Reservation-Based Cooperative Traffic Mangement Considering Turn Types at Intersection

Areeya Rubenecia, Myungwhan Choi
Department of Computer Science and Engineering
Sogang University
Seoul, Republic of Korea
{rubenecia,mchoi}@sogang.ac.kr

Hyo Hyun Choi
Department of Computer Science
Inha Technical College
Incheon, Republic of Korea
hchoi@inhatc.ac.kr

***Abstract*—This paper presents a performance evaluation of a reservation-based intersection traffic control infrastructure. The traffic control infrastructure manages the flow of vehicles at the intersection while vehicles travel at constant maximum speed. We also evaluated another model that considers the turn types of the vehicles. The performance of the models was evaluated based on its queueing delay, queue length, and efficiency.**

Keywords*—*intersection traffic management; reservation-based; autonomous driving

I. Introduction

One cause of traffic congestion in the conventional traffic system is travel at intersections that use traffic lights. Traffic lights can contribute to traffic delay in cases when vehicles are required to stop at the entrance of the intersection even though no other vehicle is currently crossing the intersection. A solution to this problem is removing traffic lights and allowing vehicles and intelligent systems to communicate and autonomously manage the access of the intersection.

For this reason, the use of autonomous vehicles has been introduced to address this problem. Consequently, it is necessary to have an efficient autonomous management of vehicles at the intersection that ensures collision-free travel of the vehicles and at the same efficiently utilize the intersection by increasing flows.

A variety of systems have been proposed on autonomous intersection management. One approach is reservation-based systems as used in [1], [2], [3]. This kind of system has intersection manager that supervises and grants access of the intersection to the vehicles. Furthermore, different algorithms were developed in deciding which vehicle will be assigned the right-of-way. Decomposition and back-tracking is used in [4]. In [5], an algorithm using ant colony system is developed. In [6], this problem is shown as a scheduling problem and a mathematical approach is used by using priority graphs. A sequence-based protocol is developed in [7] in which the intersection is managed by giving sequence to the vehicles. In [8], an algorithm is developed using game theory framework.

In this study, we simulated the scheme proposed in [1] and evaluated its performance based on its queueing delay, queue length, and efficiency. In addition, we also run simulations on a new model that considers the turn type of the vehicle when managing reservation requests. Then we compared the results of the simulation of the scheme in [1] and the new model.

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2015R1D1A1A01061155).

II. MODEL DESCRIPTION

A. *Intersection Model*

We name the model in [1] as model 1. In model 1, vehicles travel from their respective arrival locations until they reach their destinations. When traveling, the vehicle has the ability to control its speed to prevent collision with the vehicle ahead, if there is any. In general, multiple vehicles are traveling the road and they all need to traverse the intersection. To manage the vehicle's entrance at the intersection, we use the reservation-based approach in which each vehicle must send a request to the intersection manager and then the intersection manager will schedule the vehicle to a time that all of its request points are available.

One key feature of this model is it only allows the vehicle to travel at a constant maximum speed at the intersection. This is to make the vehicles spend as little time as possible in the intersection. For this reason, an acceleration lane is located before the entrance of the intersection. A vehicle at the acceleration lane must accelerate from its current speed to the required maximum speed in the intersection. Then it must maintain travelling at this speed while at the intersection.

B. *Vehicle's Turn Type*

We define three turn types which are the paths that a vehicle may take when crossing the intersection as shown in Fig. 1. We define paths similar to the orange arrow as turn-type A with shortest path length, blue arrow as turn type B with medium path length and green arrow as turn-type C with longest path length.

C. *Reservation Points*

These are the points that the vehicle will occupy when crossing the intersection and are needed in the reservation request. Thus, these also represent the paths that the vehicles will take in the intersection. For example, turn-type A will request point {2}, B will request points {1, 2} and C will request points {1, 2, 4}. We have defined only four reservation points since these are enough to represent the turn type.

978-1-5090-5140-3/17/$31.00 ©2017 IEEE

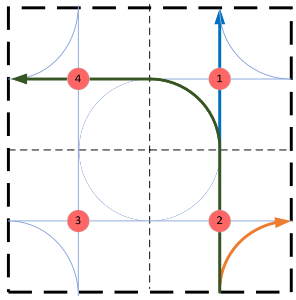

Fig. 1. Possible turn types and reservation points at intersection

Consequently, when a request is received by the intersection manager, it computes and schedules the earliest possible entrance time of the vehicle to the intersection wherein all of the reservation points are available.

D. Requesting for Reservation

A vehicle can only send a reservation request when no other vehicle is in front or when the vehicle in front is already scheduled for entrance to the intersection.

When the vehicle is already scheduled, it should move towards the acceleration zone and must accelerate to its maximum speed until it reaches the intersection. Also, it must maintain its maximum speed while crossing the intersection.

E. Scheduling Algorithm

This scheduling algorithm is our implementation of the described behavior of the intersection manager in [1].

1) Reservations

Let $RP = \{RP_1, RP_2,\ldots, RP_j\}$ denote the set of reservation points in the intersection. Additionally, RP_j has a sorted list of scheduled reservation times that a vehicle will occupy it, $RPT_j^{entrance} = \{RPT_{j,1}^{entrance}, RPT_{j,2}^{entrance}, \ldots, RPT_{j,k}^{entrance}\}$ and exit times $RPT_j^{exit} = \{RPT_{j,1}^{exit}, RPT_{j,2}^{exit}, \ldots, RPT_{j,k}^{exit}\}$ to keep track of when are the points available for reservations.

2) Vehicle

For each vehicle V_i in the lane, if no other vehicle is in front or the vehicle in front V_{i-1} is already reserved, then V_i sends reservation request. Reservation request *Req* contains expected entrance and exit times for each *RP* that it will traverse i.e. $Req = (ReqRPT_j^{entrance}, ReqRPT_j^{exit})$.

3) Intersection manager

For each *Req* received by the intersection manager, every $ReqRPT_j^{entrance}$ and $ReqRPT_j^{exit}$ should be available from the list of entrance times $RPT_j^{entrance}$ and exit times RPT_j^{exit}.

To check the availability of the points at these times, at least one of the following must be true for each *RP*:

1. $RPT_j^{entrance}$ and RPT_j^{exit} are empty.
2. There exists $RPT_j^{exit} <= ReqRPT_j^{entrance}$ and $RPT_{j+1}^{entrance} >= ReqRPT_j^{exit}$.
3. The last entry in the list, $RPT_{j,k}^{exit} <= ReqRPT_j^{entrance}$
4. $ReqRPT_j^{exit} <= RPT_{j,1}^{entrance}$, the first entry in the list.

If at least one reservation point is not available from the requested points, increment each time in $ReqRPT^{entrance}$ and $ReqRPT^{exit}$. After incrementing, check the availability again with these new times. Keep on doing this until all of the requested points are available.

Once there is a matching available time for this request, add these to $RPT_j^{entrance}$ and RPT_j^{exit}. Also, the intersection manager will inform the vehicle of the scheduled intersection entrance time. In case of more than one request received at the same time, the intersection manager will randomly choose which request to process first.

III. Performance Evaluation

A simulation was run to evaluate the efficiency of the model. For the performance evaluation, we introduce another model, model 2, to compare with model in 1. We compared the performance of the two models using the queuing delay, queue length and efficiency.

We run the simulation using exponentially distributed intergeneration time. We also randomly assigned turn type to each vehicle with probability of 1/3 for each turn type and the average of the last 100 from 8000 vehicles was computed. The simulation was also run for acceleration lane length of 5 m.

In order to compare the models, a vehicle's turn type and arrival time are the same for each run for both of the models. We also set all of the vehicles' average travel speed to 10 m/s and intersection speed to 20 m/s.

A. Model 2

This model is similar to model 1 except that we prioritize managing the request of the vehicle with a turn type that has the shorter travel time in the intersection. This means that instead of the intersection manager arbitrarily processing the requests that are received at the same time, we first schedule the vehicle with the shortest travel time among the requests that are received at the same time.

B. Queuing Delay

The queueing delay of a vehicle is from the time it stopped due to a queue of the vehicles in front until the time it entered the acceleration zone. Table 1 and Fig. 2 show the results when the acceleration lane length is 5 m. Based on the results, both models show the same performance at some runs, mostly when generation rate is low. For high generation rate, model 2 may perform better since scheduling first the vehicle with the shortest intersection travel time will reduce the queueing time of the succeeding vehicles in queue. However, this feature of model 2 has little impact to the improvement of the performance because the intersection manager allows more than one vehicle at the intersection at a time.

Another observation is that the length of the acceleration zone also has an impact on the delay. Shorter acceleration zone makes the travel time of the vehicles shorter thus decreasing the delay.

C. *Queue Length*

The queue length is the number of vehicles in the queue when the vehicle stopped due to a queue of the vehicles in front. The results for simulations with acceleration lane length of 5 is shown in Table 2 and Fig. 3. The results show similar trend to the queueing delay. This is because the queueing delay is dependent to the queue length. Also, model 2 shows slightly better result than model 1.

D. *Efficiency*

The efficiency is measured in this simulation by the number of cars that entered the intersection within a time range. In our simulation, this time range is 10 minutes. As shown in Table 3 and Fig. 4, the number of cars increases as the rate increases. Then it slowly increases as the rate reaches 0.8 or when the inter-generation time is around 1.25 seconds.

TABLE 1: COMPARISON OF AVERAGE QUEUEING DELAY OF 100 VEHICLES USING EXPONENTIALLY DISTRIBUTED GENERATION RATE

Generation rate (car/min)	Average Queueing Delay (s)	
	Model 1	Model 2
0.2	0.04	0.04
0.4	0.09	0.09
0.6	0.19	0.17
0.8	0.41	0.41
1	3.05	2.08
1.1	111	109.72

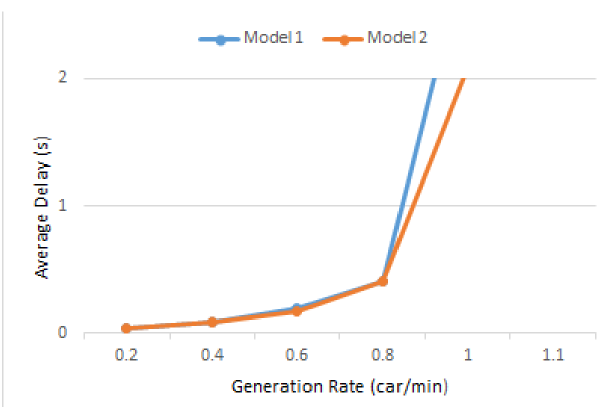

Fig. 2. Comparison of average queueing delay of 100 vehicles using exponentially distributed generation rate

TABLE 2: COMPARISON OF AVERAGE QUEUE LENGTH OF 100 VEHICLES USING EXPONENTIALLY DISTRIBUTED GENERATION RATE

Generation rate (car/min)	Average Queue Length (car)	
	Model [1]	Model 2
0.2	0	0
0.4	0.04	0.04
0.6	0.04	0.04
0.8	0.36	0.36
1	13.96	10.24
1.1	188.56	161.84

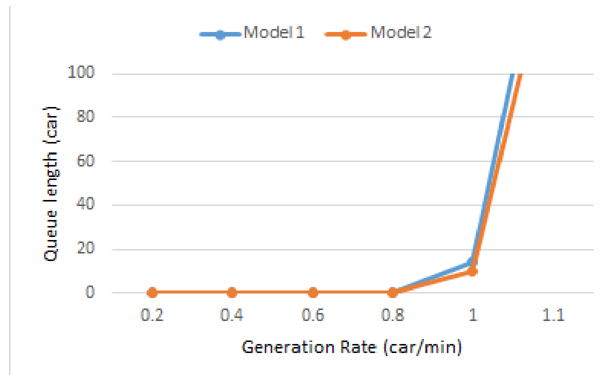

Fig. 3. Comparison of average queue length of 100 vehicles using exponentially distributed generation rate

TABLE 3: COMPARISON OF EFFICIENCY BETWEEN MODELS USING EXPONENTIALLY DISTRIBUTED GENERATION RATE

Generation rate (car/min)	Efficiency (car)	
	Model [1]	Model 2
0.2	501	501
0.4	922	922
0.6	1387	1387
0.8	1801	1801
1	1863	1860
1.1	1964	1918

IV. CONCLUSION

This paper has showed results on the performance evaluation of a reservation-based intersection traffic control infrastructure. We implemented a reservation-based scheme and evaluated its performance based on queueing delay, queue length, and efficiency. Also, we examined the impact of considering the turn type in the performance of the model. From our simulation results, it shows that with this scheduling scheme, considering the priority has a very small impact to its performance.

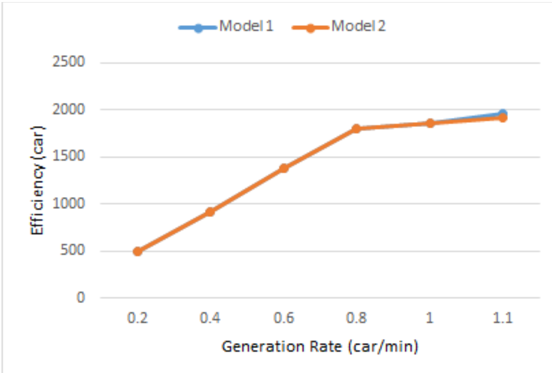

Fig. 4. Comparison of efficiency between models using exponentially distributed generation rate

REFERENCES

[1] A. de La Fortelle. "Analysis of reservation algorithms for cooperative planning at intersections." 13th International IEEE Conference on Intelligent Transportation Systems (ITSC), pp. 445-449, 2010.

[2] K, Dresner, and P. Stone. "Multiagent traffic management: An improved intersection control mechanism." Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems, pp. 471-477, 2005.

[3] M. Zhu, X. Li, H. Huang, L. Kong, M. Li, and M. Wu. "LICP: A look-ahead intersection control policy with intelligent vehicles." 6th International Conference on Mobile Adhoc and Sensor Systems, pp. 633-638, 2009.

[4] J. Wu, A. Abbas-Turki, and A. El Moudni. "Discrete methods for urban intersection traffic controlling." 69th Vehicular Technology Conference, pp. 1-5, 2009.

[5] J. Wu, A. Abbas-Turki, and A. El Moudni. “Cooperative driving: an ant colony system for autonomous intersection management.” Applied Intelligence, 37(2), pp.207-222, 2012.

[6] J. Gregoire, S. Bonnabel, and A. de La Fortelle. "Optimal cooperative motion planning for vehicles at intersections." arXiv preprint arXiv:1310.7729, 2013

[7] F. Perronnet, A. Abbas-Turki, and A. El Moudni. "A sequenced-based protocol to manage autonomous vehicles at isolated intersections." 16th International IEEE Conference on Intelligent Transportation Systems, pp. 1811-1816, 2013.

[8] I. H. Zohdy, and H. Rakha. "Game theory algorithm for intersection-based cooperative adaptive cruise control (CACC) systems." 15th International IEEE Conference on Intelligent Transportation Systems, pp. 1097-1102, 2012.