A Novel Virtual Traffic Light Algorithm Based on V2V for Single Intersection in Vehicular Networks

Longjiang Guo[1,2,3,4], De Wang[4], Peng Li[1,2,3], Lichen Zhang[1,2,3](✉),
Meirei Ren[1,2,3](✉), Hong Liu[2,3], and Ana Wang[2,3]

[1] Key Laboratory of Modern Teaching Technology, Ministry of Education,
Xi'an 710062, China
[2] Engineering Laboratory of Teaching Information Technology of Shaanxi Province,
Xi'an 710119, China
[3] School of Computer Science, Shaanxi Normal University, Xi'an 710119, China
{longjiangguo,zhanglichen,meiruiren}@snnu.edu.cn
[4] School of Computer Science and Technology, Heilongjiang University,
Harbin 150080, China

Abstract. For the vehicle passing priority, the existing virtual traffic light algorithm only considered the factor of the arrival time, a group of vehicles won't transfer the wayleave until they all pass completely. This method cannot achieve the maximum road traffic throughput capacity. This paper designs a queuing chain model and proposes a virtual traffic lights algorithm for Single Intersection (for short SVTL) based on the queuing chain model, considering comprehensively three parameters: the waiting time of each lane, the length of queue chain and the number of queues. The SVTL algorithm eliminates the phenomenon of waiting in vain basically, saves the waiting time of the vehicle, and avoids unnecessary parking conditions. The simulation shows that the SVTL shortens the average waiting time when the traffic density is high, compared with the Common Traffic Lights (CTL) and the existing virtual traffic light algorithm DVTL, the average speed is increased by 36.9% and 16.2% respectively.

Keywords: Intelligent transport system · Internet of Vehicles · Virtual traffic lights · V2V

1 Introduction

In recent years, large-scale urban traffic congestion has taken place in more and more cities for a long time, especially in morning and evening rush hours and bad weather period, which also increases rapidly in the second-tier and third-tier city. According to a report from Amap's China Major Urban Traffic Analysis in 2017, the big data analysis of 364 cities and highway networks nationwide shows

L. Guo, D. Wang and P. Li—Contributed equally to this work.

Y. Li et al. (Eds.): COCOA 2019, LNCS 11949, pp. 235–251, 2019.
https://doi.org/10.1007/978-3-030-36412-0_19

that one third of the cities are threatened by congestion during peak hours in China. More than half of city's commuter peaks are in a slow state, and 35% cities are in a slow state during the flat hump period.

There are usually three common ways to alleviate traffic congestion: reducing the number of vehicles, increasing infrastructure capacity (widening roads, increasing road network density, optimizing road structure, etc.) and increasing throughput at intersections. The throughput bottleneck at the intersection has a lot to do with the management of the intersection. Efficient traffic signal control is an effective and economical solution.

Internet of Vehicles (IoV) is an important part of the Intelligent Traffic System (ITS). Communication between vehicles and vehicles or infrastructure (Vehicle to everything, V2X) can significantly improve the safety, manageability and transport efficiency of the transportation system, and reduce energy consumption. In the field of traffic control, the Internet of Vehicles provides a more efficient means for the realization of virtual traffic signal control system solutions. Virtual traffic lights rely on the Internet of Vehicles environment to negotiate an appropriate road-rights distribution method through information exchange among vehicles, which provides a new direction for improving the throughput and efficiency of road junctions.

The contributions of this paper are given as follows.

1. This paper proposed a vehicle queuing chain model in the Internet of Vehicle, this model is closer to the application scenario of urban multi Lane mode. The common three-lane queuing chain model is adopted in this paper.
2. Based on the queuing chain model, a data collection algorithm between vehicles is proposed in this paper. The information queue organized by this algorithm is just like the real vehicle queue on the road, which is convenient for data collection and maintenance.
3. This paper proposes a virtual traffic lights algorithm for Single Intersection, to calculating the priority of traffic with the three parameters which are the waiting time of each lane, the length of queue chain and the number of queues. According to the intersection conflict table, find out the non-conflicting lanes, and let these lanes enter the green time at the same time to increase the road throughput. This algorithm eliminates the phenomenon of waiting in vain basically, save waiting time and avoids unnecessary parking.
4. The simulation shows that the SVTL shortens the average waiting time when the traffic density is high, compared with the Common Traffic Lights (CTL) and the existing virtual traffic light algorithm DVTL, the average speed is increased by 36.9% and 16.2% respectively.

2 Related Work

In vehicular networks, there are many interesting research fields. Some researchers who have focused on intersection-based forwarding protocol, privacy vehicular data and capacity analysis for vehicular networks. Xiong et al. have proposed a privacy-preserving approach to protect vehicular camera data [24].

Guan et al. have presented an intersection-based forwarding protocol for vehicular networks [7]. Huang et al. studied the theory based capacity analysis for vehicular networks [9].

There are quantities of researches about traffic signal phase position algorithm, one of which is based on traffic history data, SCOOT [19] is the widely used traffic signal timing system, which detects traffic flow on the upstream road segment and predicts traffic flow to the intersection to make some pre-adjustments. Although the control signal can be sent in real time, it sacrifices part of the accuracy in order to speed up the response of the system and cannot be optimized. Australia's SCATS system [11] constructs a traffic signal task plan based on historical data which collects real-time traffic flow by arranging sensors at the intersection stop line, and then makes plans according to the flow data. But this kind of system is not real-time; it detects the traffic flow and selects one of the best options from the existing solution, and then deploys the solution at the intersection. The method is less accurate and reliable. It is widespread that using wireless sensor nodes to monitor and direct traffic [14]: by establishing sensor network topology to monitor traffic flow and improve driving environment, use the acquired traffic flow design algorithm to avoid congestion, and use particle swarm optimization (PSO) method [10] to develop sensor placement position to optimize sensor coverage. The WSN network is used to obtain the traffic flow of the upstream road segment, and the fuzzy controller is used to estimate the traffic volume of the future intersection, thereby formulating the mission plan of the traffic signal [3].

Some researchers have studied the joint task formulation of multiple traffic junctions in cities. For example, some traffic control algorithms have adopted the particle swarm optimization algorithm [5], genetic algorithm [2], neural network computing model [4,21], machine learning [12,22] and other methods. He et al. combined particle swarm optimization algorithm with neural network model [8], using evolutionary particle swarm optimization algorithm to deal with multiple intersection traffic flow model optimization problems, and using artificial neural network model to learn multi-junction traffic flow control strategy. And in the simulation environment, the experimental results of this method are verified to be better than the single method. Although these methods take the overall traffic situation into account, they require a lot of calculation time and cannot meet the real-time performance, and the data used for calculation are all out-of-date. When the traffic flow fluctuates greatly, the result of historical data can not meet the real-time requirement.

Another solution is based on the Internet of vehicles, in which vehicles can communicate via V2I and V2V to monitor real-time information of vehicles. Some researchers have proposed that the formation of an adaptive traffic control system [20] by using the short-distance communication of vehicles, using a very short path (1–2 hops) between the vehicle and the signal to communicate. Wu et al. established a two-layer pipeline model using the V2I communication of the Internet of Vehicles system at the intersection [23] which is used to accurately detecting vehicle information and assign different weights to vehicles according to the distance from the vehicle to the intersection. The adaptive skip signal control method uses the principle of on-demand distribution to prevent the vehicle from waiting at the idle intersection. Another mode is to use the vehicle terminal to collect traffic data and send them to the traffic signal [25], and the result is calculated by the remote computer, which does not take the delay of communication into account between the traffic signal and the remote computer.

The introduction of Virtual Traffic Lights (VTL) [6] opened the other direction into intelligent traffic lights. In the Internet of Vehicles, virtual traffic lights are used to solve the traffic congestion problem. Instead of any street infrastructure, the direct communication between vehicles is used to negotiate the right-of-way assignment rules, and the driver is informed by the vehicle display device when to pass the intersection.

The Internet of Vehicles for IEEE802.11p still takes a long time to become popular. Some researchers use the Wi-Fi Direct of smartphones to communicate [13,16] to implement the virtual signal system, but the authors only implemented the system in the laboratory, without considering that Wi-Fi Direct could not communicate efficiently while the actual vehicle was running.

Some researchers have proposed V2V-based distributed algorithm named DVTL [1] to implement the basic model of the virtual signal light mentioned in [6]. When the vehicle information is exchanged, the location and the vehicle basic information are transmitted by broadcasting, and the "leader" election information is exchanged by using the unicast method. But DVTL has some deficiencies in the formulation of traffic signals: the only factor that the arrival time at the intersection is taken into account when setting the priority of the traffic, without including the number of queues and other factors; There is no strict regulation on the passage time of each lane, and the right of way will be transferred to the next lane only if the group of vehicles completely pass the intersection in the lane with the right of way; The established lane models are all single-lane models, but in reality, urban roads are mostly two or more lanes.

In this paper, a queuing chain model is proposed. Based on the model, V2V communication is used to count and collect the vehicle information. According to the collected number of vehicles, the queue length and the waiting time, the traffic signal phase position of the intersection are determined by the vehicle self-organization.

3 System Model

3.1 VTL System Model

Virtual traffic signal system requires vehicles to communicate and coordinate with each other to formulate traffic signal phase position. When the system is running, the system endows the vehicle with different identities to complete different tasks that required by the signal phase position according to different location of the vehicle.

The identity of vehicles in the system are defined as follows:

Definition 1 ***Free node.*** *The identity of the vehicle is a free node when the vehicle passes the intersection but does not drive beyond the threshold.*

The threshold is a logical line. The distance between the threshold and the nearest intersection is usually not more than a third of the distance between two adjacent intersections.

Definition 2 ***Obey node.*** *The vehicle passes the driving threshold, but it is not the nearest car to the next intersection within the lane communication range.*

Definition 3 ***Group aggregation node.*** *The vehicle passes the driving threshold and it is the nearest car to the next intersection within the lane communication range.*

Definition 4 ***The total aggregation node.*** *Before entering the intersection area, all of the group aggregation nodes negotiate and select a vehicle from themselves as the total aggregation node to calculate the traffic signal phase position.*

The VTL module needs to obtain its own information such as location, time and vehicle status, and broadcasts the data to other intersection vehicles through DSRC or LTE-V2X at the intersection. According to the received data, it is processed and handed over to the VTL group node decision module, which is used to elect the group node of each lane. Then, the group nodes share the lane data, and an aggregation node is elected. The vehicle calculates the phase of the signal and broadcasts the result to the intersection vehicle. The vehicles at the intersection pass the intersection in an orderly manner according to the result.

3.2 Intersection Conflict Model

We can deduce the conflict situation. These conflicts should be taken into account when we formulate the phase rules of the intersection. Table 1 lists the conflict situation at the intersection, L indicates the left turn, the F is straight, 0 represents the conflict, and the 1 indicates no conflict. A, B, C, and D, indicate the north, east, south and west of a crossroad respectively.

3.3 Vehicle Queuing Model

There are significant differences in its size and initial acceleration when vehicles are in queues at intersections due to different vehicle models. When arranging traffic lights according to the queuing of vehicles at intersections, it is necessary to consider the case that new queued vehicles are added. Therefore, the traffic signal phase positions formulation should take the impact of the vehicle model and the new queued vehicle into account. Although the actual vehicle models are complicated, there are no obvious differences in the acceleration for same type of vehicle. Therefore, in the experiment, three models were used to simulate different vehicle lengths and accelerations to eliminate the effects caused by vehicle type. In addition, the queuing of the vehicle can be modeled using Markov

Table 1. Intersection conflict table

	AL	AF	BL	BF	CL	CF	DL	DF
AL	1	1	0	0	1	0	0	1
AF	1	1	1	0	0	1	0	0
BL	0	1	1	1	0	0	1	0
BF	0	0	1	1	1	0	0	1
CL	1	0	0	1	1	1	0	0
CF	0	1	0	0	1	1	1	0
DL	0	0	1	0	0	1	1	1
DF	1	0	0	1	0	0	1	1

queuing theory, which estimates the possibility of joining the vehicle in turn, and develop the traffic light phase.

According to the defined vehicle attributes and the vehicle type distribution rules at the intersection, the time required for vehicles to pass through the intersection can be obtained.

The time required for a car to pass through an intersection can be divided into: driver response delay and vehicle start delay T_d, vehicle acceleration time T_a and vehicle uniform speed time T_v. The total time the motorcade passes through the intersection is equal to the time when the last car completely crossed the intersection. The time for the last car to pass the intersection is $T_{last} = T_a + T_v + T_d$, vehicle acceleration time T_a and vehicle constant speed time T_v depend on the distance that the last car will travel.

The last vehicle crossing the intersection needs to travel the distance:

$$\mathrm{S}_{last} = \frac{1}{2}a{T_a}^2 + vT_v = \sum_{i}^{N} L_i + S_{cr} + (N-1) \times \mathrm{L}_{gap}(aT_a \leq v) \tag{1}$$

where, N is the number of vehicles queuing at the junction, S_{cr} is the length of the intersection, L_i is the length of the vehicle, L_{gap} is the average length of the vehicle interval, a is the average acceleration of the vehicle, v is the maximum speed allowed by the vehicle, and $aT_a \leq v$, when $aT_a < v$, $T_v = 0$. T_d are all driver response delays and vehicle startup delays in the driveway, where T_{idelay} is the i^{th} driver response delay and vehicle startup delay:

$$\mathrm{T}_d = \sum_{i=1}^{N} T_{idelay} \tag{2}$$

In conclusion, when the last vehicle passes the intersection, the time T_{last} can be obtained, as shown in Eq. 3.

$$\mathrm{T}_{last} = \begin{cases} \sqrt{\frac{2S_{last}}{a}} + T_d & aT_a < v \\ \frac{S_{last} - \frac{v^2}{2a}}{v} + \frac{v}{a} + T_d & aT_a = v \end{cases} \tag{3}$$

By calculating the T_{last} of each lane and obtaining the lane combination of green light according to the conflict Table 1, the total time of vehicles passing through the intersection at the current intersection is minimized.

4 Data Collection Algorithm Based on Queuing Chain

4.1 Message Definition

Three kinds of data packets are used for vehicle communication in the system: information packet, notification packet and control packet. There are two types of information packets: information packets for a single car and group aggregation information packets. Information packets are used to convey their own information to surrounding vehicles, which is the basis for the aggregation node selection. The notification data packet is used to send a notice to the surrounding vehicles, to inform the aggregation node of the selection result and to inform the vehicle that the vehicle is about to enter the intersection. The function of controlling data packets is to send the phase data of the traffic lights to the waiting vehicles at the intersection.

The information packet contains the following information: Information type, Vehicle ID, Vehicle Length, Distance, Lane Number, Direction and Time Stamp. The type of information indicates which type the packet belongs to. The vehicle ID is the only code that identifies the vehicle identity. The length of the vehicle, the distance from the vehicle to the intersection and the lane number of the vehicle are used to calculate the traffic light phase. The direction indicates the direction that the vehicle will be driving at the intersection.

The group aggregation information packets represent the aggregation information of the waiting vehicles in a lane for determining the total aggregation node and calculate the traffic signal phase position.

The notification packet is sent by the aggregation node to inform other vehicle aggregation nodes identity changes, and the packet has synchronous timestamp function.

The transfer notification packet is used to transfer the queuing chain of the group aggregation node. The packet contains the data information collected from all vehicles.

The control packet contains the allocation of road rights of all vehicles in all lanes of the intersection. The start time, end time and the sequence of signal lights constitute a set of signal lamp phases, and the corresponding signal sequences of each lane are executed in the starting time and ending time.

4.2 Queuing Chain Model

In a real traffic environment, the state of the vehicle is constantly changing, and behaviors, such as lane change and overtaking, are frequent. In the multi-lane case, the group aggregation node vehicle must maintain the member information in the group, and the data structure of the linked list is very suitable for the maintenance of the queue chain. The formation of the queuing chain requires two steps: the formation of a queuing chain and the maintenance of a queuing chain.

Queuing chain formation: Before the vehicle travels past the threshold through the intersection, if the vehicle does not receive the aggregation node notification packet sent by the preceding vehicle, the vehicle changes the identity to the group aggregation node, and periodically broadcasts the notification packet to each channel. If the notification packet of the lane group aggregation node is received before the threshold is exceeded, the vehicle becomes a obey node, and the obey node periodically sends its own information to the group aggregation node to maintain the queued chain order.

Queueing chain maintenance:

Case 1: The group aggregation node receives the vehicle information sent by the own lane obey node. If the obey node is in front of the vehicle, then the group aggregation node hands over the group aggregation node identity to the obey node, otherwise the obey node is added to the queuing chain according to the Distance data in the information.
Case 2: The group aggregation node vehicle receives notification information of other vehicles in the same lane with the group aggregation node. If the front group aggregation node does not enter the decision stage of the total aggregation node, the group aggregation node in the rear group transfers the group aggregation node identity to the queue chain order first. If the front group aggregation node has entered the decision stage of the total aggregation node, then it maintains the identity of the group aggregation node.
Case 3: The obey node receives the notification information of the plurality of own lane group aggregation nodes, and selects the first group aggregation node to send the vehicle information.
Case 4: The group aggregation node leaves the queue and hands over the identity to the first car in the rear before leaving.
Case 5: The group aggregation node has not received certain obey node information in the queuing chain for a long time, and deletes the node from the queuing chain.
Case 6: The vehicle becomes a free node through the intersection.
Case 7: After the group aggregation node enters the intersection area, the identity is no longer changed, and new nodes are not accepted to join.

4.3 Data Collection Algorithm Based on Queue Chain

The vehicle identity in the virtual traffic light system is divided into the free node, the obey node, the group aggregation node and the total aggregation node. The queuing chain is composed of the group aggregation node as the chain head and the obey node as the chain body. The formation and maintenance of the queuing chain is information-driven, and the formation and maintenance process of the queuing chain is the collection process of vehicle information data. The head of the chain node (group aggregation node) has the right to broadcast in all channels, and other nodes can only communicate with the sink node directly in one-hop mode within the limited channel of the lane in which they belong to. In order to reduce communication, vehicles are allowed to send messages only after they have passed the threshold line. The obey node can only send information within the limited channel, and the group aggregation node can broadcast notification information to the whole network periodically, and control packet broadcasting only in the channel of the queuing chain in the lane.

At the intersection, according to the lane of the vehicle and the direction (left turn, right turn, straight ahead) in which the vehicle is moving, the vehicle determines which queuing chain the vehicle belongs to. According to the queuing chain model, the aggregation node of the queuing chain is the queue head, the group aggregation node of the left lane is the first left-turn vehicle, if the first car runs straight forward, it will be classified as the middle lane and the identity is the middle lane obey node. The group aggregation node of the middle lane is the first vehicle runs straight forward; otherwise it is classified as the obey node of the corresponding lane. The group aggregation node of the right lane is still the first vehicle runs straight forward. If the first vehicle is a right-turn vehicle, it is a free node and not included in the queuing chain.

Since at the intersection which queuing chain the vehicle belongs to is determined by the lane of the vehicle and the direction in which the vehicle is moving. Therefore, the lane change will have an impact on the queue chain of the lane in which it is located and the queue chain to be entered. There are two identities in a queuing chain: the first node of the queue chain (group aggregation node) and the obey node.

When a group aggregation node changes lanes, it will cause changes to the group aggregation node. Therefore, when a lane change occurs, the identity of the aggregation node is handed over to the vehicle in the original lane list that matches the identity of the aggregation node (handed over the entire queue chain). That is, the first left-turning vehicle in the left lane, the first vehicle runs straight forward in the middle lane or in the right lane. Upon completion of the identity transfer, it becomes obey node.

The selection of the group aggregation node is the chain head vehicle of each lane queue chain. The selection and maintenance algorithm of the group aggregation node is essentially to maintain the queue chain of each lane. The queue chain satisfies two conditions. Firstly, the vehicle on each chain must pass the intersection in the green light phase time of the lane. Secondly, the group aggregation node on the chain must be the first vehicle on the lane queue chain. Because each lane has a corresponding traffic signal, the first condition can ensure the accuracy of the time of passing the intersection in the lane, that is, to ensure the phase accuracy of the traffic light at the intersection. The second condition ensures that the group aggregation node is the first vehicle to enter the intersection area, and the total aggregation node vehicle is selected from the group aggregation node, so that the communication is as unobstructed as possible to ensure communication quality.

Through the above rules, each lane constitutes a respective lane queuing chain, and the first node of the queuing chain is a group aggregation node, and the information of the queuing chain vehicle is collected in the group aggregation node vehicle. The total aggregation node is elected according to the information of the vehicles in the queue chain between the lanes. The purpose of the total aggregation node is to replace the physical traffic lights at the intersection to direct traffic. Therefore, in a custom traffic light phase cycle, the queue of the total aggregation node should pass through the intersection as late as possible. The system follows the first-come first-pass and long-queue priority principles to dynamically adjust the transit time and traffic order of each lane. The total aggregation node is chosen only refer to two indicators: the arrival time and the queue length of the lane. The time reaching the intersection can be used to ensure fairness. Vehicles arriving at the intersection early have the right to pass through the intersection first. The length of the queue chain represents the degree of congestion in the lane. Allowing longer queues to pass first can appropriately relieve road pressure.

5 Virtual Traffic Light Based on V2V for Single Intersection

The phase of traffic lights generated in the virtual intelligent traffic light system is real-time. Every time a group of vehicles pass through the intersection, the corresponding phase of the signal lights will be formulated according to the vehicles in each lane. The signal lights generated by each group are customized, so it can greatly reduce the phenomenon of waiting in vain and improve the throughput of the intersection.

A set of traffic light phases consists of two elements: the order of travel and the travel time. In other word, to formulate a set of traffic light phase is to formulate a combination of a set of travel time and traffic order. The goal is to make all the vehicles at the intersection pass through the intersection as soon as possible. Calculating the transit time firstly requires calculating the traffic distance. According to the calculation formula defined by the vehicle queuing model, the distance that the last vehicle in each lane needs to pass through the intersection is obtained. The distance of the last car in the left lane through the intersection is S_{Llast}, according to the joint calculation of the middle lane and the right lane we can obtain that the distance of the last car passing through the intersection is S_{Mlast}. The distances obtained are brought into Eq. 3 to determine the time when the vehicles pass through the intersections in each lane. In order to ensure the fairness of each group, the transit time is less than 90 s.

The order of passage for each lane should consider three factors: waiting time, number of queues, and queue length. The large number of queues means that the traffic volume of the lane is large and the traffic pressure is high, so it should be preferred. (The number of queues in the middle lane is the sum of the straight movement vehicles in the middle and right lanes). In order to ensure the fairness of traffic, waiting time should have a higher priority and the longer the waiting time, the higher the priority. The length of the queue and the number of queues affect the time passing through the intersection. According to the traffic conflict list at the intersection defined in the vehicle queuing model, after calculating the transit time of each lane, it needs to make overall consideration, so that the vehicles in multiple lanes can pass through the intersection at the same time without conflict.

The priority of the pass order is determined by the wait time and the number of queues. The weights of the two factors shown in Eq. 4 are divided according to the waiting time gradient. The weight of the waiting time in 0–30 s is 0.4, the weight increases linearly within 30–90 s, and the weight of the waiting time exceeds 90 s is 1. That is, when the vehicle waiting time exceeds 90 s, the number of queues is not referred to when calculating the priority of the traffic order.

$$W_T = \begin{cases} 0.4 & 0 \leq t < 30\,\mathrm{s} \\ 0.01t + 0.1 & 30 \leq t < 90\,\mathrm{s} \\ 1 & t > 90\,\mathrm{s} \end{cases} \tag{4}$$

The waiting time and the number of queues are two kinds of metrics, and the two types of data need to be normalized separately so that the two types of data can be added to generate a priority. Normalize the waiting time, using the minimum-maximum normalization method, the minimum value is 0, and the maximum value is 90 s (After waiting for more than 90 s, regardless of the impact of the number of queues on the priority, the priority may be greater than 1), so the time-based priority is:

$$P_T = \frac{T - T_{MIN}}{T_{MAX} - T_{MIN}} = \frac{T}{90} \tag{5}$$

Then, the data of the number of queues is normalized still using the minimum maximum normalization method. The minimum value is 1 (0 means no waiting for the vehicle, does not participate in the calculation), and the maximum value is the maximum number of vehicles that can pass the intersection at an average of 60 s without congestion. In the simulation experiment, 60 s can pass through 25 vehicles. The priority based on the number of queues is:

$$P_N = \frac{N - N_{MIN}}{N_{MAX} - N_{MIN}} = \frac{N - 1}{24} \tag{6}$$

In summary, the priority of a lane passing through the intersection is a weighted sum of the priority PY based on the waiting time and the priority PN based on the number of queues, as shown in Eq. 7:

$$\mathrm{P} = W_T P_T + (1 - W_T) P_N \tag{7}$$

Calculate the priority of each lane according to Eq. 7, and select the lane with the highest priority to pass first (Long queues with the same priority take precedence). Lookup Table 1 finds that the lanes that do not conflict are added to the group of traffic phases by priority. If the time-taken is less than the highest priority lane supplemented by the next non-collision lane, if the time-taken is greater than the highest priority lane, set the time to be the same as the highest priority lane. So reciprocating until all vehicles in the group have completed the transit time, then the next group of vehicle traffic phases are turned on.

In summary, the basic steps for calculating the phase of a traffic light by the SVTL system are:

Step 1: Determine the identity of the total aggregation node vehicle. The group aggregation node vehicles elect the total aggregation node according to their respective queuing chain information. The total aggregation node needs to complete the formulation work of the traffic light.
Step 2: calculating the transit time, using the queue chain information collected by the group aggregation node of each lane to calculate the transit time required by the vehicles in each lane.
Step 3: data normalization, using the two parameters of waiting time and queuing number to determine the pass priority, so the two parameters need to be normalized.
Step 4: Calculate the priority and calculate the priority of each lane according to Eq. 6.
Step 5: Develop the intersection traffic light phase, use the priority calculated in the fourth step, select the first green light phase, and then select the phase that does not conflict with the phase and has the highest priority according to the conflict table and priority. Co-execute the first green light.
Step 6: Distribute the results. The total aggregation node first unicasts the results to the group aggregation node, and then the group aggregation node broadcasts to all the nodes in the group.

6 Simulation Results

In this section, we use the OMNeT++ [15], SUMO [18] and VEINS [17] frameworks to run the simulation. The simulation runs the Common Traffic Lights (CTL), the Distributed Virtual Traffic Lights (DVTL) [1] system and the single virtual traffic lights (SVTL) under the same experimental environment. Compare the average parking time and average driving speed of the three algorithms.

6.1 Experimental Parameter Setting

Road network environment setting: There are 49 roads in the road network, each road is 300 m long, the road width is 20 m, the road is six-way, the road max speed is 19.5 m/s, the blocking density is 300 pcu/km, the three lanes at one direction intersection are left turn lane, straight lane and right turn or straight lane; a total of 64 intersections, including 36 intersections, 24 T-junctions and 4 right-angle intersections. Set the right turn to be unrestricted, left turn and straight line must be exercised according to the signal indication.

Vehicle Setting: In the simulation process, a total of 9000 vehicles were built to participate in the simulation experiment. 9000 vehicles entered the road network at the same time interval according to the serial number. The vehicle has three models, the lengths are 4 m, 6 m and 12 m respectively, and the acceleration is set to 5 m/s^2. The parking gap is 1.5 m. In the case where the starting and ending of 9000 vehicles remain unchanged, the tools provided by SUMO are used to generate three sets of routes, and three sets of routes can be used to perform three experiments separately.

Network Settings: use perfect network status, that is, no packet loss, no network conflicts. The vehicle communication radius is 100 m, strong penetration mode. The communication protocol uses the IEEE802.11P protocol. Use the same road network configuration and vehicle settings to run CTL, DVTL and SVTL respectively. Compare the differences between the three systems and set different traffic flows to simulate the difference in performance of traffic lights under different traffic flows.

6.2 Vehicle Average Parking Time

Figure 1 shows the average parking time of the vehicle under different traffic flows. In the case of CTL, when the number of vehicles in the road network is between 200 and 1000, the parking time increases steadily, but even in the case of only 200 vehicles in the road network, the average waiting time is still 80 s. DVTL and SVTL have an average parking time of less than 2 s in the case of 200 vehicles in the road network, but as the number of vehicles increases, the average parking time also rises rapidly. This is because DVTL and SVTL basically eliminate the phenomenon of waiting in vain, but as the number of vehicles increases, vehicles waiting to pass at various intersections began to increase, resulting in a faster growth rate of average waiting time. As the density of vehicles increases, the parking time under DVTL grows faster than SVTL. This is because DVTL only considers the order of arrival at the intersection when setting priorities, so that the intersection traffic cannot be optimal.

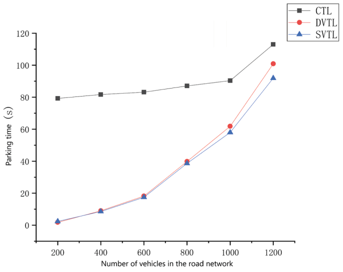

Fig. 1. Average parking time

6.3 Vehicle Average Driving Speed

Figure 2 shows the average driving speed of the vehicle under the control of three traffic lights under different traffic flows. Similar to the average parking time in the previous section, CTL performed poorly in a sparsely traffic environment, and the average speed was pulled down due to the large number of waiting in vain conditions. DVTL and SVTL basically eliminate the waiting in vain, so that its speed can better reflect the road congestion. From the flow density formula, it can be known that the vehicle density in the road will affect the speed of the vehicle. That is to say, when the traffic volume is large, the factor limiting the average speed of the vehicle is mainly the vehicle density rather than the waiting time of the traffic signal.

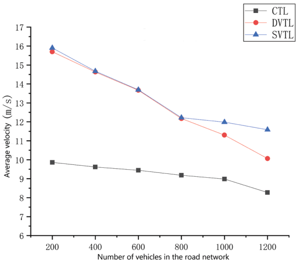

Fig. 2. Average driving speed

Figure 3 shows a comparison of average speeds in an average of 1200 vehicles participating in traffic at the same time. The average speed of the vehicle under CTL is 8.4 m/s, and the average speed of the vehicle under DVTL is about 9.9 m/s. The total average speed of the vehicle under the SVTL is approximately 11.5 m/s. Compared with CTL and DVTL, the average speed increased by 36.9% and 16.2%, respectively. In terms of speed distribution, the average speed of vehicles in a CTL environment is concentrated at 4–12 m/s, in the DVTL environment is concentrated at 8–12 m/s and in the SVTL environment is concentrated at 8–16 m/s. It is clear that the average speed that the SVTL can achieve at this vehicle density is higher than the other two algorithms, and the vehicles with an average speed of more than 16 m/s are much higher than the other two algorithms.

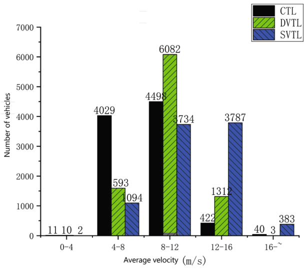

Fig. 3. Statistical histogram of average speed in congested state

7 Conclusion

This paper defines the queuing chain model and designs the algorithm for generating and maintaining the queuing chain. The establishment and maintenance of the queuing chain can assist the group aggregation node vehicles to collect and maintain the information of the vehicles in the lane, and ensure the accurate and real-time information. Proposed a real-time single virtual traffic lights (SVTL) phase algorithm. Experiments show that compared with the Common Traffic Lights method, this algorithm has better effect no matter how traffic density changes, the average speed of the vehicle is higher, and the parking waiting time is lower. Compared with the DVTL algorithm, when the traffic density is small, the performance of the two algorithms is not much different. When the density of the traffic vehicle is gradually increased, the performance of the DVTL is lower than that of the SVTL. In the subsequent research, multiple intersection factors will be considered at the same time, making the algorithm more suitable for real traffic environment.

Acknowledgement. This work is partly supported by the National Natural Science Foundation of China with No. 61877037, the Fundamental Research Funds for the Central Universities of China with No. GK201903090 and GK201801004, and the National Key R&D Program of China with No. 2017YFB1402102.

References

1. Bazzi, A., Zanella, A., Masini, B.M.: A distributed virtual traffic light algorithm exploiting short range V2V communications. Ad Hoc Netw. **49**, 42–57 (2016)
2. Chin, Y.K., Yong, K., Bolong, N., Yang, S.S., Teo, K.T.K.: Multiple intersections traffic signal timing optimization with genetic algorithm. In: 2011 IEEE International Conference on Control System, Computing and Engineering (ICCSCE), pp. 454–459. IEEE (2011)
3. Collotta, M., Bello, L.L., Pau, G.: A novel approach for dynamic traffic lights management based on wireless sensor networks and multiple fuzzy logic controllers. Expert. Syst. Appl. **42**(13), 5403–5415 (2015)

4. De Oliveira, M.B., Neto, A.D.A.: Optimization of traffic lights timing based on multiple neural networks. In: 2013 IEEE 25th International Conference on Tools with Artificial Intelligence (ICTAI), pp. 825–832. IEEE (2013)
5. El Hatri, C., Boumhidi, J.: Traffic management model for vehicle re-routing and traffic light control based on multi-objective particle swarm optimization. Intell. Decis. Technol. **11**(2), 199–208 (2017)
6. Ferreira, M., Fernandes, R., Conceição, H., Viriyasitavat, W., Tonguz, O.K.: Self-organized traffic control. In: Proceedings of the Seventh ACM International Workshop on VehiculAr InterNETworking, pp. 85–90. ACM (2010)
7. Guan, X., Huang, Y., Cai, Z., Ohtsuki, T.: Intersection-based forwarding protocol for vehicular ad hoc networks. Telecommun. Syst. **62**(1), 67–76 (2016)
8. He, L., Fu, C., Yang, L., Tong, S., Luo, Q.: Coordinated real-time control algorithm for multi-crossing traffic lights. In: 2014 10th International Conference on Natural Computation (ICNC), pp. 128–133. IEEE (2014)
9. Huang, Y., Chen, M., Cai, Z., Guan, X., Ohtsuki, T., Zhang, Y.: Graph theory based capacity analysis for vehicular ad hoc networks. In: 2015 IEEE Global Communications Conference, (GLOBECOM) 2015, San Diego, CA, USA, 6–10 December 2015, pp. 1–5 (2015)
10. Kulkarni, R.V., Venayagamoorthy, G.K.: Particle swarm optimization in wireless-sensor networks: a brief survey. IEEE Trans. Syst. Man Cybern. Part C (Appl. Rev.) **41**(2), 262–267 (2011)
11. Martin, P.T.: Scats, an overview. In: Transportation Research Board Annual Meeting (2001)
12. Mousavi, S.S., Schukat, M., Howley, E.: Traffic light control using deep policy-gradient and value-function-based reinforcement learning. IET Intell. Transp. Syst. **11**(7), 417–423 (2017)
13. Nakamurakare, M., Viriyasitavat, W., Tonguz, O.K.: A prototype of virtual traffic lights on android-based smartphones. In: 2013 10th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON), pp. 236–238. IEEE (2013)
14. Nellore, K., Hancke, G.P.: A survey on urban traffic management system using wireless sensor networks. Sensors **16**(2), 157 (2016)
15. OMNet++: https://omnetpp.org/
16. Perry, A., Verbin, D., Kiryati, N.: Crossing the road without traffic lights: an android-based safety device. In: Battiato, S., Gallo, G., Schettini, R., Stanco, F. (eds.) ICIAP 2017. LNCS, vol. 10485, pp. 534–544. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-68548-9_49
17. Sommer, C., German, R., Dressler, F.: Bidirectionally coupled network and road traffic simulation for improved IVC analysis. IEEE Trans. Mob. Comput. **10**(1), 3–15 (2011)
18. SUMO: http://sumo.sourceforge.net/
19. Unit, Traffic Advisory: The ‘SCOOT’ urban traffic control system. Traffic Advis. Leafl. **7**, 99 (1999)
20. Varga, N., Bokor, L., Takács, A., Kovács, J., Virág, L.: An architecture proposal for V2X communication-centric traffic light controller systems. In: 2017 15th International Conference on ITS Telecommunications (ITST), pp. 1–7. IEEE (2017)
21. Wang, D., Bao, H., Zhang, F.: CTL-DNNet: effective circular traffic light recognition with a deep neural network. Int. J. Pattern Recognit. Artif. Intell. **31**(11), 848–854 (2017)

22. Wang, H., Liu, J., Pan, Z., Takashi, K., Shimamoto, S.: Cooperative traffic light controlling based on machine learning and a genetic algorithm. In: 2017 23rd Asia-Pacific Conference on Communications (APCC), pp. 1–6. IEEE (2017)
23. Wu, L.B., Nie, L., Liu, B.Y., Wu, N., Zou, Y.F., Ye, L.Y.: An intelligent traffic signal control method in VANET. Jisuanji Xuebao/Chin. J. Comput. **39**(6), 1105–1119 (2016)
24. Xiong, Z., Li, W., Han, Q., Cai, Z.: Privacy-preserving auto-driving: a gan-based approach to protect vehicular camera data. In: 2019 IEEE International Conference on Data Mining, (ICDM) 2019, Beijing, China, 8–11 November 2019, pp. 1–12. IEEE (2019)
25. Younes, M.B., Boukerche, A.: Intelligent traffic light controlling algorithms using vehicular networks. IEEE Trans. Veh. Technol. **65**(8), 5887–5899 (2016)