Paper No.: 13-2186

Modeling Reservation-based Autonomous Intersection Control in VISSIM

A Paper Submitted to
the Transportation Research Board
for Review for Presentation & Publication
at the TRB 92nd Annual Meeting in Washington, D.C., January 13-17, 2013

Zhixia Li, Ph.D.
Research Associate
1249A Engineering Hall, 1415 Engineering Drive, Madison WI 53706
Phone: 513-484-2991; Fax: (608)262-5199
Email: zli262@wisc.edu

Madhav V. Chitturi, Ph.D.
Assistant Researcher
B243 Engineering Hall, 1415 Engineering Drive, Madison, WI 53706
Phone: (608)890-2439, Fax: (608)262-5199
Email: mchitturi@wisc.edu

Dongxi Zheng, M.S.
Research Assistant
1249A Engineering Hall, 1415 Engineering Drive, Madison WI 53706
Phone: (608)335-0889
Email: dzheng3@wisc.edu

Andrea R. Bill
Associate Researcher
B243 Engineering Hall, 1415 Engineering Drive, Madison, WI 53706
Phone: (608)890-3425, Fax: (608)262-5199
Email: bill@wisc.edu

David A. Noyce, Ph.D. PE
Professor
Director, Traffic Operations and Safety (TOPS) Laboratory
1204 Engineering Hall, 1415 Engineering Drive, Madison, WI53706
Phone: (608)265-1882, Fax: (608)262-5199
Email: noyce@engr.wisc.edu

Traffic Operations and Safety (TOPS) Laboratory
Department of Civil and Environment Engineering
University of Wisconsin-Madison

5237 + 2 Table ×250 +7 Figures×250 = 7487 words (Revised 11/15/2012)

ABSTRACT

Autonomous vehicles are attracting more and more attention as a promising approach to improve both highway safety and efficiency. Most previous studies on autonomous intersection management relied heavily on custom-built simulation tools to implement and evaluate their control algorithms. The use of the non-standard simulation platforms makes comparison between different systems almost impossible. Additionally, without support from standard simulation platforms, reliable and trustworthy simulation results are hard to obtain. In this context, this paper explores a way to model autonomous intersections using VISSIM, a standard microscopic simulation platform. Specifically, a reservation-based intersection control system, named Autonomous Control of Urban TrAffic (ACUTA), was introduced and implemented in VISSIM using VISSIM's External Driver Model. The operational and safety performances of ACUTA were evaluated using the easy-to-use evaluation tools of VISSIM. Compared with the optimized signalized control, significantly reduced delays were resulted from ACUTA along with a higher intersection capacity and lower volume-to-capacity (v/c) ratios under various traffic demand conditions. The safety performance of ACUTA was evaluated using the Surrogate Safety Measure Model, and presented few conflicts among vehicles within the intersection. Moreover, the key steps and elements for implementing ACUTA in VISSIM are introduced in the paper, which can be useful for other researchers and practitioners in implementing their autonomous intersection control algorithms in a standard simulation platform. By using a standard simulation platform, performance of different autonomous intersection control algorithms can be eventually compared.

INTRODUCTION

With the rapid advances in sensing, information processing, machine learning, control theory and automotive technology, wide application of autonomous vehicles on highway systems is no longer a dream, but a reality in near future. Autonomous vehicles are vehicles without human intervention (in-vehicle or remote) and are capable of driving in real-world highway systems by performing complex tasks such as merging, weaving, and driving through intersections. Many automotive manufacturers including General Motors, Ford, Mercedes-Benz, Volkswagen, Audi, BMW, Volvo, and Cadillac have already begun testing their autonomous vehicle on highway systems (*1*). Google is also developing and testing its Google driverless car. As of 2012, Florida, Hawaii, Nevada, Oklahoma, and California have legalized or are considering legalization of autonomous cars (*1*). All these facts indicate that the autonomous vehicles are set to appear on road in near future.

Most field tests for autonomous vehicles were restricted to highway segment testing. Intersection control of autonomous vehicles has been studied by researchers (*2-19*), however, implementation in practice is difficult because intersections create more conflict points than highway segments. For example, when vehicles arrive at an intersection from different approaches, the right of way for traversing the intersection needs to be determined. Traditional intersections use traffic control devices, such as stop signs and traffic signals, to regulate vehicles right of way. For managing autonomous vehicles at intersections, the right of way may be controlled by an intersection central controller through vehicle-infrastructure (V2I) communications (*2-12*), or through negotiation between vehicles via vehicle-vehicle (V2V) communications (*13-17*).

Studies have been conducted to explore ideas and algorithms for managing autonomous vehicles at intersections. By control strategy, the autonomous intersection control can be classified into centralized control and decentralized control. For centralized control, all vehicles establish communication connections to an intersection central controller, or intersection manager (*2-12*). The intersection manager determines the vehicles' passing sequence. In a decentralized control system there is no intersection manager. The passing sequence is typically negotiated by vehicles based on a certain protocol (*13-17*). Among all these available solutions, the reservation-based centralized control system has been found to work best for urban intersections with high traffic demand because of its mechanism of maximizing the intersection capacity (*14*).

Due to the complexity of field implementation, most researchers used traffic simulation to validate their developed strategies for autonomous intersection control. However, none of the exiting studies used standard commercial traffic simulation software such as VISSIM or CORSIM when evaluating the performance of their proposed strategies. Rather, simulation tools developed by the respective authors were used in the evaluation process, which made the results less reliable and hard compare with each other. In addition, it was noticed that most existing studies lacked standard usage of terms and clear description of simulation parameter settings when presenting the evaluation results. For example, when presenting the traffic volume, no clarification of whether the volume is per lane or per entire approach was presented. Also, terms to define lane configurations, speed distribution, volume, and delay, as well as the number of runs per experiment, random seed selection, and simulation period were excluded from the analyses, or were not consistently defined across different studies. Most likely the inconsistency is due to the usage of different custom-built simulation software programs, rather than standard commercial simulation software packages.

Standard simulation packages like VISSIM and CORSIM can provide standard parameter settings and outputs. In addition, using the standard package can guarantee reliable vehicle generation, car-following, lane-changing, and many other driving behavior related modeling in the simulation. Flexible settings of speed distribution, heavy vehicle percentage, and distributions of acceleration and deceleration rates can also be simply achieved, along with strong evaluation outputs like travel time and delay. Moreover, commercial packages like VISSIM have options to output vehicle trajectories, which can be directly imported into Surrogate Safety Assessment Model (SSAM) to analyze the safety performance of the intersection (*19*).

Wu et al. indicated in their paper that they chose to develop their own simulation tool rather than use standard traffic simulation packages such as VISSIM, AIMSUN, or PARAMICS, because the standard packages do not allow vehicles to be controlled individually (14). In fact, VISSIM offers flexible customization functions to facilitate building different special applications through APIs and COM extensions. All these functions offer the potential to implement applications for autonomous intersection control. In this paper, implementation of a reservation-based system in VISSIM using VISSIM's External Driver Model is presented. The establishment of the simulation model, implementation of the reservation-based control algorithm, and finally evaluations of operational and safety performance are discussed.

ENHANCED RESERVATION-BASED AUTONOMOUS INTERSECTION CONTROL

A reservation-based system utilizes a centralized control strategy for managing fully-autonomous vehicles at an intersection. All vehicles in a reservation-based system communicate only to a centralized intersection controller, namely, intersection manager (IM). The IM regulates the intersection by determining the passing sequence of all the approaching vehicles (*2-10*).

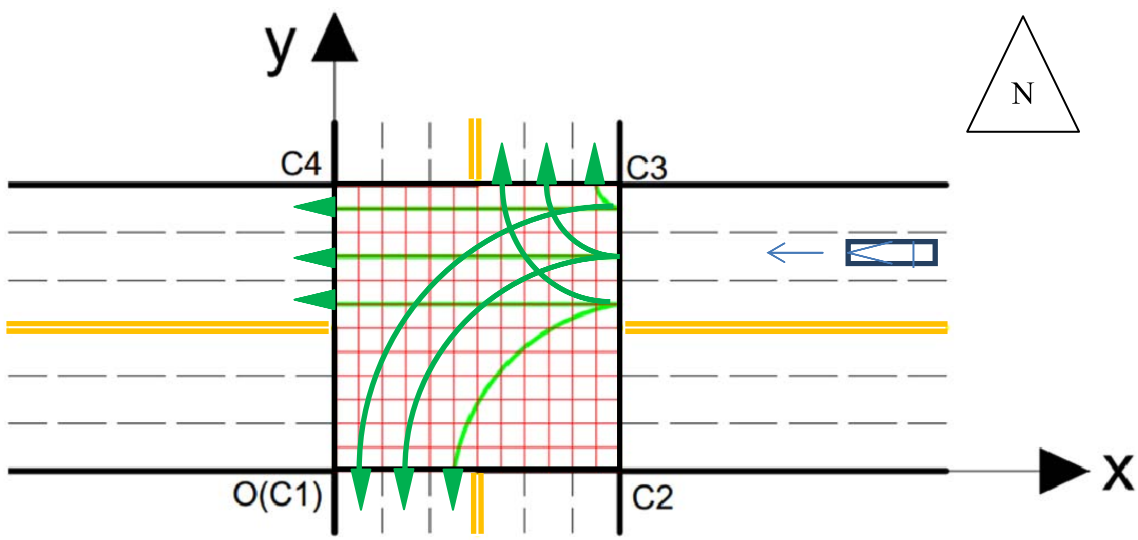


FIGURE 1 Intersection mesh of tiles and example of vehicle's possible routing decisions.

The system presented in this paper is named as Autonomous Control of Urban TrAffic (ACUTA), which is developed based on First-Come-First-Serve (FCFS) reservation-based protocol (*2*) with enhancements to improve some operational issues identified in previous studies (*2, 8*). These issues include the "starvation" issue where approaching vehicles on the side street cannot get reservations when the traffic demands on the major and side street are unbalanced; and (2) slow-speed reservation issue which unnecessarily occupies many intersection resources. ACUTA regulates an intersection which is divided into a mesh of n by n tiles, as shown in Figure 1, where n is termed as granularity, and reflects the tile density of the intersection mesh.

In ACUTA , each approaching vehicle sets up a communication connection with the IM after it enters the IM's communication range. When connected, the vehicle immediately sends the IM a reservation request along with the vehicle's location, speed and routing information (i.e., making a left/right turn or going straight), indicating its intention to traverse the intersection. The IM processes the reservation request by computing the required time-spaces for the vehicle to get through the intersection (i.e., intersection tiles that will be occupied by the requesting vehicle for all simulation steps when the vehicle traverses the intersection) based on the location, speed, maximum acceleration rate, and the routing information provided by the requesting vehicle. Acceleration from the requesting vehicle's current location to the entrance boundary of the intersection is considered when computing the required time-spaces. Using different acceleration rates can change the required time-spaces significantly. The alternative acceleration rate shall fall within the range from zero to the maximum acceleration rate of the particular vehicle, and is calculated using the following equation.

$$\begin{aligned} a_i &= 0 && (i = 1) \\ a_i &= a_{\max} - (i-1)\frac{1}{m}a_{\max} && (i > 1) \end{aligned} \tag{1}$$

Where, α_i = ith alternative acceleration rate (ft/s^2);
α_{max} = maximum acceleration rate (ft/s^2); and,
m = maximum number of internal simulations.

The maximum acceleration rate is one of the characteristics particularly pertaining to the requesting vehicle. However, the vehicle must maintain a constant speed when traversing the intersection. In other words, after the vehicle's center point enters the intersection, the vehicle speed does not change until the vehicle completely clears the intersection. The IM checks whether the required intersection tiles have already been reserved by other vehicles at every simulation step. If a conflict is detected, an alternative acceleration rate will be used to compute the required time-spaces, and conflicts will be checked again based on the updated required time-spaces. This iterative process is called internal simulation. The maximum number of trials of the alternative acceleration rates is termed as the maximum number of internal simulations (MAXNIS). If all alternative acceleration rates are tried out in the internal simulation and conflicts in reservation still exist, the reservation request will be rejected; otherwise, the reservation request will be approved by the IM. The IM automatically rejects the requests from a vehicle following a vehicle that is without a reservation.

After making a decision to reject a reservation request, the IM sends a rejection message to the requesting vehicle with a designated deceleration rate, which can be calculated using the following equation.

$$a_{Dec} = \frac{{v_0}^2}{2(s_0 - d_0 - v_0\delta)} \tag{2}$$

Where, α_{Dec} = designated deceleration rate (ft/s^2);
v_0 = vehicle's speed at the time when submitting the request (ft/s);
S_0 = vehicle's distance from intersection at the time when submitting request (ft);

δ = vehicle response time (s); and,
d_0 = distance from the intersection to the advance stop location (ft).

Vehicle response time (δ) in Equation (2) is the time interval between the instant when a vehicle receives a rejection message from the IM and the instant the vehicle applies the deceleration rate. Variable 'δ' is analogous to the driver's perception reaction time in human-operating vehicles. In ACUTA, the default δ is zero, which assumes ideal condition with negligible response time. This assumption is based on the research findings that the DSRC (Dedicated Short Range Communications), which is widely used in the Connected Vehicles research, can achieve negligible delays in milliseconds for transmitting messages and activating in-vehicle safety applications (*20, 21*). For the simplicity of modeling, the milliseconds delay is assumed as zero in the current version of ACUTA. The advance stop location (ASL) (d_0) in Equation (2) is a special parameter in ACUTA, which designates a predefined advance stop location other than the traditional stop line close to the intersection for vehicles with rejected reservation. The ASL is introduced in ACUTA as a major enhancement strategy to address the slow-reservation-speed issue pertaining to vehicles stopping at the traditional stop line. By using the ASL, vehicles with rejected reservations can stop at an upstream distance from the entrance of the intersection; hence gain higher speed when reaching the entrance point of the intersection. A higher entrance speed can increase the chance for the vehicle to get a reservation, meanwhile saving the intersection time-space resources by reducing the vehicle's total traversal time within the intersection. A vehicle with a rejected reservation request will apply the designated deceleration rate and start to decelerate as soon as the rejection message is received. The vehicle keeps sending reservation requests until the request is finally approved by the IM.

If the IM approves a reservation request, it sends an approval message to the requesting vehicle along with a designated acceleration rate that will result in no conflicts with existing reservations. Timestamps indicating when to end the acceleration and when to completely clear the intersection are also sent to the vehicle in the approval message. The approved vehicle will follow the acceleration instruction as soon as it receives the approval message until the vehicle completely clears the intersection.

MODELING ACUTA IN VISSIM

Implementation of ACUTA in VISSIM is realized in this research. In this section how ACUTA algorithm is modeled in VISSIM is presented. The establishment of the simulation model, the algorithm for determining occupied intersection tiles, and implementation of ACUTA using VISSIM external driver model are elaborated in this section.

Simulation Model of ACUTA Intersection

ACUTA was modeled at a four-legged intersection with three lanes per direction, as shown in Figure 2.a. Different from traditional signalized intersections, vehicles can turn from any lanes in an ACUTA intersection, (shown in Figure 2.b) to eliminate en-route lane changes required for turning vehicles, which are a significant contributing factor to vehicle delays due to conflicts caused by vehicle lane change maneuvers.

Each approach of the intersection is more than 2000 feet long with a fixed lane width of 12 feet. The input traffic volume of each lane is identical to create balanced traffic demands from all lanes of the intersection. Each lane has three routing decisions: left turn, through, and right turn. The volume assignments to the routing decisions are the same for all lanes, namely 25%

for left turn, 60% for through, and 15% for right turn. Figure 2.c illustrates the routing decisions of a particular lane. Vehicle composition used is 93% passenger cars and 7% heavy vehicles. The speed distribution of traffic is also fixed at a setting equivalent to the 30 mph speed limit. These settings of VISSIM parameters like approach length, volume distribution, heavy vehicle percentage, made it a unique case of simulation. VISSIM provides simple options to change its parameter configurations including all of the aforementioned settings. Different settings are not expected to make ACUTA more complicated.

No priority rules, conflict areas, desired speed decisions, reduced speed areas, traffic signals, or stop signs are used in the simulation model, because the traffic control at the intersection is governed by the intersection manager only. Figure 2.d illustrates the screenshot of a simulation run, in which the red vehicles are vehicles that do not have a reservation; green vehicles are vehicles that have a reservation and are in the process of passing the intersection; and, yellow vehicles are those that have already cleared the intersection.

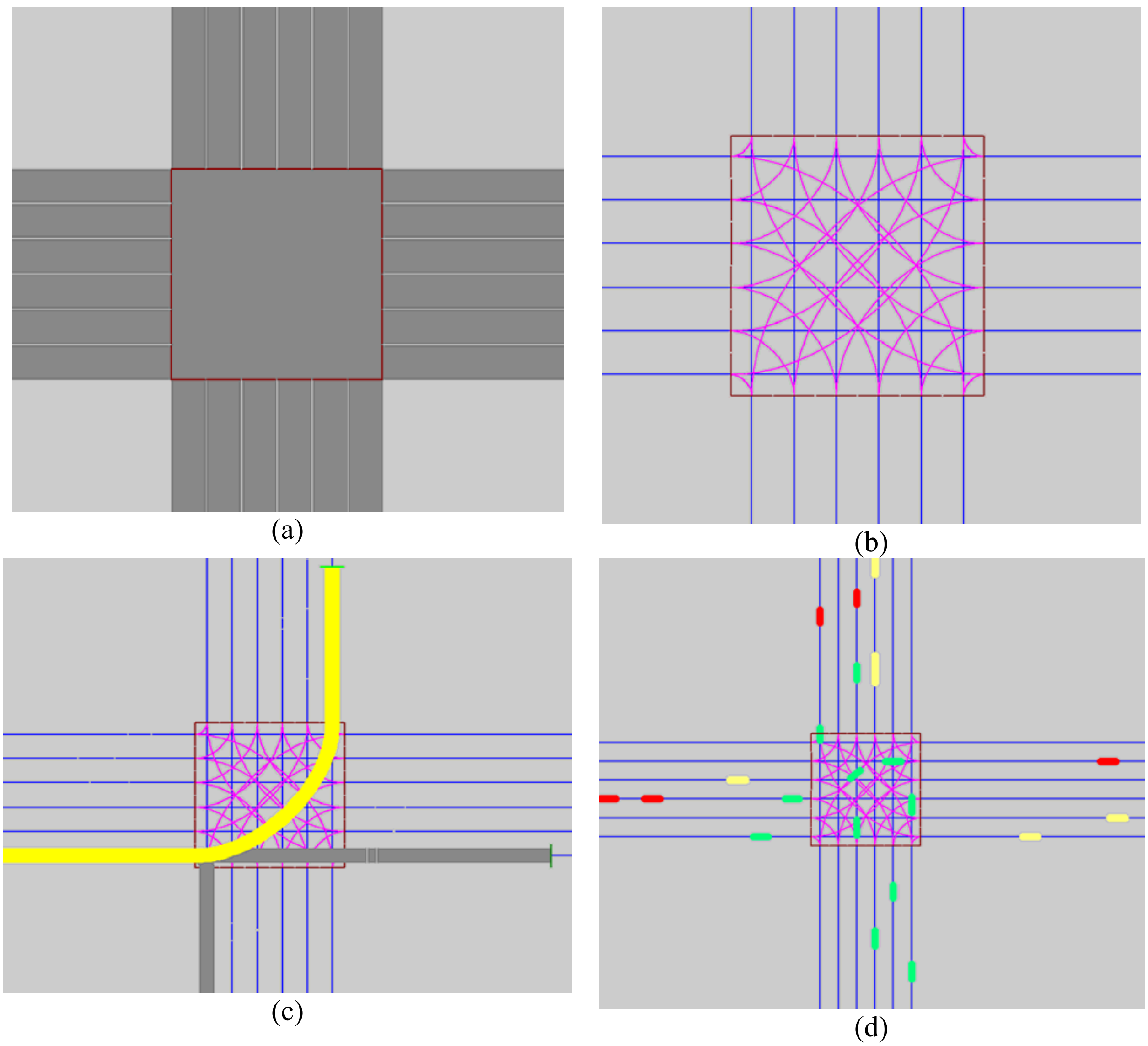


FIGURE 2 Simulation model of the ACUTA intersection.

Implementation of ACUTA using VISSIM's External Driver Model

Before the VISSIM External Driver Model (EDM) was selected for implementing ACUTA, feasibility of using VISSIM COM Interface and VISSIM C2X API was investigated. The C2X API specializes in modeling Car-Car communications with a designated communication range for each vehicle. Therefore, by using the C2X API, it might not be possible to obtain information from all of the vehicles, which is not appropriate for implementing centralized control strategies. The COM interface is quite flexible and versatile in collecting vehicles information and modifying vehicles parameters during the simulation period. However, the COM interface does not provide a direct function to modify a vehicle's acceleration rate. It was also found that executing a command through COM interface may take up to 0.2 sec, which is too long to assure the efficiency of ACUTA simulations.

The VISSIM EDM, on the other hand, can meet all requirements for implementing ACUTA. Through EDM, VISSIM provides an option to bypass and replace VISSIM's internal driving behavior. During a simulation run, VISSIM calls the EDM DLL at every simulation step to pass the current state of each vehicle to the DLL. Therefore, in this research, an intersection manager class was built in the EDM DLL to collect each vehicle's speed, location, vehicle class, maximum acceleration rate, length, width, and many other parameters pertaining to the particular vehicle at each simulation step. The intersection manager processes all reservation requests at the beginning of each simulation step, and passes its decision and the suggested acceleration/deceleration rate to the vehicles in the same simulation step. The vehicle then passes its acceleration/deceleration rate back to VISSIM at the same simulation step, thus the real-time control of each vehicle's acceleration rate is realized.

In summary, EDM offers technical readiness for implementing ACUTA in VISSIM. Key steps for realizing the reservation-based system are discussed in the following subsections.

Modeling the Intersection Mesh in VISSIM

In VISSIM, an intersection can be viewed as an overlapping square between the two crossing roads. The entire intersection area can be divided into a mesh of n by n tiles, as shown in Figure 1. 'n' is the granularity of the intersection mesh. More or fewer tiles can be obtained by adjusting the granularity. Using westbound direction as an example, the green lines with arrows illustrate all possible vehicle paths to traverse the intersection.

In Figure 1, a two-dimensional coordinate system is projected on to the intersection area to facilitate the computation of a vehicle's location. The origin O is located at the southwest corner (C1) of the intersection. The southeast, northeast, and northwest corners are labeled by C2, C3, and C4, respectively. The following sections use this coordinate system as a global coordinate system for computing vehicle's location.

Locating Vehicle's Central Point

A key step in the internal simulation is to compute a vehicle's location at a given simulation time step. For convenience in the following discussion, beginning of time is assumed to be the moment when a vehicle's central point reaches boundary of the intersection area (i.e., at the Point *S* in Figure 3).

In ACUTA, a vehicle maintains a constant speed after its central point enters and before its central point clears the intersection area. Figure 3.a illustrates a case of through movement. The path of a through vehicle is parallel to either of the axes (Figure 3.a) depending upon whether the vehicle is going EB/WB or NB/SB. Assuming that the through vehicle's central point reaches the boundary point *S(x_s,y_s)* at time 0, the coordinates of the vehicle's central point can be calculated using the following equation.

$$\begin{cases} x_t = x_s - L \\ y_t = y_s \end{cases} \tag{3}$$

Where, x_t = x coordinate of the vehicle's central point at time t (ft);
y_t = y coordinate of the vehicle's central point at time t (ft);
x_s = x coordinate of the vehicle's central point at time *0* (ft);
y_s = y coordinate of the vehicle's central point at time *0* (ft);
L = $v \times t$ (ft);
v = speed of the vehicle when it is in the intersection (ft/s); and,
t = any time when the vehicle's central point is within the intersection (s).

For turning movements, the vehicle's path within the intersection can be modeled as arcs whose center coordinates are known (left turn shown in Figure 3.b and right turn shown in Figure 3.c, with the arc centers denoted as P). Assuming that the left-turn vehicle's central point reaches the boundary point *S(x_s,y_s)* at time 0, the coordinates of the vehicle's central point can be calculated using the following equation.

$$\begin{cases} x_t = x_p - R \times \sin(\alpha + \beta) \\ y_t = y_p + R \times \cos(\alpha + \beta) \end{cases} \tag{4}$$

Where, x_t = x coordinate of the vehicle's central point at time t (ft);
y_t = y coordinate of the vehicle's central point at time t (ft);
x_p = x coordinate of the turning arc's center (ft);
y_p = y coordinate of the turning arc's center (ft);
R = $\sqrt{(x_p - x_s)^2 + (y_p - y_x)^2}$, the radius of the turning arc (ft);
α = A/R , radian;
β = $\arctan(\frac{|x_p - x_s|}{|y_p - y_s|})$ (radian);
x_s = x coordinate of the vehicle's central point at time *0* (ft);
y_s = y coordinate of the vehicle's central point at time *0* (ft);
A = $v \times t$, the arc length (ft);
v = speed of the vehicle when it is in the intersection (ft/s); and,
t = Any time when the vehicle's central point is within the intersection (s)

Similarly, assuming that the right-turn vehicle's central point reaches the boundary point *S(x_s,y_s)* at time 0, the coordinates of the vehicle's central point can be calculated using the following equation.

$$\begin{cases} x_t = x_p - R\times\sin(\alpha+\beta) \\ y_t = y_p - R\times\cos(\alpha+\beta) \end{cases} \quad (5)$$

Where, x_t = x coordinate of the vehicle's central point at time t (ft);

y_t = y coordinate of the vehicle's central point at time t (ft);

x_p = x coordinate of the turning arc's center (ft);

y_p = y coordinate of the turning arc's center (ft);

R = $\sqrt{(x_p - x_s)^2 + (y_p - y_x)^2}$, the radius of the turning arc (ft);

α = A/R (radian);

β = $\arctan(\frac{|x_p - x_s|}{|y_p - y_s|})$ (radian);

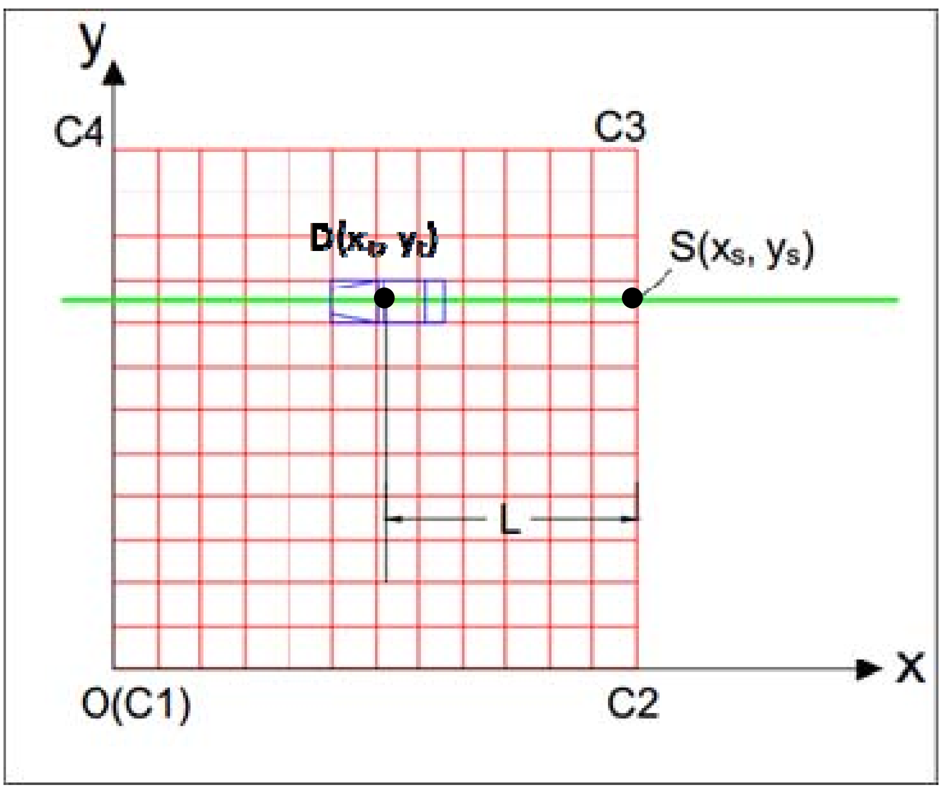
x_s = x coordinate of the vehicle's central point at time *0* (ft);

y_s = y coordinate of the vehicle's central point at time *0* (ft);

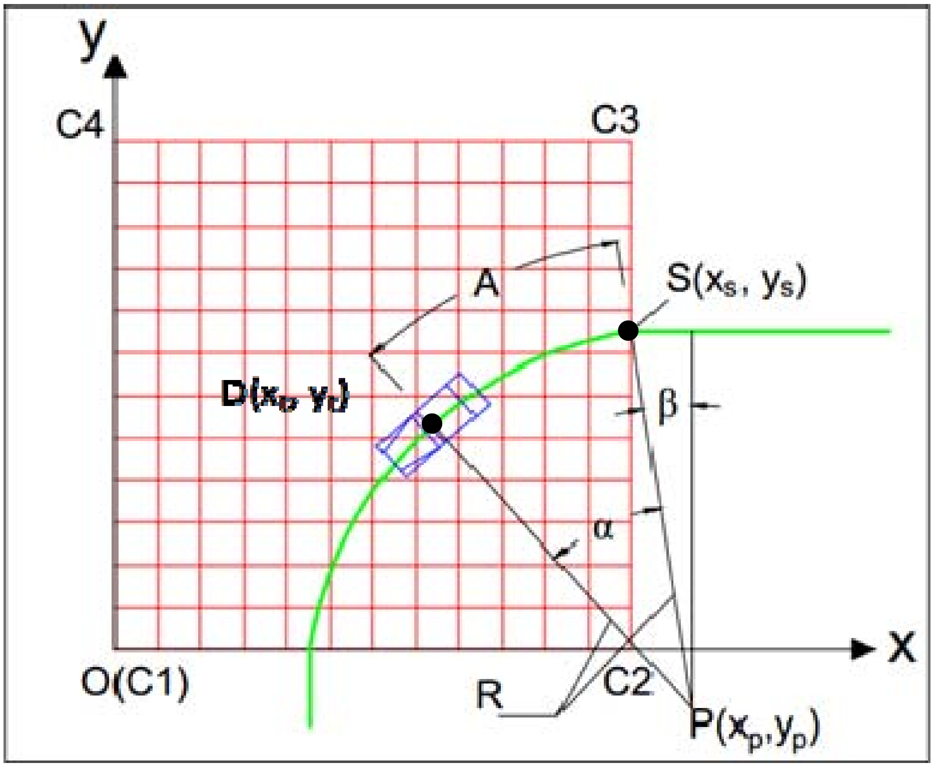
A = $v\times t$, the arc length (ft);

v = speed of the vehicle when it is in the intersection (ft/s); and,

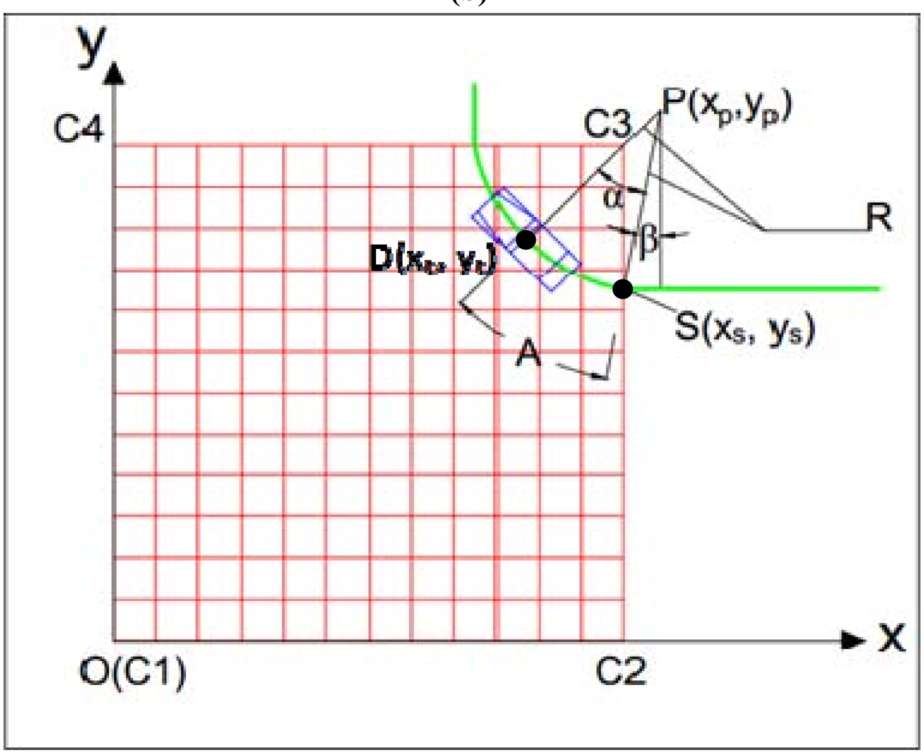
t = any time when the vehicle's central point is within the intersection (s);



(a)



(b)



(c)

FIGURE 3 Determination of vehicle central point location in intersection: (a) through movement; (b) left-turn; (c) right-turn.

Calculating the Coordinates of Vehicle Vertices

Representing a vehicle with its central point is not adequate to describe a vehicle's location. A more comprehensive representation of a vehicle is by coordinates of the vehicle's vertices. Figure 4 illustrates the vehicle's vertices in the intersection mesh. In Figure 4, the length of the rectangle is l_v and the width of the rectangle is w_v, equal to the corresponding vehicle's length and width, respectively. The vertices of the rectangle represents the four corners of a vehicle: head left (PT_{HL}), head right (PT_{HR}), tail left (PT_{TL}), and tail right (PT_{TR}). When the coordinates of the vehicle central point are known, they can be used to calculate coordinates of the four vertices. When the vehicle is parallel to either of the axes, coordinates of the four vertices can be easily calculated using the central point coordinates by subtracting or adding an offset of $l_v/2$ or $w_v/2$. When a vehicle is in a position shown in Figure 4, more complex coordinate transformation is needed.

To conduct the coordinate transformation, a local coordinate system (in comparison with the global coordinate system defined in Figure 1) needs to be defined. The origin of the local coordinate system is located at the central point of the vehicle, with the x axis pointing against the vehicle's traveling direction. To avoid confusion with the global coordinate system, an apostrophe is added to the notations of local coordinate systems (e.g., *x'* and *y'* in Figure 4).

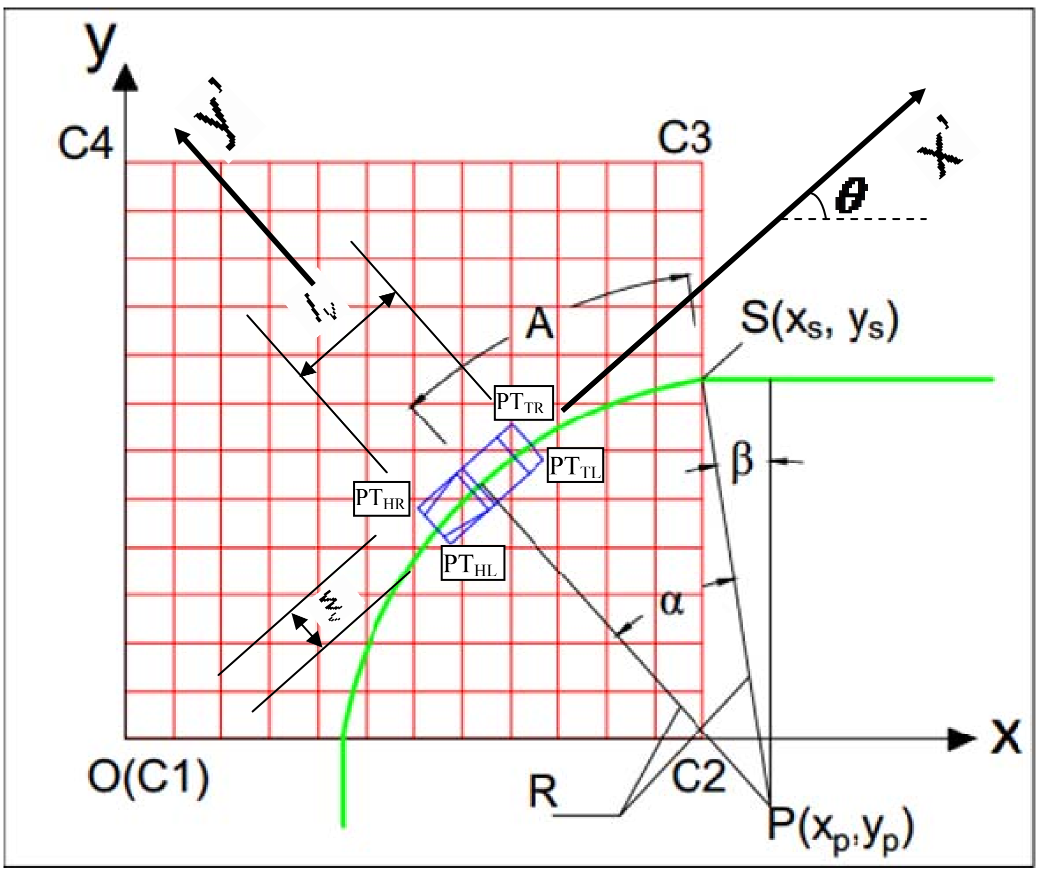


FIGURE 4 Determination of the coordinates of vehicle vertices.

Given a point (x', y') in the local coordinate system, its coordinates in the global system (x, y) can be calculated using a coordinate rotation followed by a coordinate transfer. The formula is given below:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \times \begin{bmatrix} x' \\ y' \end{bmatrix} + \begin{bmatrix} x_t \\ y_t \end{bmatrix} \quad (6)$$

Where, x_t = x coordinate of the vehicle's central point at time t (ft);

y_t = y coordinate of the vehicle's central point at time t (ft); and,

θ = the smallest angle measured counterclockwise from the x axis to the x' axis. In the case of Figure 1, $\theta = \alpha + \beta$ (radian).

Based on Equation 6, the global coordinates of the vehicle vertices can be easily converted from their local coordinates. For example, the local coordinates of the PT_{HR} vertex are (x' = $-l_v/2$, y' = $w_v/2$). By substituting x' and y' with $-l_v/2$ and $w_v/2$ in Equation 6, the global coordinates of PT_{HR} are ($x = -\frac{l_v \cdot \cos\theta + w_v \cdot \sin\theta}{2} + x_t$, $y = -\frac{l_v \cdot \sin\theta - w_v \cdot \cos\theta}{2} + y_t$).

Determining Tile Occupation

When coordinates of a vehicle's vertices are known, the intersection manager needs to determine which tiles are occupied by the vehicle. Figure 5 depicts a vehicle with all occupied tiles highlighted in red. The criterion to determine whether a tile is occupied by a vehicle is: at least one vertex of the tile is inside the vehicle rectangle.

In ACUTA, a vector based method is used to decide whether a point falls in the vehicle rectangle. As shown in Figure 5, four vectors are defined counterclockwise along the vehicle rectangle. The four vectors are $\overrightarrow{v_1}$ (PT_{HR}→PT_{HL}), $\overrightarrow{v_2}$ (PT_{HL}→PT_{TL}), $\overrightarrow{v_3}$ (PT_{TL}→PT_{TR}), and $\overrightarrow{v_4}$ (PT_{TR}→PT_{HR}). A point is within the vehicle rectangle only if it falls to the left of all the four vectors. Given a point $p(x_0, y_0)$ and a vector $\overrightarrow{v_i}$ [(x_{start}, y_{start}) → (x_{end}, y_{end})], p falls to the left of $\overrightarrow{v_i}$ only when the following formula is satisfied:

$$(x_0 - x_{start}) \times (y_{end} - y_0) - (x_{end} - x_0) \times (y_0 - y_{start}) < 0 \quad (7)$$

Where, x_0 = x coordinate of the testing point (ft);

y_0 = y coordinate of the testing point (ft);

x_{start} = x coordinate of the vector's start point (ft);

y_{start} = y coordinate of the vector's start point (ft);

x_{end} = x coordinate of the vector's end point (ft); and,

y_{end} = y coordinate of the vector's end point (ft);

On the other hand, deciding whether a vertex of a vehicle rectangle falls in a tile is relatively easy. The reason is that a tile is bounded by two horizontal lines and two vertical lines. More specifically, any point within the area of a tile can be formulated as:

$$\begin{cases} x_{low} < x_0 < x_{high} \\ y_{low} < y_0 < y_{high} \end{cases} \quad (8)$$

Where, x_0 = x coordinate of the testing point (ft);

y_0 = y coordinate of the testing point (ft);

x_{low} = shared x coordinate of left vertices of the tile (ft);
y_{low} = shared y coordinate of bottom vertices of the tile (ft);
x_{high} = shared x coordinate of right vertices of the tile (ft); and,
y_{high} = shared y coordinate of top vertices of the tile (ft);

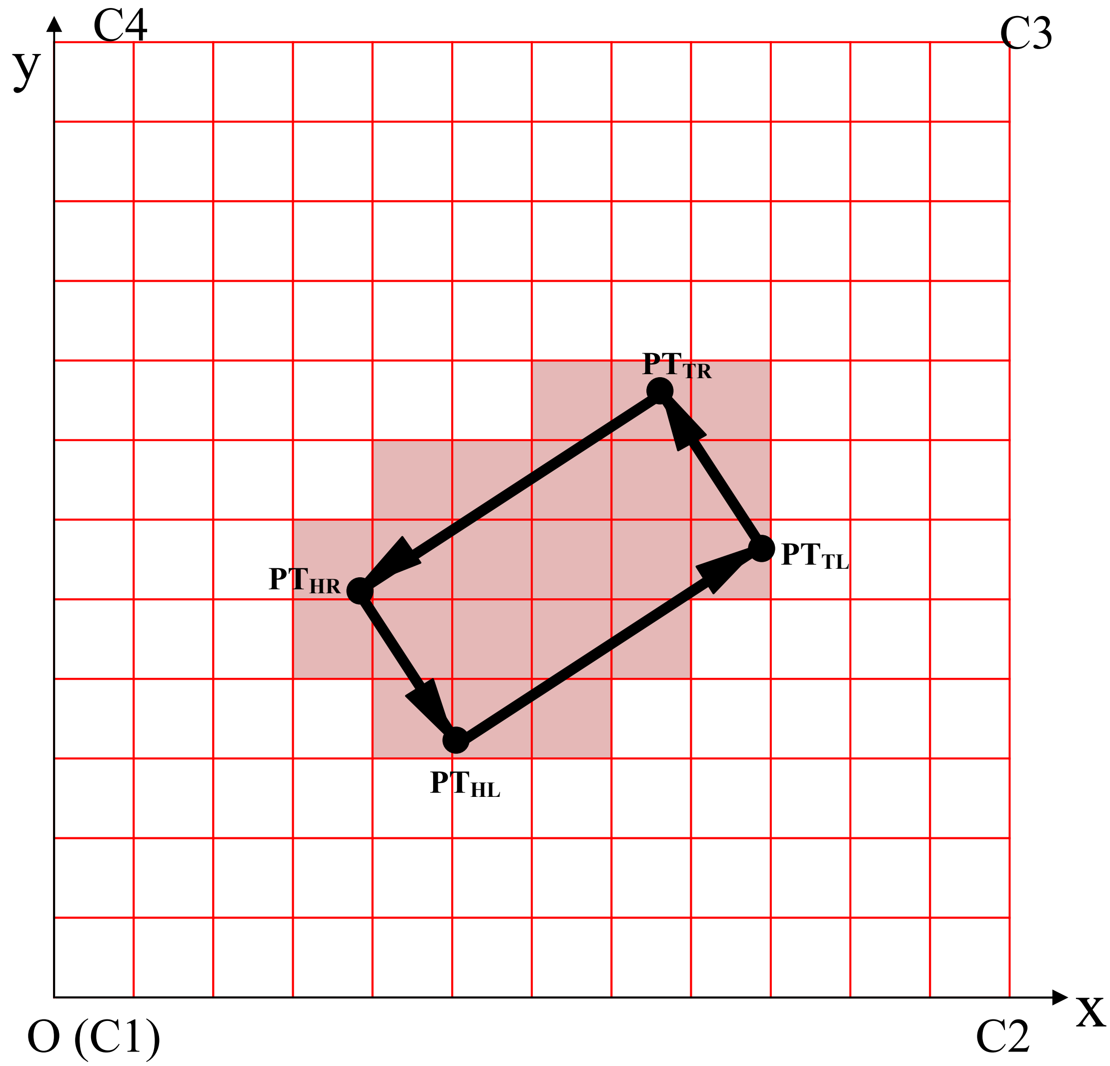


FIGURE 5 Tile occupation by a vehicle rectangle.

In summary, given a tile and a vehicle rectangle, Equations 7 and 8 are used to judge whether a vehicle rectangle has occupied a tile. If any of the four vertices of a tile satisfies Equation 7 or if any of the four vertices of a vehicle rectangle satisfies Equation 8, the tile is considered occupied by the vehicle.

EVALUATION OF ACUTA PERFORMANCE

VISSIM provides a wide range of evaluation tools for its simulation models. The section discusses the evaluation for ACUTA's operational and safety performance by using VISSIM's evaluation functions.

Operational Performance

ACUTA's operational performance under different traffic demand conditions was evaluated using the simulation results, and was further compared with performance of a comparable signalized intersection. The signalized intersection modeled in VISSIM has a left-turn lane, a through lane, and a shared through and right-turn lane designated to each approach. Traffic demands for each movement were identical between ACUTA model and the signalized intersection model. Other parameters except lane configurations are all identical between the two models.

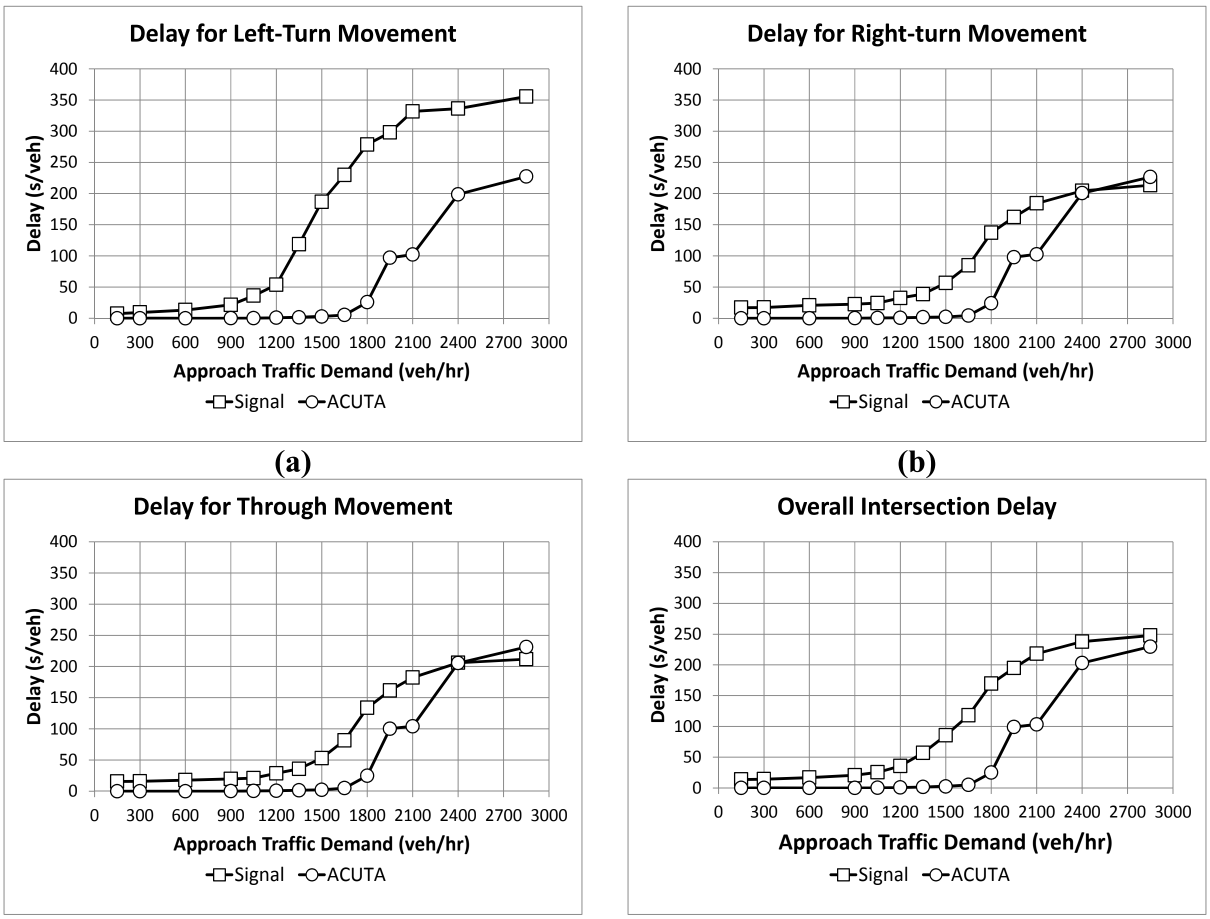
For each traffic demand condition, five simulation runs with different random seeds were performed. Each simulation run lasted 2,100 seconds, with the first 300 warm-up seconds dropped from the evaluation. Specifically, the demand for each approach increased from 150 to 2850 veh/hr to cover the possible range of traffic demands. Proportions of traffic demands for left turn, through and right turn movements were fixed as 25%, 60%, and 15%, respectively for all the simulation runs. Specific demands by movement are summarized in Table 1. For the signalized intersection model, signal timing was optimized using Highway Capacity Software (*22*). Optimization was conducted for each tested traffic demand. Table 1 lists phasing and optimized timings for the signalized intersection along with the corresponding optimized cycle lengths.

TABLE 1 Traffic Demand Inputs and Optimized Timing Plan

Approach Traffic Demand (veh/hr)	Approach Demand by Movement (veh/hr)			Signal Timing Plan				
					Phase Timing (s)			
	LT	*Thru*	*RT*	*Cycle Length (s)*				
150	38	90	23	40	6	6	6	6
300	75	180	45	40	6	6	6	6
600	150	360	90	60	6	16	6	16
900	225	540	135	60	6	16	6	16
1050	263	630	158	60	6	16	6	16
1200	300	720	180	90	10	28	9	27
1350	338	810	203	90	10	28	9	27
1500	375	900	225	110	12	35	12	35
1650	413	990	248	110	12	35	12	35
1800	450	1080	270	110	12	35	12	35
1950	488	1170	293	110	12	35	12	35
2100	525	1260	315	110	12	35	12	35
2400	600	1440	360	120	12	39	13	40
2850	713	1710	428	120	12	39	13	40

Operational performances of ACUTA and optimized signal control were assessed by delays, which were obtained directly from VISSIM's output. Volume-to-capacity (v/c) ratios for left turn, right turn and through movements as well as the overall intersection v/c ratio were also computed for both ACUTA and optimized signal control. When computing v/c ratios, capacity (c) was measured as the maximum throughput among all demand conditions, while volume (v) was directly obtained from VISSIM's output for that specific demand condition.

Based on simulation results, capacities for different movements at the signalized intersection were identified to be 366 veh/hr, 218 veh/hr, and 908 veh/hr for left turn, right turn, and through movements, respectively. Capacity for an entire approach of the signalized intersection was 1480 veh/hr. Capacities for left turn, right turn, and through movements of an approach of ACUTA intersection were measured to be 501 veh/hr, 288 veh/hr, and 1185 veh/hr, respectively. Capacity for an entire approach of ACUTA intersection was 1974 veh/hr. Comparing ACUTA with signalized control, ACUTA successfully increased left turn, right turn and through capacities by 37%, 32%, and 31%, respectively. The overall approach capacity was increased by 33% by implementing ACUTA.



(a) **(b)** **(c)** **(d)**

FIGURE 6 Operational performance of ACUTA with comparison with signalized intersection: (a) left-turn delay, (b) right-turn delay, (c) through delay, and (d) overall intersection delay

TABLE 2 Comparison of Operational Performances between ACUTA and Optimized Signalized Intersection

Approach Traffic Demand (veh/hr)	Optimized Signalized Control								ACUTA (default setting)							
	v/c ratio				Delay (s/veh)				v/c ratio				Delay (s/veh)			
	LT	*Thru*	*RT*	*Overall*	*LT*	*Thru*	*RT*	*Overall*	*LT*	*Thru*	*RT*	*Overall*	*LT*	*Thru*	*RT*	*Overall*
150	0.10	0.10	0.10	0.10	7.36	15.54	17.06	13.70	0.07	0.07	0.07	0.07	0.00	0.00	0.00	0.00
300	0.22	0.19	0.20	0.20	9.26	15.90	17.26	14.34	0.12	0.12	0.12	0.12	0.00	0.00	0.00	0.00
600	0.45	0.39	0.39	0.40	13.12	17.72	20.74	16.90	0.31	0.28	0.31	0.29	0.00	0.00	0.00	0.00
900	0.65	0.59	0.59	0.61	21.52	19.74	22.48	20.62	0.49	0.43	0.45	0.45	0.04	0.04	0.06	0.02
1050	0.75	0.69	0.69	0.71	36.24	21.04	24.38	25.48	0.55	0.51	0.53	0.52	0.26	0.42	0.44	0.38
1200	0.84	0.79	0.79	0.81	53.62	28.70	32.56	35.66	0.62	0.59	0.61	0.60	0.98	0.70	0.76	0.78
1350	0.90	0.88	0.89	0.89	118.72	35.82	38.68	56.86	0.70	0.67	0.67	0.68	1.46	1.48	1.64	1.50
1500	0.92	0.96	0.95	0.96	186.70	53.02	56.64	85.44	0.77	0.76	0.74	0.76	2.82	2.30	2.14	2.42
1650	0.97	0.98	0.99	0.99	230.04	81.46	84.82	117.90	0.84	0.83	0.83	0.83	5.16	4.98	4.32	4.94
1800	0.98	0.98	0.98	0.99	278.72	133.74	137.08	169.42	0.90	0.90	0.87	0.89	25.70	24.78	24.12	24.90
1950	0.98	0.99	0.98	0.99	298.04	161.54	162.30	194.98	0.91	0.91	0.89	0.91	97.00	100.20	97.86	99.04
2100	0.97	1.00	1.00	1.00	331.78	182.34	184.22	218.32	0.99	0.99	0.98	0.99	102.20	104.04	102.52	103.34
2400	0.99	0.98	0.98	0.99	336.26	206.02	204.48	237.88	0.97	0.96	0.96	0.96	198.72	205.50	200.64	203.06
2850	1.00	0.98	0.98	0.99	355.66	211.78	213.28	247.86	1.00	1.00	1.00	1.00	227.24	231.28	226.52	229.58

All evaluation results including the v/c ratios and delays are summarized in Table 2. The signalized intersection reached the 0.99 overall v/c ratio when the approach traffic demand was around 1650 veh/hr, while ACUTA did not reach the 0.99 overall v/c ratio until the approach traffic demand reached 2100 veh/hr. These facts indicate that the ACUTA intersection can process 450 extra vehicles per hour per approach without being oversaturated when compared with the optimized signalized intersection.

Figure 6 depicts the relationships between the delays and traffic demands. Figures 6.a through 6.c illustrate the delays for left turn, right turn, and through movements, respectively. These figures indicate that operational performance of different traffic movements in ACUTA was very balanced as delays for left-turn, right-turn, and through movements were similar under all traffic demand conditions. Overall intersection delay shown in Figure 6.d was computed by taking weighted average of delays for all the movements. According to Figure 6.d, overall intersection delay for ACUTA remained at an extremely low level (under 5 s/veh) when approach traffic demand was less than 1650 veh/hr, while signalized intersection already started to operate at near capacity conditions when approach traffic demand reached 1350 veh/hr. Delay for ACUTA started to increase rapidly when traffic demand reached 1800 veh/hr. However, delays were still significantly less than delays for signalized intersection for approach traffic demands greater than 1800 veh/hr and less than 2100 veh/hr. The superiority of ACUTA became marginal at extremely high approach traffic demands of 2400 and 2850 veh/hr.

Safety Performance

VISSIM can output vehicle's trajectories, which can be directly imported into SSAM to analyze traffic conflicts, enabling evaluation of safety performance of ACUTA. The result of a safety performance study of ACUTA using SSAM is shown in Figure 7, which illustrates an example of a conflict map obtained from SSAM. Only one traffic conflict was found within the intersection during a simulation run of 1800 simulation seconds. This conflict could have been eliminated by incorporating safety buffer, which will be done in the next phase of this study.

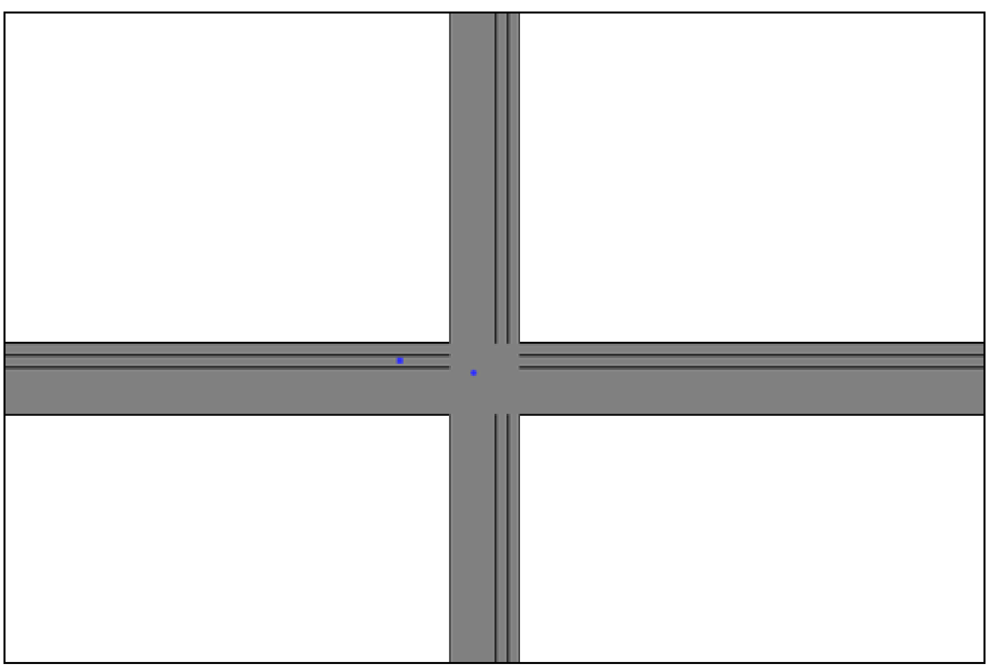


FIGURE 7 Conflict map from SSAM

CONCLUSIONS

A major contribution of this research is the successful implementation of a reservation-based autonomous intersection system in a standard simulation platform, VISSIM. Feasibility of using VISSIM's External Driver Model for modeling autonomous vehicle operations at a centralized controlled intersection through V2I communications has been demonstrated. This type of implementation has not been realized before or even been discussed in literatures. Particularly, key steps for implementing ACUTA in VISSIM are introduced in this paper, providing references to other researchers who are interested in implementing autonomous intersections in a standard simulation platform. By using standard simulation platform, simulation results can become more reliable and trustworthy. Most importantly, operational performance of different autonomous intersection control algorithms can be eventually compared to each other under the same simulation platform.

Evaluation results obtained from VISSIM demonstrated that ACUTA operated with a high efficiency (i.e. intersection delay < 5 s/veh) when the approach traffic demand was less than 1650 veh/hr. In addition, ACUTA had balanced delay distributions for left-turn, right-turn, and through movements than under all traffic demand conditions. Comparing ACUTA with the optimized signal control, ACUTA successfully increased left turn, right turn and through capacities by 37%, 32%, and 31%, respectively. The overall approach capacity was increased by 33% by implementing ACUTA. The analysis on the v/c ratios indicates that the ACUTA intersection could process 450 extra vehicles per hour per approach without being oversaturated when compared with the optimized signalized intersection. Finally, the safety assessment showed only one conflict during a simulation run. All these findings indicate that ACUTA was well modeled in the VISSIM environment. .

ACKNOWLEDGEMENT

The research presented in this paper was funded by the National Center for Freight & Infrastructure Research and Education (CFIRE) at the University of Wiscosnsin-Madison.

REFERENCES

1. Autonomous Car*, Wikipedia,* http://en.wikipedia.org/wiki/Autonomous_car#cite_note-24*, accessed 7/31/2012.*
2. Dresner, K. and Stone, P. (2004) "Multiagent traffic management: A reservation-based intersection control mechanism." *The Third International Joint Conference on Autonomous Agents and Multiagent Systems*, New York, New York, USA, July 2004, 530–537.
3. Dresner, K. and Stone, P. (2005). "Multiagent Traffic Management: An Improved Intersection Control Mechanism." *The Fourth International Joint Conference on Autonomous Agents and Multiagent Systems*, Utrecht, The Netherlands, July 2005, 471-477.
4. Dresner, K. and Stone, P. (2005). "Turning the Corner: Improved Intersection Control for Autonomous Vehicles", *Proceedings of the 2005 IEEE Intelligent Vehicles Symposium*, Las Vegas, Nevada, USA, June 2005.
5. Dresner, K. and Stone, P. (2008). "A Multiagent Approach to Autonomous Intersection Management." *Journal of Artificial Intelligence Research*, 31, 591–656.
6. Dresner, K. and Stone, P. (2008). "Mitigating Catastrophic Failure at Intersections of Autonomous Vehicles", *Proceedings of the Seventh International Conference on Autonomous Agents and Multiagent Systems*, Estoril, Portugal, May 2008, 1393-1396.

7. Shahidi, N., Au, T.C., and Stone, P. (2011). “Batch reservations in autonomous intersection management.” *Proceeding of The 10th International Conference on Autonomous Agents and Multiagent Systems - Volume 3*, Richland, SC, 2011.
8. Au, T.C., Shahidi, N., and Stone, P. (2011). "Enforcing Liveness in Autonomous Traffic Management", *Proceedings of the 25th AAAI Conference of Artificial Intelligence*, 1317-1322.
9. Quinlan, M., Au, T.C., Zhu, J., and Stiurca, N., and Stone, P. (2010). “Bringing Simulation to Life: A Mixed Reality Autonomous Intersection.” *Proceedings of 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2010)*, October 2010.
10. Fajardo, D., Au, T.C., Waller, S.T., Stone, P. and Yang D. (2011). "Automated Intersection Control: Performance of Future Innovation Versus Current Traffic Signal Control", *Transportation Research Record: Journal of the Transportation Research Board, No. 2259*, 223-232.
11. Wu, J., A. Abbas-Turki, and A. El Moudni. (2009) “Intersection traffic control by a novel scheduling model.” *IEEE/INFORMS International Conference on Service Operations, Logistics and Informatics, 2009. SOLI '09*, 2009.
12. Yan, F. Dridi, M., and Moudni, A. E. (2009) “Autonomous Vehicle Sequencing Algorithm at Isolated Intersections”, *Proceedings of the 12th International IEEE Conference on Intelligent Transportation Systems*, St. Louis, MO, USA, October 3-7, 2009.
13. Wu, J., A. Abbas-Turki, A. Corréïa and A. EL Moudni, (2007). “Discrete Intersection Signal Control”, *IEEE International Conference on Service Operations and Logistics, and Informatics*, Aug. 2007.
14. Wu, J., A. Abbas-Turki, and A. El Moudni. (2010) “Contextualized Traffic Controlling At Isolated Urban Intersection.” *The 14th World Multi-Conference on Systemics, Cybernetics and Informatics: WMSCI 2010*, 2010.
15. VanMiddlesworth, M., Dresner, K., and Stone, P. (2008). “Replacing the Stop Sign: Unmanaged Intersection Control for Autonomous Vehicles.” *Proceedings of AAMAS Workshop on Agents in Traffic and Transportation*, Estoril, Portugal, 94–101.
16. Alonso, J., Milanés, V., Joshué, P., Onieva, E., González, C., de Pedro, T., (2011). "Autonomous vehicle control systems for safe crossroads", *Transportation Research Part C*, 19, 1095–1110.
17. Ball, R. and Dulay, N. (2010). "Enhancing Traffic Intersection Control with Intelligent Objects", *First International Workshop the Urban Internet of Things*, 2010.
18. Vasirani, M. and Ossowski, S. (2009). "Evaluating Policies for Reservation-based Intersection Control." *Proceedings of the 14th Portuguese Conference on Artificial Intelligence (EPIA'09)*, Vol. 14, 2009.
19. Surrogate Safety Assessment Model and Validation: Final Report, *FHWA-HRT-08-051*, Federal Highway Administration, Washington, D.C., 2008.
20. *DSRC: The Future of Safer Driving*, U.S. Department of Transportation, Research and Innovative Technology Administration. http://www.its.dot.gov/factsheets/pdf/JPO-034%20DSRC%20V5.5%20F.pdf (accessed 11/14/2012)
21. Hu, B. and Gharavi, H. (2011) “A Joint Vehicle-Vehicle/Vehicle-Roadside Communication Protocol for Highway Traffic Safety”, *International Journal of Vehicular Technology, 2011,* doi:10.1155/2011/718048
22. Highway Capacity Software 2010, McTrans, University of Florida.