Evaluations of Intelligent Traffic Signal Control Algorithms under Realistic Landmark-based Traffic Patterns over the NCTUns Network Simulator

Shie-Yuan Wang and Yu-Wei Li
Department of Computer Science
National Chiao Tung University, Taiwan
Email: shieyuan@cs.nctu.edu.tw

Abstract—In this paper, we use the NCTUns network simulator to study intelligent traffic signal control algorithms. NCTUns is both a microscopic traffic simulator and a network simulator. We imported the real-life map of a district of the Taipei city into NCTUns to let simulated vehicles move on the roads of the imported map. Then, we used a new and unique feature of NCTUns – each simulated vehicle moves towards its assigned landmark on the map as its destination, to create more realistic traffic patterns in the studied district. With these realistic settings, we first studied an intelligent traffic signal control algorithm that has been proposed in the literature. We found that this algorithm has unfairness problems and thus we proposed two mechanisms to improve it. Our simulation results show that, when compared with the original algorithm, the improved algorithm can further reduce the average time needed for a vehicle to reach its destination. In addition, the improved algorithm can mitigate the unfairness problem.

In the next-generation intelligent transportation systems, it is envisioned that the DSRC (dedicated short range communication) technology will be extensively used. Our simulations also studied how the DSRC packet loss rates on the roads may affect the effectiveness of traffic signal control algorithms.

I. INTRODUCTION

Traffic signal control is an important research topic in intelligent transportation systems (ITS) because it directly affects the efficiency of a city’s transportation systems. A good traffic signal control algorithm can shorten the required travel time of most drivers on the roads, which also reduces the cost of fuel and the air pollution caused by fuel consumption.

Traffic signal control has been researched for many years and several approaches have been proposed in the past. Compared with the approaches used in the past, the approach used in this paper has the following prominent differences. First, this paper uses the microscopic simulation approach with vehicles moving on the roads of an imported city (Taipei) map. Second, the simulated vehicles use realistic driving behaviors (e.g., car-following, overtaking, stop/wait/start in front of traffic signals, etc.) when moving on the roads. Third, and also the most important feature, in the studies of this paper, we can designate some places on the map as landmarks, and each simulated vehicle is randomly assigned a landmark to reach as its destination. When arriving at an intersection, the vehicle will make a turn along its shortest path to its current landmark. After reaching its current landmark, it will randomly select another landmark as its new destination and repeat this process until the simulation is ended. With this unique capability, simulated vehicles now make meaningful turns when they arrive at an intersection. This traffic behavior resembles the behavior of a driver in the real world as normally a driver must have a destination in his/her mind and will not make random turns when arriving at an intersection. With this realistic traffic behavior, our simulation results about the effectiveness of traffic signal control are more convincing than those obtained using random turns at intersections.

In addition to the above differences, in this paper we envision that each simulated vehicle has a DSRC (Dedicated Short Range Communication) On-Board Unit (OBU) and each traffic signal has a DSRC Road-Side Unit (RSU). While waiting in front of a traffic signal, a vehicle’s OBU will send a packet to the signal’s RSU. By receiving these DSRC packets, a traffic signal can therefore know how many vehicles are currently stopping and waiting in front of it. Because DSRC packets can be lost due to weak received wireless signal and packet collisions caused by hidden-terminal problems, in this paper, we investigate how the packet loss rate of these DSRC packets will affect the effectiveness of traffic signal control algorithms.

Our simulation studies are carried out using the renowned NCTUns VANET (Vehicular Ad Hoc Network) network simulator [1] [2] [3] [4], which supports the landmark-based traffic pattern and high-fidelity DSRC packet transmission and reception simulation. To the best of the authors’ knowledge, there is no paper in the literature that studied the performances of traffic signal control algorithms under landmark-based traffic patterns and different DSRC packet loss rates.

II. RELATED WORK

In fixed-time control, Webster [5] gave equations for the optimal cycle length and the green phase time assignment. Based on Webster’s theory, Akselik [6] [7] proposed time-dependent expressions to predict traffic operating characteristics at signals. These methods set traffic signal periods based on history records. If the current traffic conditions are similar to history records, they can perform well with low calculation costs. However, when traffic condition varies dynamically, they cannot respond well in real time. With computers being...
widely used for the design and control of traffic facilities, many advanced systems have been developed to adjust signal timings according to real-time traffic information. For instance, extending green signals according to the detected traffic conditions in real time is one such method. A number of adaptive traffic control systems such as SCOOT [8], SCATS [9], OPAC [10] and RHODES [11] have been deployed in the world.

Most of the papers that we have surveyed only use average waiting time or average queue length to assess an algorithm’s performance. However, a low average waiting time may come at the cost of delaying some vehicles for too long due to unfair traffic control. In this paper, in addition to the average waiting time performance metric, we use “fairness” as another performance metric to assess an algorithm’s performance. To increase fairness during traffic signal control, we improved the algorithm proposed in [12] and compared the performance of the original and the improved algorithms under more realistic landmark-based traffic patterns.

III. THE ORIGINAL ALGORITHM

Because the improved algorithm is based on the algorithm proposed in [12], in the following we use the notations and equations used in [12] to explain the improved algorithm. To help the explanations, a four-leg signalized intersection is depicted in Fig. 1.

The queue length is an important variable that describes the traffic state of an intersection. The queue evolves as

\[ Q_i(n+1) = Q_i(n) + q_i(n) - d_i(n)S_i(n) \]  

(1)

where \( i = 1,2,3 \), and \( M \) is the index of the traffic streams across all intersections; \( n = 0,1, \ldots ,N-1 \) is the index of the discretized time intervals; \( Q_i(n) \), in unit of number of vehicles, is the queue length of the \( i \)-th stream at the onset of the \( n \)-th time interval; \( q_i(n) \) is the number of vehicles that depart from the queue in leg \( i \) at the saturation flow rate in the \( n \)-th time interval and its value must be less than or equal to \( Q_i(n) + q_i(n) \). Because leg 1 and leg 3 are controlled by the same signal while leg 2 and leg 4 are controlled by another signal and these two signals are opposite signals, to minimize the \( W(n+1) \) of an intersection, for the \( n \)-th time interval we can set \( S_1(n) = 1 \) (green light) if \( d_1(n) + d_3(n) \) is greater than \( d_2(n) + d_4(n) \) and set it to 0 (red light) otherwise. This condition test and the resulting signal assignment are carried out in every time interval and we call this approach “the original algorithm” in this paper.

IV. THE IMPROVED ALGORITHM

We implemented and evaluated the original algorithm over the NCTUns network simulator under various realistic landmark-based traffic patterns. We found that generally the original algorithm can reduce the average signal waiting time by 30% as compared with when no algorithm is used to intelligently control the signals (i.e., when only the fixed-time signal control is used). This is a good result. However, we found that the original algorithm can be further improved in two ways.

In the original algorithm, the traffic flow condition (i.e., the \( W_i(n+1) \)) at an intersection is evaluated at a fixed time interval \( T \) (in this paper, it is set to 15 seconds). However, due to the use of fixed-time intervals, the traffic signal can only be changed after a fixed period of time. In certain circumstances, this may result in waste of time. Suppose that \( d_1(n) + d_3(n) \) is very small but \( d_2(n) + d_4(n) \) is very large, much of the green light interval for leg 1 and leg 3 will be wasted if the
fixed time interval T is too long for leg 1 and leg 3. Instead, if we can use dynamic time intervals, it will be more efficient to give a shorter green light interval to leg 1 and leg 3 but give a longer green light interval to leg 2 and leg 4.

To solve this problem, we improved the original algorithm by using dynamic time intervals. If both $d_1(n) + d_3(n)$ and $d_2(n) + d_4(n)$ are large, the improved algorithm will use a longer green light interval (say, 30 seconds) for both directions to increase transportation efficiency. On the other hand, if $d_1(n) + d_3(n)$ is small but $d_2(n) + d_4(n)$ is large, the improved algorithm will use a shorter green light interval for leg 1 and leg 3 but a relatively longer green light interval for leg 2 and leg 4 to reduce unnecessary waiting time. The chosen interval is proportional to $d_1(n) + d_4(n)$ (or $d_2(n) + d_3(n)$) but must be greater than the minimum time interval and less than the maximum time interval, which are set to 10 and 60 seconds, respectively, in this paper.

Another mechanism that we used to improve fairness is the aging mechanism. Although the improved algorithm uses dynamic time intervals to increase transportation efficiency and reduce unnecessary waiting time, the decision is made independently for each time interval. A situation may occur in which the algorithm decides to continuously give green light to the same direction of flow (say, leg 1 and leg 3) in consecutive time intervals if the queues in that direction are relatively much longer than the queues in the other direction. This will increase the waiting time of the vehicles waiting in a short queue and may even cause them to “starve.” To overcome this problem, we used an aging mechanism to increase the weight (priority) of a short queue if it is not given green light in a time interval. The weight-increasing policy is designed so that a short queue can only be deprived of its opportunity of green light at most two times. After that, as long as its queue length is not zero, it will get the green light no matter how short the queue is.

V. SIMULATION SETTINGS

In this paper, we used the NCTUns network and traffic simulator as our simulation platform. NCTUns is a well-known network simulator and there have been many published researches conducted on this platform. Based on an innovative simulation methodology, NCTUns generates realistic simulation results by directly using the real-world network applications and the real-world TCP/IP protocol stack in the Linux kernel. In addition, NCTUns is also a powerful traffic simulator. One unique feature of NCTUns is that it can make traffic simulation results more realistic by loading a real-world roadmap in the shapefile format. For example, Fig. 2 shows the loaded roadmap of a district of the Taipei city. NCTUns possesses many features in the traffic simulation. Regarding road simulation, it supports multi-lane road simulation, which enables more diverse road and vehicle movement simulations. NCTUns has two operation modes when controlling traffic signals. In the first mode, traffic signals are controlled by a centralized control unit. In the second mode, each traffic signal can be independently controlled by its own control unit. In our research, we implemented and evaluated the original and the improved algorithms in the second mode.

A. Landmark-Based Traffic Pattern

Intelligent traffic signal control algorithms are based on real-time traffic conditions to dynamically set the traffic signals. To use the simulation approach to evaluate the performance of such algorithms, one must have a practical and realistic traffic model that creates realistic congestion conditions on the roads. As presented before, most other VANET network simulators only provide a simple traffic model in which a vehicle makes a random turn when arriving at an intersection. The traffic patterns created by this simple traffic model is very unlike the real-world traffic pattern. This is because normally most drivers must have a destination in his/her mind and will make a turn that is on the shortest path to the destination. Our simulation results also revealed that there is no way to create traffic congestion if the random-turn traffic model is used during simulation. To evaluate the performances of traffic signal control algorithms, we implemented a more realistic traffic model called the Landmark-Based Traffic Model (LBTM) over the NCTUns network simulator.

We created a new type of object, called “landmark,” in NCTUns. A landmark can be viewed as a hot spot in a city and one can select many places on the map as landmarks. For example, Taipei 101 building and the National Palace Museum can be designated as landmarks because they are popular hot spots. During simulation, each vehicle randomly selects a landmark as its current destination and moves towards it. This means that the vehicle will not make a random turn when arriving at an intersection but will make a meaningful turn to try to reach its destination along the shortest path. After the vehicle arrives at its current destination landmark, it randomly selects another landmark as its new destination and the above-mentioned process continues. With this design, if one properly sets the number of landmarks, their locations, and their probability distribution of being selected as the next landmark, one can create very realistic traffic patterns on the roads. In this way, congested roads can be created and
the waiting queues of traffic streams at intersections can be formed.

As a verification of the effectiveness of LBTM, we randomly deployed one hundred vehicles on the map as shown in Fig. 3. In the first case, we did not specify any landmark on the map but instead let these vehicles make random turns at intersections. The simulation time of this case is 600 seconds. At the end of the simulation, the final locations of these vehicles are shown in Fig. 4. One sees that because these vehicles make random turns at intersections, these vehicles are quite evenly distributed on the map at the end of the simulation. In this situation, one has no way to create traffic congestion and hot spots in the city. As a result, it is difficult to use this random-turn traffic model to study the performances of traffic signal control algorithms.

In the second case, we deployed four landmarks on the map as shown in Fig. 5. During simulation, each vehicle selects a landmark as its destination and moves towards it. After reaching the current landmark, then it selects another landmark.
as its new destination and repeats this process. Again, the movements of these vehicles are simulated for 600 seconds and the final locations of these vehicles are shown in Fig. 6. One sees that now these vehicles are jammed on the roads connecting these landmarks, which enables us to study the performance of traffic signal control algorithms under traffic congestion conditions.

VI. SIMULATION RESULTS

A. Performance Comparison

We first evaluated the performances of the fixed-time, the original and the improved algorithms. We used three different traffic patterns called traffic pattern A, B and C. In traffic pattern A, B, and C, 5, 10, and 20 landmarks are randomly deployed on the map, respectively. Three different traffic densities are used in our studies. They correspond to deploying 70, 140, and 210 vehicles on the map. Various combinations of traffic patterns and traffic densities are used to compare the performances of these algorithms.

Two performance metrics are used in this paper. One is the average time spent of the vehicles to reach their respective landmark destinations. The unit of this performance metric is second and the lower the value, the higher the performance. The other metric concerns the fairness of an algorithm. To see whether an algorithm may sacrifice some vehicles (by letting them wait too long in front of traffic signals) to reduce the average spent time of vehicles, we also see the statistics distribution of the time saved for each vehicle and the time increased for victim vehicles. If algorithm A can result in fewer victims than algorithm B, we view that algorithm A is fairer than algorithm B.

Fig. 7 shows the comparison of the three algorithms under traffic pattern A, B, and C, when there are only 70 vehicles moving on the roads (which represents a low traffic density environment). “Fixed-Time” means that we do not use any intelligent algorithm to control the traffic signals. They simply switch between the red and green lights every 40 seconds. “Original” means that we use the original algorithm proposed in [12] to control the traffic signals. We let the algorithm assess the traffic condition to adjust the traffic signals every 15 seconds. “Improved” means that we use the improved
algorithm, which uses dynamic time intervals and an aging mechanism, to assess the traffic condition and adjust the traffic signals. A clear finding is that the original and the improved algorithms, both of which adjust traffic signals based on real-time waiting queue lengths, greatly outperform the fixed-time algorithm, which does not take the real-time traffic conditions into account. Another finding is that in a low traffic density environment like the one studied in this case where only 70 vehicles are moving on the roads, under traffic pattern A, B, and C, the improved algorithm can further reduce the average time spent of vehicles than the original algorithm.

In Fig. 8, Fig. 9, and Fig. 10, we compare the fairness of the original and the improved algorithms, under traffic pattern A, B, and C, respectively, when the number of vehicles are 70. For each vehicle, we compute the difference of its waiting time between when the fixed-time algorithm is used and when the original or the improved algorithm is used. A positive value means a waiting time decrease while a negative value means a waiting time increase. When the original or the improved algorithm is used, we make a statistics distribution of the time changes of all vehicles and classify them into various ranges. The data on the “Original” row show the distribution of the time changes of all vehicles when the original algorithm is used, as compared with the fixed-time algorithm. On the other hand, the “Improved” row show the distribution when the improved algorithm is used as compared with the fixed-time algorithm.

From these figures, one first sees that most vehicles save their waiting time due to the uses of these intelligent algorithms. However, one also sees that a small number of vehicles increase their waiting time, which we call victims. The improved algorithm can reduce the number of victims than the original algorithm. When one looks at the number of vehicles classified into the ranges of [-100, 0] and [-200, -100], which indicates a waiting time increase between 0 and 100 seconds and a waiting time increase between 100 seconds and 200 seconds, one sees that under traffic pattern A, B, and C, the improved algorithm can reduce the level of unfairness.

In Fig. 11, we sum up the increased waiting time of all victims under the original and the improved algorithms, respectively. Here, one clearly sees that the improved algorithm indeed reduces the total increased waiting time of victim
In the following, we present the evaluation results in a high traffic density environment, where 210 vehicles are moving on the roads. A new finding from Fig. 12 is that the improved algorithm does not always reduce the average waiting time than the original algorithm when the traffic density is high. This phenomenon is reasonable as in a saturated environment, it is very difficult to further reduce the average waiting time. However, Fig. 13, Fig. 14, Fig. 15 show that the improved algorithm can reduce the level of unfairness suffered by victims. The results of Fig. 16 clearly show that the improved algorithm can reduce the total increased waiting time of victim vehicles in such a high traffic density environment.

B. DSRC Packet Loss Rate Effects

In the following, we present results showing how different DSRC packet loss rates may affect the effectiveness of the original and the improved algorithms. Fig. 17 and Fig. 18 show the average time spent of vehicles under different DSRC packet reception rates (100%, 80%, 60%, 40%, 20%, 0%) when the number of vehicles are 70 and 210 vehicles, respectively. Note that the DSRC packet reception rate is (1 - DSRC packet drop rate). If the DSRC packet reception rate is set to x, it means that a traffic signal can only detect the existence of x% of all the vehicles waiting in front of it.

From these figures, one sees that the performance of the original and the improved algorithms are affected by the DSRC packet reception rate. As for the fixed-time algorithm, because it does not use DSRC packets to detect real-time traffic conditions, its performance is not affected by the DSRC packet reception rate. One sees that when the DSRC packet reception rate decreases, because the traffic information gathered for the original and the improved algorithms to make decisions become more and more incorrect, both the original and the improved algorithms perform worse than when they have 100% correct traffic information.

One sees that when the DSRC packet reception rate is very low, both the original and the improved algorithms may perform worse than the fixed-time algorithm. This phenomenon is reasonable as in such a condition the decisions made by these two algorithms are based on incorrect traffic information, which can easily result in bad traffic jams. However, one also sees that as long as DSRC packet reception rates can be maintained above 40%, both the original and the improved algorithms can still outperform the fixed-time algorithm. These results indicate that when using the DSRC technology to communicate traffic information between vehicles and traffic signals, one needs to design good network and communication protocols at the medium-access-control layer to avoid severe packet collisions. In addition, one needs to provision enough communication bandwidth and network capacity for vehicles using DSRC radios so that DSRC packets will not be dropped easily due to insufficient network bandwidth.

One also sees that both the original and the improved algorithms perform better in the high traffic density environment (70 vehicles). This phenomenon can be explained as follows. When the traffic density is high, more vehicles will be waiting in the queues at the intersections. This will increase the chance of traffic signals detecting some vehicles waiting in the queues and thus making prompt adjustments. In contrast, in a low traffic density environment few vehicles will be waiting in the queues. In such a condition, if some vehicles cannot be detected due to DSRC packet loss rates, the traffic signals may think that there is no vehicle waiting in the queue and thus will not give green lights to the queue at all. As a result, the vehicles waiting in the queue will be blocked for a long time until more and more vehicles have joined the queue. At that time, the traffic signal will have a high probability to detect that some vehicles are waiting in the queue and thus make adjustments.

From these figures, one also sees that the improved algorithm is much less affected by the DSRC packet loss rates as compared with the original algorithm. The reason is that the improved algorithm sets green time according to the number of vehicles that are detected. If the traffic signal only detects part of the vehicles waiting in a queue (i.e., underestimate the number of the vehicles waiting in the queue), it will set a shorter green time for the queue and trigger the next assessment soon. Therefore, if the improved algorithm makes a wrong decision, it can quickly correct the mistake in the next assessment. This is the reason why the improved algorithm outperforms the original algorithm under the same DSRC packet reception rate.

VII. Conclusion

In this paper, we use the microscopic simulation approach with realistic landmark-based traffic patterns to evaluate the performances of intelligent traffic signal control algorithms. The goal of an intelligent traffic signal control algorithm is to shorten the time needed for a vehicle to reach its destination. We chose such an algorithm from the literature and then used the NCTUns network simulator to study its effectiveness under various conditions. We found that the chosen algorithm generally can shorten the time for a vehicle to reach its destination but may achieve the goal at the expense of some vehicles waiting too long due to unfair traffic signal
control. Therefore, we improved the original algorithm and used NCTUns to evaluate the performance of the original and the improved algorithms.

In the next-generation intelligent transportation systems, it is envisioned that DSRC wireless networks will be used to exchange traffic information between the OBU (On Board Unit) on vehicles and the RSU (Road Side Unit). In this paper, we assume that the information about the number of vehicles waiting in front of a traffic control signal is detected by a traffic signal via the DSRC packet transmission and reception between the vehicles and the traffic signal. Because DSRC packets may be dropped due to collisions, poor signal quality, and insufficient network bandwidth, we also studied how the DSRC packet loss rate may affect the effectiveness of the original and the improved algorithms. The improved algorithm is quite robust against the DSRC packet loss rate. Our simulation results show that as long as the DSRC packet loss rate is less than 20%, the performance of the improved algorithm can remain about the same as when no DSRC packets are dropped on the roads.

REFERENCES