Development of a Distributed Cooperative Vehicles Control Algorithm Based on V2V Communication

Yi Zheng, Lisheng Jin*, Linlin Gao, Keyong Li, Yan Wang and Faji Wang

School of Transportation, JiLin University, ChangChun 130024, China

Abstract

This paper presents a new distributed Cooperative Vehicles Control (CVC) algorithm without any infrastructure. The algorithm is developed to manipulate individual vehicles’ travelling maneuvers based on V2V communication, so that vehicles can cross the intersection safely and rapidly. Assuming that all vehicles are fully automated, by eliminating the collision probability of vehicles at intersections, the CVC algorithm seeks a safe maneuver for vehicles approaching the intersection and manipulates them. To evaluate the performance of algorithm, we develop an integrated simulation framework. The results show that the CVC algorithm can significantly improve the efficiency of intersection compared with conventional signal control system under varying congestion conditions, moreover, the impact of communication performance on algorithm is also considered.

Keywords: cooperative vehicles control algorithm; V2V communication; collision probability; simulation framework

1. Introduction

The growth of traffic flow in urban networks has recently brought lots of social problems such as heavy traffic congestion, traffic accidents and environment pollution. According to the statistics report of National Highway Traffic Safety Admin (NHTSA), in 2008, about six million traffic accidents occur on US Roads [15]. The total
number of crashes reported in 2009 was 14% lower than that in 1999, still, 33808 people died, and about 2.2 million people were injured in 2009 [15], which is a traffic safety statistics that remains unsatisfactory. Intersection is the key area influencing the safety and efficiency of transportation system, unsignalized intersection is a common intersection type. Consequently, solving the safety and efficiency problem of the unsignalized intersection has become an imminent transportation demand.

The Connected Vehicles (CV) [1] is not a dependent technology solution, but an integration of wireless communications, data analytics, robotics, automatic etc. Compared with technologies such as visual sensors and computer processing, the CV can identify hazards on the roadway, and provide a two-way wireless communication environment enabling V2V/V2I communications. In vehicular ad-hoc network (VANET), vehicles equipped with communication devices could not only collect previous traffic data such as individual vehicle’s travelling maneuver, origination/destination, and trajectory but also share these data with other vehicles or infrastructures as well. In fact, cooperation between vehicles and/or between vehicles and infrastructure under CV environment has received a great deal of attention for its potential benefits. Many research projects on cooperative control system have been conducted to improve road safety, reduce exhaust emissions and avoid vehicle collisions.

Yusuke and Yasuhisa et al. [21] proposed a control mechanism of cooperative vehicles at intersection without any infrastructure, embedding communication device into the automated vehicles. Nonetheless, the limitations of algorithm are also notable, more specifically, simulation environment is too simple to reflect the actual situation accurately. Ashil and Lamine presented an adaptive, decentralized and cooperative collision avoidance system, which applies a constrained coupled EKF to estimate the states of a realistic, nonlinear, 5-state variable vehicle model [9]. The estimates are transmitted wirelessly to other vehicles using the up-to-date WAVE standards, and then vehicles use these estimates to optimize future movements. Qiu Jin et al. developed an intersection management system using a multi-agent approach which includes vehicle agent and intersection agent [16]. In addition, multi-layer reservation policies and vehicle motion planning algorithm were developed. The simulation results showed that the system can considerably mitigate traffic congestion as well as reduce emissions and fuel consumption. Lee J and Park B proposed a decentralized cooperative vehicle intersection control (CVIC) algorithm [12]. The authors assumed that an intersection controller embedded within the roadside equipment, and the controller tried to find a safe maneuver for every vehicle by eliminating the potential overlaps of vehicular trajectories coming from all conflicting approaches. Its main limitation is that communication performances are assumed to be perfect, resulting in no packet drop or transmission delay.

As discussed above, cooperative vehicle intersection control is a significant road safety issue. Despite such promising benefits addressed by relevant studies reviewed in this section, treatments for packet drop and transmission delay were not clearly presented. In addition, most of studies were insufficient to cover varying traffic congestion cases. To handle such challenges, a distributed CVC algorithm based on V2V communication is proposed, which aimed at improving unsignalized intersection efficiency and ensuring vehicle safety. With respect to wireless communication, it was assumed that all involved vehicles are fully automated, equipping with communication device insides.

The goal of this paper is to measure the effectiveness of the CVC algorithm, and quantify the potential benefits of algorithm under varying traffic congestion and communication conditions. The algorithm is aimed at unsignalized intersections without any infrastructure, such as traffic signals, signs, and our main contributions can be summarized as follows.

1) The best driving maneuvers are obtained by estimating the collision probability at intersections, and adjust the speed of vehicles globally to avoid the crash.

2) The coupled simulation framework that integrated with traffic simulator VISSIM 4.3, network simulator NS-3.20 is developed.

3) An improved shuffled frog leaping algorithm is presented for solving the optimization problem derived from the collision probability of vehicles.

4) The performances of the distributed CVC algorithm are evaluated under varying traffic congestion and communication conditions.
2. Methodology

2.1. The Estimation of Collision Probability

The size of time to collision (TTC) [11] indicates the severity of conflicting events, that is the probability that a collision could result from a conflicting, hence, a lower TTC indicates a higher probability of a collision. Please notice that, the meanings of TTC for rear-end and crossing collision is different. When two consecutive vehicles get close, the value of TTC becomes big, and this situation is so dangerous. Thus, this paper mainly considers crossing collision, vehicles are not allowed to change lane and make a turn. With the prediction of TTC, we introduced a transformed coefficient CPI [11] (Collision Probability Indicator) for describing the possibility of potential collision for two vehicles, it is computed with a simple form as (1), note that collision scenario involving three or more vehicles isn’t considered in this paper.

\[ P_{cp}(k) = e^{-\frac{1}{\delta}(\text{TTC}(k))} \]  

(1)

Where \( \delta \) is a scale parameter, from this equation, it could be easily found that \( P_{cp}(k) \) is a monotone decreasing function about TTC that satisfies \( P_{cp}(k) \geq 0, 1 \). When \( P_{cp}(k) \geq \varepsilon \), we believe it is necessary for adjusting the speed of vehicles. Let the value of \( \delta \) and \( \varepsilon \) be 2 and 0.005 according to test results. The intersection with phase diagram is shown in Fig. 1, the conflicting relationship between phases can be described by the conflict matrix \( C = [c_{ij}] \), 1 if phases are conflicted, 0 otherwise. Since it is unnecessary to consider non-conflicting phases when summing up the collision probability of phase \( i \), corresponding counter phases would be selected based on the conflict matrix \( C \).

![Fig. 1. The intersection with phase diagram.](image)

In this paper, we propose an improved collision probability estimation of vehicles at intersections, which is based on Stefan and Bastian’s collision probability algorithm [17] using a probabilistic model for the trajectories of vehicles. The dynamics of vehicles are expressed as follows: previous location \( (x, y) \), vehicle speed \( v \), acceleration \( a \) and heading angle \( \theta \). The trajectory of vehicle can be identified by a tuple \( (x, y, v, a, \theta) \), hence, we can define a collision function only depending on these values. The general form of collision probability estimation is formulated as (2), and our objective is to minimize \( J \) as much as possible.

\[ \text{Obj.Fun: } J = \sum_{i=1}^{P} \sum_{k=1}^{L_i} \sum_{m=1}^{N_x} \sum_{j=1}^{L_j} \sum_{n=1}^{N_y} \sum_{s=1}^{a_{max}} \sum_{a_{min}} \int f(X_n, X_n) \text{coll}(x_n, y_n, v_n, a_n, \theta_n) da_n \]  

(2)
Subject to:
1) Initial state conditions: with current time being $t_0$, $X_i$ represents the trajectory of vehicle $i$.

$$\dot{X}_i(t)|_{t = t_0} = v_i$$  \hspace{1cm} (3a)

$$\ddot{X}_i(t)|_{t = t_0} = a_i$$  \hspace{1cm} (3b)

2) Minimum Headway: the final form of two consecutive vehicles’ (e.g. vehicle $i$ and $i+1$) minimum headway constraint is derived as (4a)-(4c).

$$S \left( \frac{1}{2} (a_i - a_{i+1}) R^2 \right) - (a_i - v_i + v_{i+1}) R + S > 0$$  \hspace{1cm} (4a)

$$S = \frac{1}{2} a_i - v_i - X_i(t_0) + X_{i+1}(t_0)$$  \hspace{1cm} (4b)

$$R = \frac{1}{a_{i+1}} (-v_{i+1} + \sqrt{v_{i+1}^2 + 2 a_i X_{i+1}(t_0)})$$  \hspace{1cm} (4c)

3) Box bounds on control actions:

$$a_i \geq \max(a_{\min} \cdot \frac{\sqrt{v_i^2 - 2 v_{i}^2}}{2 x_i(t_0) - 2 x_i(t)}),$$

$$a_i \leq \min(a_{\max} \cdot \frac{\sqrt{v_{i}^2 - v_i^2}}{2 (x_i(t_0) - x_i(t))})$$  \hspace{1cm} (5)

Where:

- $P$: total phase numbers.
- $i, j$: phase number indices.
- $k, l$: lane identifier.
- $L_i, L_j$: total number of lanes of phases $i$ and $j$, respectively.
- $N_{ik}, N_{jl}$: total number vehicles on lane $k$ and $l$ of phases $i$ and $j$, respectively.
- $f(X_m, X_n)$: the probability that trajectories $X_m$ and $X_n$ are chosen.
- $coll(X_m, X_n)$: collision function between vehicle $m$ and $n$.

2.2 Control Algorithm

The motion control algorithm is distributed, which implies that there are no infrastructures (i.e. intersection controller) coordinating the motions of involved vehicles. The algorithm is developed based on Guo and Parker’s distributed motion planning algorithm [8]. However, we also considered the impact of communication performance, which reflects the reality.

Upon approaching the intersection, if the vehicle is ready to cross the boundary, it starts broadcasting current motion states to other vehicles. When it leaves the intersection completely, it will stop broadcasting and no longer participate in cooperative collision avoidance. Once receiving information from other vehicles, the vehicle will examine the current collision conditions and partition all involved vehicles into Collision Groups (CGs). As shown in Fig. 2, vehicle A, B and C are in one CG, where Collision Group is a collection of vehicles whose trajectories has at least one intersection with other members in the same CG. If vehicle A is the only member in the CG, which means that vehicle A wouldn’t collide with any other vehicle, vehicle A can keep its current speed crossing the intersection. Otherwise, vehicle A and its CG member vehicle B and C will have to adjust their acceleration rates to avoid the crash.
In this paper, the priority phase and non-priority phase are determined based on comparison of the total number of vehicles. We believe that vehicles on the priority approach do not need to change their current speeds. As illustrated in Fig. 2, vehicle A and C locate on the priority approach, and vehicle B locates on the non-priority approach. Therefore, vehicle B estimates the TTC with vehicle A and C, then slows down using the optimal deceleration rate. Until vehicle C completely leaves the intersection, meaning that no potential conflicts are expected at intersection area, and vehicle B begins to speed up and cross the intersection rapidly. Once unavoidable situations occur, such as system failure, the optimization algorithm cannot obtain optimal solutions for involved vehicles, the vehicles that locate on the non-priority approach brake urgently and wait before stop line and continue to analyze the collision states between vehicles, vehicles on the priority approach pass through the intersection as soon as possible. Besides, a global order should be generated through wireless communication among vehicles and they behave following the global order. In this paper, van der Pol model (Steven 1994), that is one of the reaction-diffusion systems, is used to make a temporal pattern.

![Fig. 2. Example of vehicle collision groups.](image)

### 3. Simulation Cases

In order to explore the performance of the CVC algorithm, an appropriate simulation environment is quite necessary. This paper developed a loosely integrated simulation framework incorporating VISSIM 4.3 [20] for microscopic level vehicular simulation and NS-3.20 for V2V wireless communication simulation. The optimal travelling strategy is obtained by external optimizer to solve a non-linear constrained optimization problem. The simulation framework was developed under the Visual C# programming environment and the optimization method was implemented using MATLAB’s script language. Note that the framework has the potential to simulate more complex traffic scenarios in future research.

#### 3.1. The Shuffled Frog Leaping Algorithm

It is generally understood that the evolution-based heuristic methods such as the shuffled frog leaping algorithm (SFLA) (Eusuff et al., 2003, 2006) or the genetic algorithm (GA) [7] have a better chance of finding an acceptable solution, although it may need longer computational time. We presented an improved SFLA to solve the optimization problem as (2), and obtain better solutions for this paper. The whole strategies are shown as follows:

1. **The first Jumping:**
   - If rand () \(\geq 0.7\)
     \[
     D_j = \text{rand}(\cdot) \times (p_b - p_a)
     \]
   - Else \(D_j = 0\)

2. **The second Jumping:**
   - \(\text{new } p_v = \text{old } p_v + D_j\)
Where: \( -D_{\text{max}} \leq D_j \leq D_{\text{max}} \)

The second Jumping:

\[ D_j = \text{rand}(\ ) \times (p_j - p_w) \]  

(8)

Where: \( t \in (1, 2, ..., m) \)

The third Jumping:

\[ \text{new}_p = LB + \text{rand}(\ ) \times (UB - LB) \]  

(9)

Where: \( D_j \) is the moving distance on component \( j \). \( m \) is the number of sub-memeplex, which contains \( n \) virtual frogs. \( \text{rand}(\ ) \) describes a random number between 0 and 1. \( D_{\text{max}} \) represents the maximum value of position changing allowed by frogs. \( P_b \) and \( P_w \) the best and worst frog’s position in the entire population, and \( P_t \) the best frog’s position in the sub-memeplex. The \( UB \) and \( LB \) are the upper and lower limitations of the searching space, respectively. For details regarding SFLA, we refer the reader to Eusuff et al. [4,5]. In order to prove the effectiveness of the improved SFLA, feasibility tests were conducted using three benchmark functions which include Sphere function, Rosenrock function and Schaffer’s f7 function. The results indicate that the improved SFLA has better acceptable solution than GA and SFLA for given benchmark functions, as shown in Table 1.

Table 1. The results of feasibility tests.

<table>
<thead>
<tr>
<th>Test functions</th>
<th>Optimization algorithm</th>
<th>Optimal solution</th>
<th>Average solution</th>
<th>Running time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sphere</td>
<td>GA</td>
<td>3.671E-3</td>
<td>1.364E-2</td>
<td>29.3s</td>
</tr>
<tr>
<td></td>
<td>SFLA</td>
<td>2.194E-4</td>
<td>1.251E-3</td>
<td>13.8s</td>
</tr>
<tr>
<td></td>
<td>ISFLA</td>
<td>0.000</td>
<td>4.275E-4</td>
<td>25.7s</td>
</tr>
<tr>
<td>Rosenbrock</td>
<td>SFLA</td>
<td>3.764</td>
<td>6.598</td>
<td>23.8s</td>
</tr>
<tr>
<td></td>
<td>ISFLA</td>
<td>1.954</td>
<td>2.768</td>
<td>16.3s</td>
</tr>
<tr>
<td></td>
<td>GA</td>
<td>3.252E-3</td>
<td>9.874E-3</td>
<td>27.4s</td>
</tr>
<tr>
<td>Schaffer’s f7</td>
<td>SFLA</td>
<td>8.341E-1</td>
<td>0.114</td>
<td>45.7s</td>
</tr>
<tr>
<td></td>
<td>ISFLA</td>
<td>7.473E-1</td>
<td>5.489</td>
<td>36.2s</td>
</tr>
<tr>
<td></td>
<td>ISFLA</td>
<td>3.561E-2</td>
<td>4.374E-1</td>
<td>29.8s</td>
</tr>
</tbody>
</table>

3.2. Simulation Scenarios

In this paper, we focus on an isolated urban intersection with one lane in each direction without turning movements. Simulations were carried out to validate and evaluate the potential benefits of algorithm. However, it should be pointed out that the algorithm was initially designed to be applicable for general intersections including multiple lanes and turning movements, even roundabouts.

For the performance evaluation, a total of 40 volume scenarios covering the volume to saturation ratios (v/s) of each lane, ranging from 0.1 to 0.7 were created, and 1800 vehicles per hour is the saturation flow used for simulation cases. The v/s ratios were converted to the corresponding volume to capacity ratios (v/c) with the optimal timing plans obtained from the Synchro program [9] for all volume scenarios. The algorithm performances were evaluated compared with the fixed timing signal control system under different conditions. With a 3600s simulation period, each scenario was replicated 40 times with different random seeds, thus, a total of 1600 simulations were implemented. Traffic simulation parameters are listed in Table 2.

3.3. Wireless Communications

Dedicated short range communication (DSRC) is dedicated entirely to serve the safety issues related to transportation. In other words, they act as a communication link to send or receive messages from other vehicles. In this paper, we applied wireless multi-hop broadcasts for exchanging safety critical information via DSRC. Communication-relevant parameters are summarized in Table 3. For simulating the physical and medium access
control (PHY/MAC) layers, we employed the 5.9GHZ frequency based on the IEEE 802.11p protocol [3], passed messages down to the MAC, and calculated the collision probability based on the cooperative awareness messages (CAMs) (2010) and basic safety messages (BSMs) (2011) received from other vehicles.

With respect to wireless communication, it was assumed that each vehicle in road network equipped with communication device. The vehicles within communication range continuously monitor the beaconing message and answer it. Once new vehicles successfully receive the beaconing message and answer, the new vehicles were registered as mobile nodes. Then, addresses were assigned to all involved nodes, which communicated with each other by the IP-based VANET. Considering the existence of transmission delay, the vehicles must have great computational ability, can perform a sequence of optimization process to obtain optimal acceleration/deceleration rates, assuming that the time required for performing the optimization process takes less than 100 ms. Meanwhile, vehicles need extract vehicular data from the neighboring vehicles approaching the intersection. The vehicular data mainly include: vehicle ID, speed, previous location, acceleration/deceleration rate, origination/destination and heading angle.

4. Results and Discussion

Two intersection control strategies have been conducted for simulation cases: 1) Cooperative vehicles control algorithm using connected vehicles strategies, and 2) Fixed timing control algorithm: total cycle is 80 seconds, with green phase 35 seconds, yellow phase 3 seconds, and all-red clearance 2 second.

The performance gains of CVC algorithm are summarized in Table 4. It is shown that the algorithm is able to significantly reduce the total stopped delay times as well as the average queue length, resulting in 96.3% and 33.1% of improvements. In addition, the total travel times was also improved by 21.8%. Such huge reduction may result from better utilization to intersection by avoiding frequent stop-and-go maneuver. The total numbers of vehicles that went across the intersection per hour were also raised up to 7.10%. Traffic flow felled into a congestion state with the increase of vehicles approaching the intersection, and vehicles had to stop and wait before the stop line until right-of-way are obtained under FTC environment. As a result, the corresponding total stopped delay times got longer, even reached 37.5h. The CVC algorithm avoided this situation by adjusting the accelerations of vehicles cooperatively. Therefore, the total stopped delay times in CVC system always maintained at a quite low level which was less than 1h, all else followed. Taking into consideration that the algorithm is designed to ensure vehicles cross
the unsignalized intersections avoiding the crash. Such visible improvements substantially demonstrate the promising benefits of the CVC algorithm, as shown in Fig. 3.

<table>
<thead>
<tr>
<th>Measure</th>
<th>FTC</th>
<th>CVC</th>
<th>Gain</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stopped delay times</td>
<td>13.5h</td>
<td>0.51h</td>
<td>96.3%</td>
</tr>
<tr>
<td>Queue length</td>
<td>46.3m</td>
<td>31.0m</td>
<td>33.1%</td>
</tr>
<tr>
<td>Travel times</td>
<td>37.6h</td>
<td>29.4h</td>
<td>21.8%</td>
</tr>
<tr>
<td>Throughputs</td>
<td>1356veh</td>
<td>1450veh</td>
<td>7.10%</td>
</tr>
</tbody>
</table>

Fig. 3. The performance gains of CVC algorithm.

Fig. 4 indicates the performance gains of the CVC algorithm under varying congestion conditions. We can clearly see that the gains of the total travel times are significant when v/c ratio is more than 0.7, almost no gains were observed when v/c ratio is less than 0.7. It means that under uncongested condition, when the optimal speeds are below the free flow speeds, the total travel times of CVC algorithm are longer than that of FTC system. But the CVC algorithm can more efficiently utilize the intersection with traffic volume getting congested, as illustrated in Fig. 4(a). The average queue length and total throughput are similar to the cases of the total travel times, but its critical v/c ratio is 0.8. When v/c ratio is approximately 1, the gain of average queue length reaches the maximum, and then decreases gradually. In addition, Fig. 4(d) shows the gain of total stopped delay times is always above 90%.

This paper has evaluated the performance of the CVC algorithm under perfect communication environment in previous section. But in the real world, the performance of V2V communication (e.g. packet drop rate and transmission delay) is influenced by many factors, such as transmission distance, the number and speeds of vehicles. Simulations were conducted to investigate mentioned situation, and we only take the average queue length as an example. The queue length increased from 30m to 90m while transmission delay ranged from 100ms to 300ms, as depicted in Fig. 5(a). The packet drop rate is associated with the packet length and transmission frequency. This paper sets the size of packet drop rate is 5%, 10% and 15% respectively. It was obviously observed in Fig. 5(b) that the average queue length sharply increased as packet drop rate changed, showing significant volatility. We can draw a conclusion that the performance of the CVC algorithm is directly influenced by wireless communication, becomes severe when V2V communication environment is imperfect.
5. Conclusions

This paper has presented a distributed CVC algorithm based on V2V communication. The algorithm is developed to ensure vehicles can cross the unsignalized intersection without colliding with other vehicles. The potential benefits of CVC algorithm over conventional fixed timing control system are evaluated under varying congestion conditions. Simulation results showed that the proposed CVC algorithm outperformed conventional control system in traffic safety and mobility. Moreover, this paper also considered the effect of imperfect communication on algorithm, and we found that communication performance directly affects the quality of algorithm. Hence, we need to make sure that V2V communication is reliable while running the algorithm.

In future work, the proposed distributed CVC algorithm will be tested in more complicated scenarios. A more realistic communication protocol between vehicles based on DSRC and SAE J2735 will be developed and implemented. Furthermore, we would like to improve the distributed vehicle control mechanism, and extend our simulation framework, making it have on-line processing ability.

References


