i) \tag{16}$$

where $J(V_i)$ is the original cost function that we defined in equation (15), $h(n)$ is the penalty value at the current iteration number n, and $H(x_i)$ is the penalty factor which is defined as bellow:

$$H(x_i) = \sum_{j=1}^{m} \theta\left(q_j(x_i)\right) \cdot q_j(x_i)^{r(q_j(x_i))} \tag{17}$$

where m is the number of constraints, and in this paper, we have speed limit, power limit, collision avoidance, and change rate of acceleration as the four constraints, hence $m = 4$. $q_j(x_i)$ are defined as follows:

$$\begin{aligned} q_1(x_i) &= \max(0, v_i - v_{\text{limit}}) \\ q_2(x_i) &= \max\left(0, \frac{P_{\text{tractive},i}}{\eta_i} - P_{\text{engine},i}\right) \\ q_3(x_i) &= \max(0, -\delta_i - d_i) \\ q_4(x_i) &= \max\left(0, |\Delta a_i| - |\Delta a|_{\max}\right). \end{aligned} \tag{18}$$

$\theta(q_j(x_i))$ is a multi-stage assignment function, $r(q_j(x_i))$ is the power of the penalty function, and their forms are given as bellow:

$$\theta\left(q_j(x_i)\right) = \begin{cases} \nu_1, & \text{if } 0 < q_j(x_i) < \mu_1 \\ \nu_2, & \text{if } \mu_1 \le q_j(x_i) < \mu_2 \\ \vdots & \\ \nu_n, & \text{if } q_j(x_i) \ge \mu_{n-1} \end{cases} \tag{19}$$

$$r(q_j(x_i)) = \begin{cases} 1, & \text{if } 0 < q_j(x_i) \le 1 \\ 2, & \text{if } q_j(x_i) > 1. \end{cases} \tag{20}$$

The constraints on control input (4) is not included in the cost function, they can be added into the initialization procedure of the optimization algorithm.

Therefore, at each time step k, the contradictory control objectives can be summarized as to find the control input $u_i^*(k)$ which minimizes the cost function value. $U_i(k)$ is the feasible domain of control input with respect to all the constraints above.

$$u_i^*(k) = \arg \min_{u \in U_i(k)} \left[J_{\text{constraint}}(V_i)\right]. \tag{21}$$

2) Optimization Algorithm: An optimization algorithm needs to be applied to solve the constrained optimization problem (16) that we have established. Due to the nonlinearity of the objectives and constraints, the optimal input could be difficult to find using deterministic algorithms. Thus, we tend to utilize heuristic algorithms which can handle both nonlinear objective and constraint functions without requiring gradient information. Actually, there are four popular heuristic algorithms derived from natural phenomena: Simulated Annealing Algorithm (SA), Genetic Algorithm (GA), Ant Colony Optimization Algorithm (ACO) and Particle Swarm Optimization Algorithm (PSO). Comparisons among these algorithms can be found in [40]–[42].

The particular swarm optimization (PSO) is a population-based stochastic algorithm developed by Eberhart and Kennedy [43], [44], and it is inspired by the social behavior of animal swarm like group of birds. In this algorithm, a particle is like an individual bird with flies through a multi-dimensional search space where each position in the space is evaluated by a fitness function; each particle adjusts its own trajectory towards its previous best position and the best position of all the particles in a local neighborhood (the neighborhood can refer to the whole swarm). Thus, a particle can benefit from the discovery of its neighbors and the previous discoveries. PSO has the best integrated performance among these four algorithms [40], due to its relatively simple formulation to perform exploration and exploitation, thus lower computational cost can be achieved. Unlike genetic algorithm, there is no selection and mutation procedure, and all the particles survive till the end.

The pseudo code of PSO is summarized in Algorithm 1. At the beginning, the swarm is initialized in the feasible area of the moment with randomized positions and velocities. Basically, each particle in the swarm updates its velocity and position according to the following two equations (22) and (23) at the current iteration n.

$$V_{pi}^{n+1} = \phi \cdot V_{pi}^{n} + c_1 \cdot r_1 \cdot \left(P_{bi} - X_{pi}^{n}\right) + c_2 \cdot r_2 \cdot \left(P_{gb} - X_{pi}^{n}\right) \quad (22)$$

$$X_{pi}^{n+1} = X_{pi}^{n} + V_{pi}^{n} \quad (23)$$

where pi refers to the ith particle and $i = 1, 2, \ldots, N_{\text{swarm}}$, N_{swarm} is the population of the swarm.

Algorithm 1 Pseudo Code of PSO

Input:
Swarm size N_{swarm}, searching area's dimension D_{search}, area's boundaries $(X_{\min}, X_{\max})$, velocity limit $V_{\max}$;
Output:
Global best solution $P_{g\text{best}}$;

```
Initialization of particles' positions and velocities within the boundaries;
repeat
  for particle pi in N_swarm do
    if f(X_pi) < f(P_bi) then
      P_bi = X_pi;
    end if
    if f(P_bi) < f(P_gb) then
      P_gb = P_bi;
    end if
  end for
  for particle pi in N_swarm do
    Update the velocity of the particle in every dimension by (22), and adjust the velocity if it exceeds the boundaries;
    Update the position of the particle in every dimension by (23), and adjust the position if it exceeds the boundaries;
  end for
until iteration > Max_iteration
```

The equation (22) is used to update the velocity of the particle based on the current velocity V_{pi}^{k}, the best position P_{bi} that the particle has reached so far, and the current best position P_{gb} among the neighborhood. ϕ is the inertia weight, r_1 and r_2 are two independent random number, and c_1 and c_2 are called the learning factors. The three parts on the right side of the equation (22) are separately called the inertia part, the cognition part and the social part. The first part represents the exploration process while the second and the third part represent the exploitation process of the algorithm. The inertia weight make the particle tend to explore new areas of the searching space since it cannot easily change its velocity towards the best solutions. The three weights should be tuned carefully, so that PSO can establish trade-off between exploration and exploitation. If not, the algorithm may suffer from premature convergence, trapping in a local optima and stagnation. After the velocity update, the equation (23) updates the position of the particle.

C. Platoon Reorganization

In this subsection, we focus on how to reorganize the platoons in consideration of the road capacity and the power limit of vehicles to maximize the throughput of the intersection. Several problems need to be solved:

- How to quantify the redundant road space that can be taken advantage of?
- How many vehicles to which can be distributed this space?
- How to verify if a vehicle is capable to arrive at the decided position with a certain velocity at the definitive moment?
- If so, how to find an optimal trajectory for this vehicle to save fuel?

These questions can be responded in our trajectory planning and space arrangement algorithms.

1) Trajectory Planning Algorithm: This algorithm is designed to find the solution of the last two questions. First of all, vehicles should be divided into three categories: vehicles which will get through the intersection with a constant velocity; vehicles which will get through the intersection by accelerating; the vehicles which will decelerate to a smaller velocity till the lights turns green again. These three categories are separately given the labels C_1, C_2 and C_3. Our objective is to plan the trajectory for the vehicles with labels C_2 and C_3. Basically, a vehicle needs to arrive at the decided position with a certain velocity at the definitive moment; at the same time, the constraints, like speed limit, power limit and stop-free, need to be followed; besides, the fuel consumption should be considered as well. Hence a feasible trajectory with least fuel consumption should be found.

In our scenario, the platoon of vehicles with label C_2 needs to accelerate at first, then to travel at a constant velocity; as the distance to the nearest preceding platoon of the whole platoon should be reduced, the platoon must have a greater cruise speed; at last, the platoon should decelerate to make its velocity in accordance with the preceding platoon. Thus we can divide the whole process into three sections: acceleration, cruise and deceleration. The platoon of vehicles with label C_3 has a similar three-section trajectory: the platoon should decelerate to a lower cruise speed, and then accelerate to the original speed to pass the intersection. The duration of the three sections are separately noted as t_1, t_2 and t_3.

In this paper, we consider an uniform input for the acceleration and the deceleration section to simplify the trajectory planning process. And the acceleration and deceleration process have the opposite control input, while the input during the cruising process is zero. The research [5] shows that fuel consumption can be reduced by avoiding sharp acceleration/deceleration maneuver, hence the input value should be minimized in order to have a environment friendly driving process. Based on their algorithm, several modifications are proposed according to our scenario. Once the optimal input is found, we can determine the acceleration/deceleration at any time, and the velocity profile can be planed. We note the current position and velocity of the vehicle V_i as p_i and v_i, its future position as p_i', the time to the moment when the light turns red/green as T_r/T_g. Then the trajectory planning problem can be translated into solving the following optimization program:

$$\text{minimize}|u_i| \tag{24}$$

with constraints for vehicles with label C_2:

$$t_1 + t_2 + t_3 \leq T_r, \quad t_1, t_2, t_3 \in (0, T_r) \tag{25}$$

$$v_i(t_1 + t_2 + t_3) = v_{\text{target}} \tag{26}$$

$$\frac{P_{\text{tractive},i}\,(v_i(t_1))}{\eta_i} \leq P_{\text{engine},i} \tag{27}$$

where v_{target} is the velocity of the last platoon of vehicles with label C_1.

Constraints for vehicles with label C_3

$$t_1 + t_2 + t_3 \geq T_g, \quad t_1, t_2, t_3 \in (0, T_g) \tag{28}$$

$$v_i(t_1 + t_2 + t_3) = v_{\text{original}} \tag{29}$$

where v_{original} is the original velocity of the platoon.

Constraints for all the vehicles

$$d\,(v_i(0), u_i, t_1) + d\,(v_i(t_1), 0, t_2) + d\,(v_i(t_1 + t_2), -u_i, t_3) = p_i' - p_i \tag{30}$$

$$0 < v_i(t_1 + t_2) \leq v_{\text{limit}} \tag{31}$$

where $d(v_i(t), u_i, t_j)$ is the distance covered by the vehicle with initial speed $v_i(t)$ and control input u_i, during time t_j.

The constraints (25), (28), (26), (29), and (30) are separately the constraints of time, velocity and position. (27) and (31) are the power limit and the speed limit. The distance and velocity of each moment are calculated by using the vehicle's kinematic model (1). For the vehicles with label C_2, the greatest demanded tractive power is at time t_1. And for vehicles with label C_2/C_3, the greatest/smallest velocity is achieved at time $t_1 + t_2$, thus this velocity should be constrained below the speed limit and above zero to respect the traffic rules and to avoid total stop.

This trajectory planning algorithm can also be used to verify if a vehicle with label C_2 is capable to follow an accelerating trajectory within constraints. If an input value can be found for vehicle V_i, it means V_i is capable to get through the intersection within the current green phase by accelerating, thus V_i can be added into the accelerating platoon. If not, the vehicle is abandoned by the platoon, and it should change its label to C_3 to get through the intersection till the light become green again.

2) Space Arrangement Approach: This part is used to solve the first two problems that we propose in the beginning of this section. In order to apply the trajectory planning algorithm that we introduce before, the vehicle's future position p_i' must be decided. Since the traffic signal timing is broad-casted to all the vehicles within the V2I range, each platoon can find out if it can totally get through the intersection in time by maintaining the current speed. If the last vehicle of the last platoon which can pass is V_j, then we can define the space between the future position p_j' of V_j and the position of traffic lights p_T as the "opportunity space" S.

$$S = p_j' - p_T = p_j + v_j * T_r - p_T \tag{32}$$

where p_T is the position of the traffic lights; for unsignalized intersection, it is the position of the intersection entrance.

As we introduced in the platoon model, a vehicle needs a safety spacing d_i to the preceding one, which is a function of its own features and velocity. We can define the "demanding space" of V_i as the sum of the vehicle length and the safety spacing according to the targeted velocity v_{target}.

$$S_{D,i} = l_i + d_i' = l_i + \gamma_i \cdot d_{\min,i} + h_i \cdot v_{\text{target}}. \tag{33}$$

The idea of our space arrangement approach is to distribute the "opportunity space" to as many vehicles as possible according to their demanding spaces, while to reduce their distances to the future positions.

In the platoon reorganization procedure of our algorithm, after receiving the broad-casted traffic signal timing via V2I communication, by evaluating the future position of the last vehicle in the platoon, each platoon can predict if it can pass by maintaining the current speed. If so, each vehicle in this platoon is classified into the first category, and its label is set to C_1; if not, it is temporarily classified into second category with label C_2. Then the last vehicle of the platoon with label C_1 sends immediately a message to the intersection manager indicating its future position, thus the opportunity space can be derived. The vehicle's response after it receives a broad-casted message is summarized in Algorithm 2.

Algorithm 2 Vehicle's response to a broad-casted message

Input:
The broad-casted message M_{broad} coming from the intersection manager;
Output:
The vehicle V_i's label and message to intersection manager;
1: **if** The message is well received **then**
2: Based on the signal timing T_r and the current velocity v_i, predict the future position p_i' of the last vehicle of the platoon;

TABLE I
UPSTREAM MESSAGE

Message type
Sender vehicle ID
Remaining space

TABLE II
DOWNSTREAM MESSAGE

Message type
Vehicle ID
Demanding spaces
Corresponding IDs

if The Platoon can pass in time, $p'_i > p_T$ **then**
    Set labels of vehicles in this platoon as C_1;
    Send a message to the intersection manager indicating the future position, so that the opportunity space can be calculated;
else
    Set labels of vehicles in this platoon as C_2;
    Send no message;
end if
end if

After the intersection manager has decided the opportunity space, it sends immediately this information combined with the targeted velocity to the first platoon with label C_2. Then each vehicle in this platoon can find its own demanding space according to the targeted velocity. A message propagates form the leader to the tail of the platoon in order to distribute this opportunity space. We call this message the "upstream message," because it goes from downstream to upstream of the travel direction. The message is formed by three fields separately indicating the message type, the sender's ID, and the remaining space which is the opportunity space in the upstream message it receives minus its demanding space. The structure of an upstream message is shown in Table I.

If a vehicle in the platoon receives an upstream message in which the remaining opportunity space is less than its demanding space, the vehicle and its following ones must depart form the current platoon and these vehicles should change their labels from C_2 to C_3. Several operations need to be done in this case: firstly, this vehicle sends an abandoning message to its following vehicles in the platoon so that they can classifies themselves to the third category; secondly, this vehicle sends a V2I message to the intersection message indicating that the opportunity space has been totally distributed, and this message will be rebroadcasted to the following platoons with label C_2, so that they can be informed to change their labels to C_3; thirdly, this vehicle sends a message to the nearest preceding vehicle which is composed of four fields as shown in Table II. This message is called the "downstream message" because of its opposite propagation direction to the upstream message. For a vehicle with label C_3, the "Demanding spaces" field is set to 0; while for a vehicle with label C_2, this field is the set of the demanding spaces of all its following vehicles with label C_2 and itself. The sum of this field is the future position of the nearest preceding vehicle. This setting can ensure that the vehicles' future positions are the nearest to the current ones, thus the distances to be covered are minimal, and fuel consumption is minimized. The future position of vehicle V_i can be expressed as:

$$p'_i = \sum_{j>i, j\in C_2} S_{D,j} + p_T. \tag{34}$$

If the upstream message propagates to the tail of the platoon while the remaining opportunity space is still larger than the demanding space, it means the opportunity space has not been totally distributed, the platoon sends a V2I message to the intersection manager indicating the remaining opportunity space. Then the manager sends this remaining space to the next platoon to restart the distribution process till the space is totally distributed.

The vehicle's response to an upstream message is summarized in Algorithm 3.

Algorithm 3 Vehicle's response to an upstream message

Input:
  The upstream message M_{up} coming from the nearest preceding vehicle;
Output:
  The vehicle V_i's label, upstream message, downstream message and V2I message;
if The message is well received **then**
  if The remaining opportunity space is greater than the demanding space **then**
    Calculate the new remaining opportunity space;
    if V_i is the tail of platoon **then**
      Send a V2I message to the intersection manager indicating the remaining opportunity space;
    else
      Send an upstream message to the nearest followin vehicle V_{i+1};
    end if
  else
    Abandon joining in the vehicle platoon, set its label as C_3 and send the abandoning messages to the following vehicles;
    Set the demanding space as 0 and send the downstream message to the nearest preceding vehicle V_{i-1};
    Send a V2I message to the intersection manager indicating the opportunity space is totally distributed;
  end if
end if

When a vehicle with label C_2 receives a downstream message, it sums up all the items in the "Demanding spaces" to obtain its own future position, and then lunches the trajectory planning algorithm to find an optimal input profile. If the solution exists and the vehicle is the platoon leader, it sends a confirmation message to all the vehicles with label C_2, and a new platoon is formed; if the solution exists and vehicle is not the leader, it adds information to the last two fields of the downstream message and sends it to the nearest preceding vehicle;

if the solution does not exist, the vehicle sends an abandoning message to the last vehicle with label C_2, recalculates the future position and lunch the trajectory planning again; this procedure is repeated until the vehicle finds a solution. When the vehicle becomes the last one in the platoon but the solution still does not exist, it must depart from the accelerating platoon and send a downstream message with zero demanding space. This process is repeated until all the vehicles in the platoon are capable to arrive at the decided future positions with respect to the speed and power limits. The vehicle's response to a downstream message is summarized in Algorithm 4.

Algorithm 4 Vehicle's response to a downstream message

Input:
The downstream message M_{down} coming from the nearest following vehicle;
Output:
The vehicle V_i's label and a downstream message or a confirmation message or an abandoning message;

if The message is well received **then**
  Calculate its future position using the information in the downstream message;
  Lunch the trajectory planning procedure;
  while Solution does not exist && V_i is not the last vehicle in the current platoon **do**
    Abandon the last vehicle, recalculate the future position p'_i and lunch the velocity planning again;
  end while
  if Solution exists **then**
    if Vehicle V_i is the platoon leader **then**
      Send the confirmation message to all the vehicles with label C_2;
    else
      Send the downstream message to the nearest preceding vehicle V_{i-1};
    end if
  else
    Abandon joining in the platoon;
  end if
end if

An example of the space arrangement approach is given in Fig. 3. We assume that V_1 is the last vehicle with label C_1, thus we can define the opportunity space as shown in the figure; with the propagation of upstream messages, the opportunity space is distributed to five vehicles; the vehicle V_7 has to abandon joining in the accelerating platoon because its demanding space is greater than the remaining opportunity space, therefore V_7 sends abandon messages to its following vehicles, a downstream message to V_6, and a V2I message to the intersection manager; then each vehicle temporally with label C_2 begins to plan its input profile after receiving a downstream message in which the relative information can be used to calculate its future position; the vehicle V_6 can find an optimal input profile while the vehicle V_5 can't, therefore V_5 sends an abandoning message to V_6, and the trajectory planning is lunched again; the solution still does not exist, therefore V_5 sends a downstream message with a zero demanding space to V_4; V_2, V_3 and V_4 are all capable to find the solution according to their future positions, therefore a platoon of three vehicles is formed.

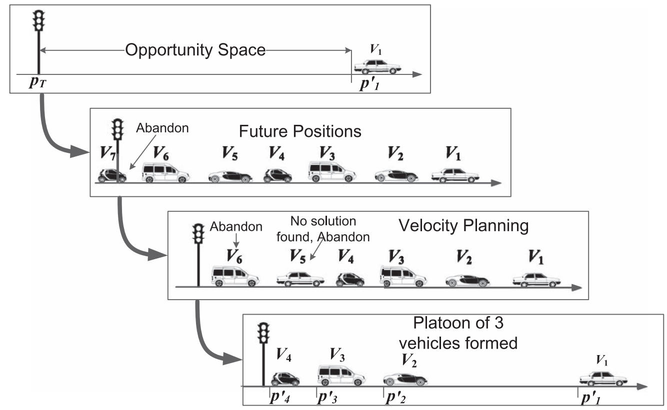


Fig. 3. An example for space arrangement approach.

The future positions of vehicles with label C_3 are quite easy to be decided: the first vehicle in C_3 is denoted as V_k, and its future position at T_g is $p_T - l_k$, because it can only enter the intersection after T_g; the future positions of the rest can be decided by:

$$p'_i = p_T - l_k - \sum_{j>k, j\in C_3} S_{D,j} \tag{35}$$

where $S_{D,j}$ is the demanding space of a vehicle V_j with label C_3:

$$S_{D,j} = l_j + d'_j = l_j + \gamma_j \cdot d_{\min,j} + h_j \cdot v_{\mathrm{original}}. \tag{36}$$

Then the trajectory of vehicles in C_3 can be decided by the trajectory planning algorithm. The platoon should slow down to a smaller velocity then accelerates to its original velocity when the light turns green again. If an optimal control input can be found, the vehicle can get through the intersection without full stop.

By using the trajectory planning algorithm and the space arrangement approach, all the four questions can be responded. The platoons are reorganized and their trajectory are planned separately: the platoons of vehicles with label C_1 get through the intersection with constant velocity; all the vehicles with label C_2 reform a new platoon, and this platoon gets through the intersection by accelerating; all vehicles with label C_3 reform another new platoon which gets through the intersection by the next green phase.

D. Mixed Control Strategy

In fact, the trajectories of vehicles in the platoons with label C_2 and C_3 are planned in advance, and the vehicles in these platoons are capable to arrive at the decided position with a certain velocity at the definitive moment; however, the trajectories are planned individually for each vehicle according to the heterogeneous characteristics, without any consideration of the existence of other vehicles; therefore, a collision-free

TABLE III
VEHICLES' PARAMETERS SETTING

Vehicle No.	p_i (m)	l_i (m)	$P_{engine,i}$ (kw)	τ_i (s)	h_i (s)	$d_{max,i}$ (m)	γ_i	A_i (m^2)	Platoons
V_1	-80.00	5.0	150	0.45	0.40	4.5	1.0	1.5	G_1
V_2	-90.80	4.5	140	0.30	0.30	3.0	1.1	2.0	G_1
V_3	-103.30	4.0	130	0.45	0.40	4.5	1.0	1.8	G_1
V_4	-165.00	4.5	100	0.30	0.30	3.0	1.1	1.7	G_2
V_5	-175.85	3.5	140	0.40	0.35	3.5	1.1	1.6	G_2
V_6	-190.85	5.0	150	0.30	0.40	5.0	1.2	2.0	G_2
V_7	-223.20	5.0	120	0.30	0.35	3.5	1.1	2.0	G_3
V_8	-233.20	3.5	100	0.40	0.35	3.0	1.0	2.2	G_3
V_9	-243.05	3.0	125	0.45	0.30	3.5	1.1	1.8	G_3

procedure cannot be guaranteed by following these preplanned trajectories. Thus, in our algorithm, only a platoon leader's preplanned trajectory is kept, while the rest vehicles in the platoon are still piloted by the PSO based platoon control algorithm with V2V communication.

For the vehicles newly reorganized in a platoon, a mixed strategy is designed: they are treated as a "sub-platoon" in the beginning, which merges into its preceding platoon when the spacing error becomes small enough. In other words, the previous platoon leader utilizes the its preplanned trajectory at first, then it switches to the PSO based control strategy when the distance to the preceding vehicle is small enough; and the other vehicles in the "sub-platoon", if there is any, are always controlled by the PSO based control algorithm. The switch from preplanned trajectory to PSO based algorithm is not reversible. The mixed control strategy can be expressed as following:

$$u_i(k) = \begin{cases} u_{\text{preplanned},i}(k), & \text{if } \delta_i(k) \geq E \\ u_{\text{PSO},i}(k), & \text{if } \delta_i(k) < E \end{cases} \tag{37}$$

where E is the threshold of switch.

IV. SIMULATION

A. Setup

In this section, the CACC-VI algorithm is tested and verified by a designed scenario, in which, nine vehicles circulate in the same lane, the current phase of the traffic lights is green, and it turns to red in $T_r = 18$ s, then the next green phase is at $T_g = 36$ s. The vehicles are divided into three platoons: $V_1 \sim V_3$ are in platoon G_1; $V_4 \sim V_6$ are in platoon G_2; and $V_7 \sim V_9$ are in platoon G_3. Vehicles V_1, V_4 and V_7 are separately the leaders of these three platoons. It is assumed that all the vehicles are cruising at the same constant velocity $v = 10$ m/s in the beginning, thus the targeted velocity v_{target} and the original velocity v_{original} are equal to 10 m/s, therefore the platoons with labels C_2 and C_3 should enter the intersection with velocities of this value. The platoons are controlled by the proposed control algorithm, and the tracking errors in each platoon at $t = 0$ s including spacing error, velocity error and acceleration error are all zeros. The road speed limit is set to $v_{\text{limit}} = 50$ km/h (13.89 m/s) which is a common value of urban road. The control input u_i of each vehicle is limited as $u_i \in [-1.5, 1.5]$. To guarantee the passenger comfort, the changing rate of acceleration should be limited as $\Delta a_i \in [-0.5 \text{ m/s}^3, 0.5 \text{ m/s}^3]$. The traffic lights which indicate the entrance of intersection are installed at $p_T = 0$ m, while the initial positions and features of the nine vehicles are shown in Table III. This table shows that the vehicles are heterogeneous in terms of engine power, length, user preference and many other features, and this heterogeneity accords with the design objective of our CACC-VI algorithm. The threshold in the mixed control strategy is set as $E = 4$ m. The total simulation time is $T = 40$ s, and the sample time is set to $T_s = 0.02$ s. As for the PSO based platoon control algorithm, the parameters are set as followings: $N_{\text{swarm}} = 10$, Max_iteration $= 30$, $\phi = 0.729$, $c_1 = 2.988$, $c_2 = 2.988$. The simulation is executed in *Matlab 2010b* on a computer with *Intel Core i5-2410M* 2.3 GHz.

Based on the signal timing of traffic lights and the current velocities of vehicles, it can be predicted that the first platoon formed by $V_1 \sim V_3$ can fully get through the intersection by maintaining the current velocity, so they are classified into the first category and given the label C_1. Therefore the opportunity space, that we defined in (32) as the space left behind the last vehicle with label C_1 at moment T_r, can be decided as 76.7 m.

B. Simulation Results

1) Platoon Reorganization: The opportunity space is distributed, using the trajectory planning algorithm and the space arrangement approach, to as many vehicles as possible. According to the targeted velocity and the heterogeneous features, the demanding spaces can be calculated, therefore the opportunity space could be distributed to the rest six vehicles.

However, after launching the trajectory planning algorithm for the first time, none of the vehicles can find a feasible solution within constraints for the designed future positions. Thus vehicle V_9 is abandoned, then the future positions are recalculated for the second trajectory planning. This time, V_4, V_5 and V_6 can find a solution while V_7 and V_8 cannot. So vehicle V_8 should abandon joining in the accelerating platoon. In the third trajectory planning with updated future positions, all the four vehicles V_4, V_5, V_6 and V_7 can find feasible solutions. Therefore, their labels should stay unchanged as C_2, and the new platoon G_2' is formed. This platoon is capable to get through the intersection during the current green phase by accelerating while respecting the constraints. As for the rest two vehicles V_8 and V_9, they should change their labels to C_3, and a new platoon G_3' which will get through the intersection in the next green phase, is also formed. In other words, in the

TABLE IV
PLATOON REORGANIZATION OF CACC-VI ALGORITHM

Vehicle No.	$S_{D,i}$ (m)	1^{st} Arrangement Positions	Solutions	2^{nd} Arrangement Positions	Solutions	3^{rd} Arrangement Positions	Solutions	Labels	Platoons
V_4	10.80	61.05	Not Exist	51.20	Exist	41.20	Exist	C_2	G_2'
V_5	10.85	50.20	Not Exist	40.35	Exist	30.35	Exist	C_2	G_2'
V_6	15.00	35.20	Not Exist	25.35	Exist	15.35	Exist	C_2	G_2'
V_7	12.35	22.85	Not Exist	13.00	Not Exist	3.00	Exist	C_2	G_2'
V_8	10.00	12.85	Not Exist	3.00	Not Exist	-	-	C_3	G_3'
V_9	9.85	3.00	Not Exist	-	-	-	-	C_3	G_3'

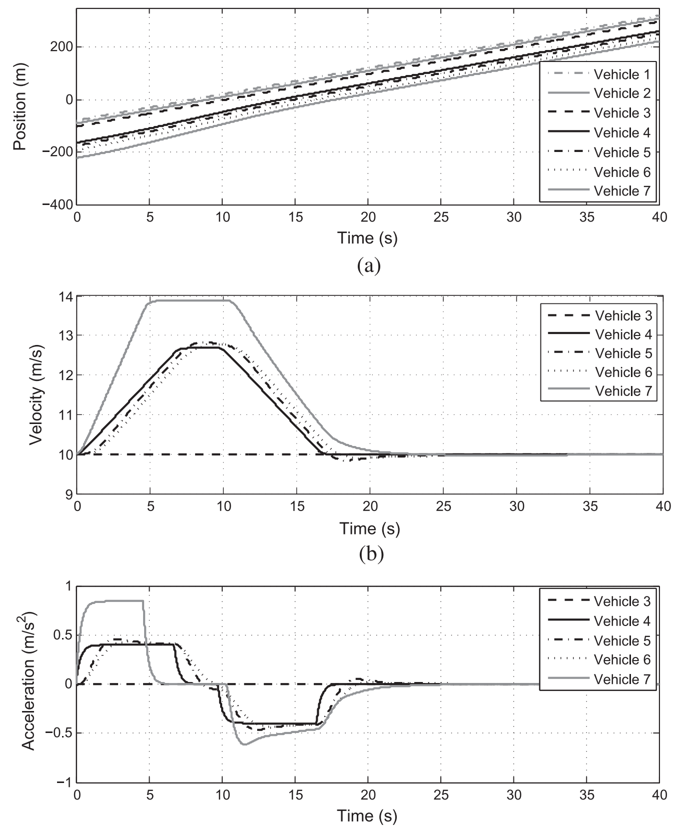
platoon reorganization procedure, the vehicle V_7 departs from the current platoon G_3 and joins in the preceding platoon G_2 to form a new platoon G_2'; at the same time, the rest of G_3 becomes a new platoon G_3'.

This is the optimal result regarding to constraints such as space, velocity, engine power and time. The results of the platoon reorganization are summarized in Table IV.

2) Accelerating Platoon: For the platoon G_2', which has the opportunity to get through the intersection by accelerating during the current green phase of traffic lights, the leader V_4 utilizes the preplanned trajectory while the previous leader V_7 of platoon G_3 utilizes the mixed control strategy, and the rest is controlled by the platoon control algorithm based on PSO. For illustration, the position, the velocity and the acceleration of the vehicles are shown in Fig. 4, the tracking errors are shown in Fig. 5, and the control inputs are shown in Fig. 6. By the result of trajectory planning, V_4 needs to accelerate for $t(V_4)_1 = 7$ s with control input $u(V_4)_1 = 0.4$; then cruising with $u(V_4)_2 = 0$ for $t(V_4)_2 = 3$ s; finally, decelerate with negative input $u(V_4)_3 = -0.4$ in $t(V_4)_3 = 7$ s. Due to the equality of the original velocity and the targeted one, the acceleration and the deceleration process have the same duration. The previous platoon leader V_7 tends to utilize the preplanned trajectory at first, because it is still far away from its nearest preceding vehicle V_6. The preplanned trajectory for V_7 is to accelerate with the input $u(V_7)_1 = 0.8$ during $t(V_7)_1 = 5$ s; then to cruise with $u(V_7)_2 = 0$ for $t(V_7)_2 = 7$ s; at last to decelerate with $u(V_7)_3 = -0.8$ during $t_3 = 5$ s.

The members of G_2' except V_4 need to find the optimal control input within the constraints based on its current state and the information coming form related vehicles: the platoon leader and the nearest preceding vehicle. In this scenario, the vehicle V_5 receives the information only from V_4, because the platoon leader and the nearest preceding vehicle are the same one; while V_6 receives the information from V_4 and V_5, and V_7 from V_5 and V_6.

The positions of the vehicles in G_2' at $T_r = 18$ s are separately 41.10 m, 30.61 m, 15.51 and 2.15 m, so the whole platoon passes the intersection. The intersection throughput during the current green phase is improved from three vehicles to seven vehicles. Although there are some errors to the future positions decided in the platoon reorganization process, the main objective of the algorithm is achieved. All the vehicles can arrive at the targeted velocity 10 m/s after $t = 20$ s, so the design objective of velocity is also achieved. The PSO based platoon control algorithm works well for V_5 and V_6, because they are capable to track the trajectory of leader V_4, and all the tracking errors are well limited and eliminated. From Fig. 4(b), the three-section velocity trajectory of V_4 and V_7 can be clearly noticed, and their top velocities are well limited below $v_{\text{limit}} = 13.89$ m/s. The initial spacing error of V_7 is $\delta_7(0) = 20$ m, therefore it must catch up with the majority of G_2': at first, it follows the preplanned trajectory with helps it to arrive at the speed limit in five seconds, so that the spacing error can be quickly reduced, whereas the velocity error and acceleration error increase in this period; when V_7 is close enough to the preceding vehicle V_6 ($\delta_7 < 4$ m) at around $t = 10$ s, it switches to PSO based platoon control algorithm in order to eliminate



(a) (b) (c)

Fig. 4. Motion of accelerating platoon G_2'. (a) Positions: All the vehicles in the platoon get through the intersection at $T_r = 18$ s with positions $p_4 = 41.10$ m, $p_5 = 30.61$ m, $p_6 = 15.51$ m and $p_7 = 2.15$ m; the intersection throughput is improved from 3 to 7. (b) Velocities: The velocities are well limited below the speed limit $v_{\text{limit}} = 13.89$ m/s; the velocities tend to converge to the targeted velocity $v_{\text{target}} = 10$ m/s. (c) Accelerations: V_4 utilizes the preplanned trajectory, V_5 and V_6 utilizes the PSO based control algorithm, and V_7 utilizes the mixed control strategy; the delay between V_6 and V_5 is smaller than that between V_5 and V_4.

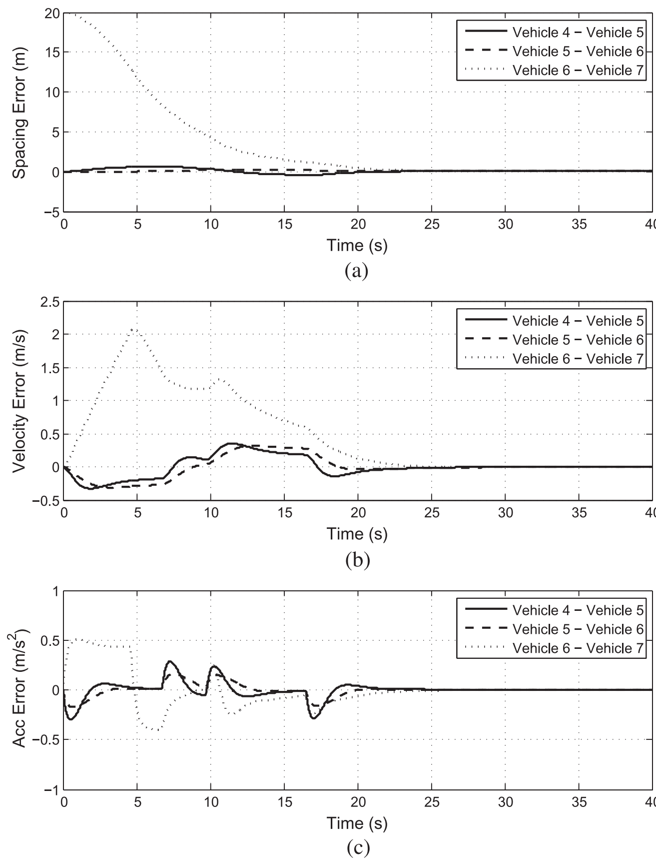


Fig. 5. Tracking errors in the platoon G_2'. (a) Spacing errors: The spacing error δ_7 reduces with time; all the spacing errors become zero after $t = 25$ s; the rear-end collision is absolutely avoided. (b) Velocity errors: The velocity error $\Delta v_{e,7}$ increases during the first 5 s; then all the velocity errors in the platoon G_2' are limited and converge to 0 after the platoon passes the intersection. (c) Acceleration errors: Acceleration errors in the platoon G_2' are also limited and converge to 0 after the platoon passes the intersection.

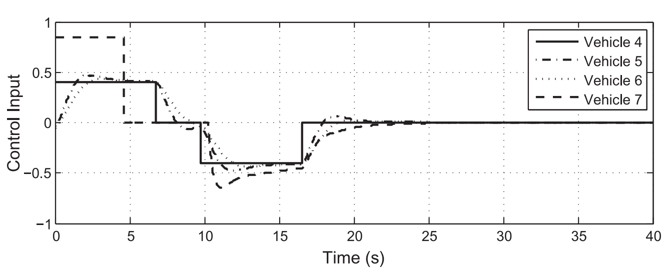


Fig. 6. Control inputs of vehicles in the platoon G_2'.

all the tracking errors. Due to the limit of acceleration changing rate, the passenger will not feel the switch process. The relative information can be referred to Figs. 4(b), (c), 5, and 6.

All the tracking errors of platoon G_2' are eliminated after $t = 25$ s; rear-end collision is absolutely avoided; the platoon is stabilized when facing the influence of the leader's changing behavior. Moreover, the difference between the two communication topologies can also be noticed: in Fig. 4(b) and (c), we can find a smaller delay between V_5 and V_6 than that between V_4 and V_5. Due to the communication with the platoon leader and the nearest preceding vehicle, V_6 can predict the behavior of V_5, so that it can execute the maneuver in advance.

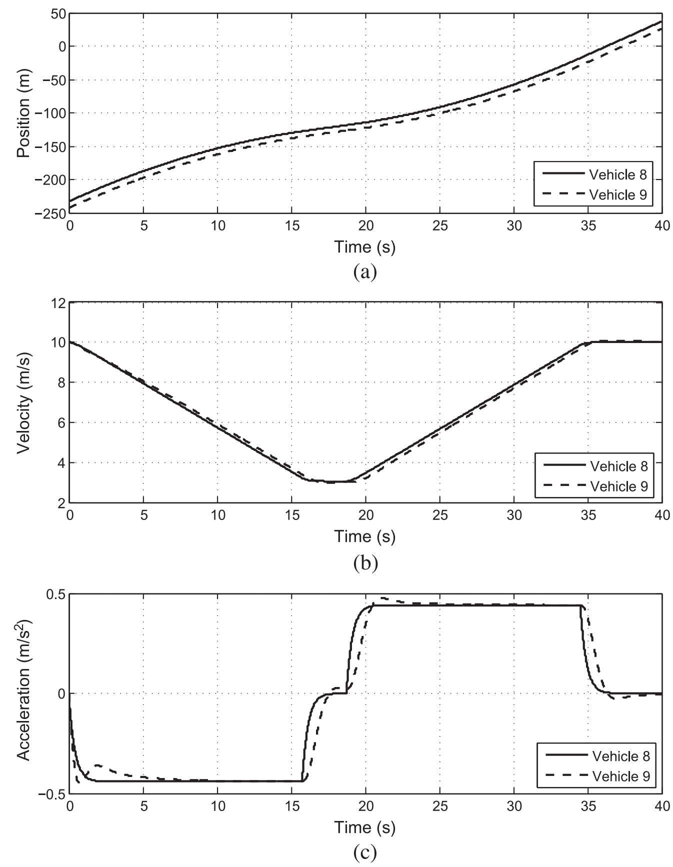


Fig. 7. Motion of decelerating platoon G_3'. (a) Positions: At $T_g = 36$ s, the positions of V_8 and V_9 are separately $p_8 = -3.55$ m and $p_9 = -14.10$ m; the rear-end collision is absolutely avoided with limited spacing error. (b) Velocities: The lowest speed is 3 m/s, the platoon G_3' does not need to fully stop; the velocities at $T_g = 36$ s are all equal to 10 m/s. (c) Accelerations: V_8 utilizes the preplanned trajectory while V_9 applies the PSO algorithm to track its behavior.

In this scenario, the computational cost of the PSO algorithm is quite low: by applying the setup in the Section IV-A, to find a solution for a vehicle in one time step, it costs at most 0.032 s, at least 0.015 s, on average 0.017 s which is smaller than the sampling time $T_s = 0.020$ s. However, it still cannot be guaranteed that the vehicle can find a solution during the sampling time. In the near future, when higher computational ability can be obtained more cheaply, this PSO based platoon control algorithm can be implemented for real-time usage.

3) Decelerating Platoon: For the platoon G_3', which is formed by V_8 and V_9, it cannot get through the intersection within the current green phase, therefore it chooses to decelerate to a lower speed without full stop and to pass the intersection during next green phase.

Similar to the platoon G_2', the leader V_8 follows the preplanned trajectory: V_8 should decelerate with $u(V_8)_1 = -0.44$ during $t(V_8)_1 = 16$ s, then cruise with $u(V_8)_2 = 0$ in $t(V_8)_2 = 4$ s, at last, accelerate with $u(V_8)_3 = 0.44$ within $t(V_8)_3 = 16$ s. V_9 calculates the near optimal solution using the PSO based control algorithm according to the designed constraints and the information from V_8. The results can be found in Fig. 7. At $T_g = 36$ s when the next green phase begins, the positions of V_8 and V_9 are separately $p_8 = -3.55$ m and $p_9 = -14.10$ m which are very close to their distributed future position -3.50 m and -13.35 m. The lowest speed is 3 m/s, in other words, the platoon G_3' does not need to fully stop to wait for the next green phase. The velocities at $T_g = 36$ s are all equal to 10 m/s, it means the platoon recovers to its original velocity. Thus the control objectives of G_3' are all achieved.

The simulation results can prove that the CACC-VI algorithm that we designed achieves all the control objectives that we established in the beginning.

V. Conclusion

The most important contribution of this paper is the presentation of a potential application of V2X communication to ameliorate the road transportation systems in terms of safety, time-saving, and environment friendly. In this paper, we have proposed a decentralized cooperative adaptive cruise control algorithm for vehicles in the vicinity of intersections (CACC-VI), which aims at maximizing the throughput of intersection within a limited green phase of traffic lights, by making use of the remnant road capacity, while considering safety, fuel saving, velocity limitation, heterogeneous dynamics of vehicles and passenger comfort.

The CACC-VI algorithm composes of two main parts: platoon reorganization and platoon control. In the first part, a trajectory planning algorithm is introduced which is used to find a feasible input profile to accomplish designed motion within constraints; and a space arrangement approach is designed, in which the vehicles use different types of messages to share the information of the related spaces, the targeted velocity and the pass capability in order to take full use of the opportunity space while reduce the distances should be covered by the vehicles in the platoon; with these two methods, the platoons become flexible: vehicles can choose to accelerate to join in the preceding platoon or decelerate to depart from the current platoon. In the second part, different control strategy are used: the current platoon leaders follow the preplanned trajectories designed in the first part; the previous leaders follow a mixed control strategy; and the rest vehicle use a constrained PSO algorithm to find the optimal control input to minimize the value of the cost function, which is defined considering the tracking errors, the control input and the different types of constraints.

Simulation experiments show that our algorithm is capable to control vehicles of different situations; the number of vehicles which can get through the intersection during the green phase can be greatly improved; and the optimal control input can be found within the defined constraints so that the disturbances like accelerating and decelerating from the platoon leader to the rest of the platoon can be attenuated.

However, there are still some aspects that we have not taken into account. For example, in reality, the vehicles enter the effect area of the algorithm progressively, it is necessary to decide the activation moment of the algorithm; a larger scale simulation need to be conducted for fully functional intersections to further test the effectiveness of this algorithm. These aspects will be studied in our future research.

References

[1] S. L. Poczter and L. M. Jankovic, "The Google Car: Driving toward a better future?" *J. Bus. Case Stud.*, vol. 10, no. 1, pp. 7–14, 1st Quart. 2013.

[2] H. Hartenstein and K. Laberteaux, *VANET: Vehicular Applications and Inter-Networking Technologies*. Hoboken, NJ, USA: Wiley, 2010.

[3] R. Sengupta *et al.*, "Cooperative collision warning systems: Concept definition and experimental implementation," *J. Intell. Transp. Syst.*, vol. 11, no. 3, pp. 143–155, 2007.

[4] J. W. Wedel, B. Schunemann, and I. Radusch, "V2X-based traffic congestion recognition and avoidance," in *Proc. 10th ISPAN*, 2009, pp. 637–641.

[5] S. Mandava, K. Boriboonsomsin, and M. Barth, "Arterial velocity planning based on traffic signal information under light traffic conditions," in *Proc. 12th IEEE ITSC*, 2009, pp. 1–6.

[6] B. Asadi and A. Vahidi, "Predictive cruise control: Utilizing upcoming traffic signal information for improving fuel economy and reducing trip time," *IEEE Trans. Control Syst. Technol.*, vol. 19, no. 3, pp. 707–714, May 2011.

[7] M. Madireddy *et al.*, "Assessment of the impact of speed limit reduction and traffic signal coordination on vehicle emissions using an integrated approach," *Transp. Res. D, Transp. Environ.*, vol. 16, no. 7, pp. 504–508, Oct. 2011.

[8] R. K. Kamalanathsharma and H. A. Rakha, "Leveraging connected vehicle technology and telematics to enhance vehicle fuel efficiency in the vicinity of signalized intersections," *J. Intell. Transp. Syst.*, vol. 20, no. 1, pp. 33–44, 2016.

[9] R. K. Kamalanathsharma and H. Rakha, "Agent-based modeling of eco-cooperative adaptive cruise control systems in the vicinity of intersections," in *Proc. 15th IEEE ITSC*, 2012, pp. 840–845.

[10] K. J. Malakorn and B. Park, "Assessment of mobility, energy, and environment impacts of intellidrive-based cooperative adaptive cruise control and intelligent traffic signal control," in *Proc. IEEE ISSST*, 2010, pp. 1–6.

[11] C. Desjardins and B. Chaib-draa, "Cooperative adaptive cruise control: A reinforcement learning approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 4, pp. 1248–1260, Dec. 2011.

[12] B. Van Arem, C. J. van Driel, and R. Visser, "The impact of cooperative adaptive cruise control on traffic-flow characteristics," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 4, pp. 429–436, Dec. 2006.

[13] V. Milanés *et al.*, "Cooperative adaptive cruise control in real traffic situations," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 1, pp. 296–305, Feb. 2014.

[14] S. Calvert, T. Van den Broek, and M. Van Noort, "Modelling cooperative driving in congestion shockwaves on a freeway network," in *Proc. 14th IEEE ITSC*, 2011, pp. 614–619.

[15] W. J. Schakel, B. van Arem, and B. D. Netten, "Effects of cooperative adaptive cruise control on traffic flow stability," in *Proc. 13th IEEE ITSC*, 2010, pp. 759–764.

[16] J. Zhao, M. Oya, and A. El Kamel, "A safety spacing policy and its impact on highway traffic flow," in *Intell. Vehicles Symp., 2009 IEEE*, 2009, pp. 960–965.

[17] J. Ploeg, B. Scheepers, E. Van Nunen, N. Van de Wouw, and H. Nijmeijer, "Design and experimental evaluation of cooperative adaptive cruise control," in *Proc. 14th IEEE ITSC*, 2011, pp. 260–265.

[18] K. Lidström *et al.*, "A modular CACC system integration and design," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 3, pp. 1050–1061, Sep. 2012.

[19] M. R. Nieuwenhuijze, T. van Keulen, S. Öncü, B. Bonsen, and H. Nijmeijer, "Cooperative driving with a heavy-duty truck in mixed traffic: Experimental results," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 3, pp. 1026–1032, Sep. 2012.

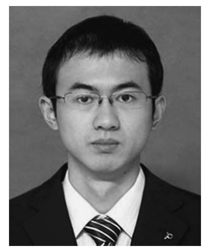
[20] L. Guvenc *et al.*, "Cooperative adaptive cruise control implementation of team Mekar at the Grand Cooperative Driving Challenge," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 3, pp. 1062–1074, Sep. 2012.

[21] D. de Bruin, J. Kroon, R. van Klaveren, and M. Nelisse, "Design and test of a cooperative adaptive cruise control system," in *Proc. IEEE Intell. Veh. Symp.*, 2004, pp. 392–396.

[22] S.-Y. Han, Y.-H. Chen, L. Wang, and A. Abraham, "Decentralized longitudinal tracking control for cooperative adaptive cruise control systems in a platoon," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, Oct. 2013, pp. 2013–2018.

[23] A. Geiger *et al.*, "Team AnnieWAY's entry to the 2011 Grand Cooperative Driving Challenge," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 3, pp. 1008–1017, Sep. 2012.

[24] R. Kianfar *et al.*, "Design and experimental validation of a cooperative driving system in the Grand Cooperative Driving Challenge," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 3, pp. 994–1007, Sep. 2012.
[25] S. Li, K. Li, R. Rajamani, and J. Wang, "Model predictive multi-objective vehicular adaptive cruise control," *IEEE Trans. Control Syst. Technol.*, vol. 19, no. 3, pp. 556–566, May 2011.
[26] L. Li, D. Wen, and D. Yao, "A survey of traffic control with vehicular communications," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 1, pp. 425–432, Feb. 2014.
[27] Q. He, K. L. Head, and J. Ding, "PAMSCOD: Platoon-based arterial multi-modal signal control with online data," *Transp. Res. C, Emerging Technol.*, vol. 20, no. 1, pp. 164–184, Feb. 2012.
[28] X.-F. Xie, S. F. Smith, L. Lu, and G. J. Barlow, "Schedule-driven intersection control," *Transp. Res. C, Emerging Technol.*, vol. 24, pp. 168–189, Oct. 2012.
[29] Y. Jiang, S. Li, and D. E. Shamo, "A platoon-based traffic signal timing algorithm for major-minor intersection types," *Transp. Res. B, Methodol.*, vol. 40, no. 7, pp. 543–562, Aug. 2006.
[30] X.-F. Xie, G. J. Barlow, S. F. Smith, and Z. B. Rubinstein, "Platoon-based self-scheduling for real-time traffic signal control," in *Proc. IEEE Conf. Intell. Transp. Syst.*, 2011, pp. 879–884.
[31] C. Priemer and B. Friedrich, "A decentralized adaptive traffic signal control using V2I communication data," *Proc. IEEE Conf. Intell. Transp. Syst.*, 2009, pp. 1–6.
[32] L. Li and F.-Y. Wang, "Cooperative driving at blind crossings using intervehicle communication," *IEEE Trans. Veh. Technol.*, vol. 55, no. 6, pp. 1712–1724, Nov. 2006.
[33] K. Dresner and P. Stone, "A multiagent approach to autonomous intersection management," *J. Artif. Intell. Res.*, vol. 31, no. 1, pp. 591–656, Jan. 2008.
[34] I. H. Zohdy and H. A. Rakha, "Intersection management via vehicle connectivity: The intersection cooperative adaptive cruise control system concept," *J. Intell. Transp. Syst.*, vol. 20, no. 1, pp. 17–32, 2016.
[35] C. Frese and J. Beyerer, "A comparison of motion planning algorithms for cooperative collision avoidance of multiple cognitive automobiles," in *Proc. IEEE Intell. Veh. Symp.*, 2011, pp. 1156–1162.
[36] I. H. Zohdy and H. Rakha, "Game theory algorithm for intersection-based cooperative adaptive cruise control systems," in *Proc. 15th IEEE ITSC*, 2012, pp. 1097–1102.
[37] J. Wu, A. Abbas-Turki, and A. El Moudni, "Cooperative driving: An ant colony system for autonomous intersection management," *Appl. Intell.*, vol. 37, no. 2, pp. 207–222, Sep. 2012.
[38] P. Mirchandani and F.-Y. Wang, "RHODES to intelligent transportation systems," *IEEE Intell. Syst.*, vol. 20, no. 1, pp. 10–15, Jan./Feb. 2005.
[39] K. E. Parsopoulos and M. N. Vrahatis, "Particle swarm optimization method for constrained optimization problems," in *Proc. Euro.-Int. Symp. Comput.*, 2002, vol. 76, pp. 214–220.
[40] W. Meihong and Z. Wenhua, "A comparison of four popular heuristics for task scheduling problem in computational grid," in *Proc. Int. Conf. Wireless Commun. Netw. Mobile Comput.*, 2010, pp. 1–4.
[41] V. Selvi and D. R. Umarani, "Comparative analysis of ant colony and particle swarm optimization techniques," *Int. J. Comput. Appl.*, vol. 5, no. 4, pp. 1–6, Aug. 2010.
[42] R. Hassan, B. Cohanim, O. De Weck, and G. Venter, "A comparison of particle swarm optimization and the genetic algorithm," presented at the 46th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics Materials Conf., Austin, TX, USA, Apr. 18–21, 2005, Paper AIAA 2005-1897.
[43] R. C. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *Proc. 6th Int. Symp. Micro Mach. Human Sci.*, Nagoya, Japan, 1995, vol. 1, pp. 39–43.
[44] J. Kennedy, "Particle swarm optimization," in *Encyclopedia of Machine Learning*. New York, NY, USA: Springer-Verlag, 2010, pp. 760–766.



Bing Liu received the B.S. degree and both the M.S. degree and the General Engineering Diploma from Beihang University, Beijing, China, in 2011 and 2014, respectively, and the Master Diploma in control systems from the École Centrale de Lille, Villeneuve d'Ascq, France, in 2013. In 2012–2013, he was an exchange student with the École Centrale de Lille. For six months, he was an Intern with the Vélizy-Villacoublay R&D Center, PSA Peugeot Citroën. He is currently working toward the Ph.D. degree (second year) with the Centre de Recherche en Informatique, Signal et Automatique de Lille (CRIStAL), École Centrale de Lille, with the specialty in intelligent transportation, under the supervision of Prof. Abdelkader El Kamel. His research interests are in the areas of intelligent transportation systems, advanced driver-assistance systems, V2X communication, traffic safety, and traffic efficiency.

Abdelkader El Kamel (M'96–SM'00) received both the Engineering Diploma and the Master Diploma and the Ph.D. degree from the École Centrale de Lille, Villeneuve d'Ascq, France, in 1990 and 1994, respectively. For five years (2008–2013), he was in charge of the Department of Computer and Telecommunication and Robotics, École Centrale Beijing. He is a Permanent Visiting Professor in major engineering/business schools in Tunisia and a Visiting Professor in Chili, Canada, India, and Japan. He is currently a Full Professor with the École Centrale de Lille. His major research interests include intelligent systems monitoring, complex systems analysis and control, computational intelligence, and optimization. Applying intelligent technologies, virtual reality, and optimization techniques in transportation systems and mobile cooperative robots are among the current focuses of work. He has published more than 100 technical papers in international journals and conferences, and he has participated in five books in his field of interest. Dr. El Kamel has been an invited speaker for more than 25 plenary lectures or tutorials in international conferences, and he serves on the Program Committee for about 150 IEEE SMC and CSS, IFAC, and several other conferences. He has edited more than 15 proceedings of various conferences and workshops