GENERAL-PURPOSE CONSTRUCTION SIMULATION AND VISUALIZATION TOOLS FOR MODELLING AND ANIMATING URBAN VEHICULAR TRAFFIC OPERATIONS

SUBMITTED: February 2008
REVISED: October 2008
PUBLISHED: December 2008 at http://www.itcon.org/2008/35
EDITOR: John Messner

Brian W. Timm, Research Engineer
Microsoft Corporation, Redmond, WA 98052, USA
Formerly BSE Student of Computer Science and Engineering, University of Michigan
email: Brian.Timm@microsoft.com

Vineet R. Kamat, Assistant Professor of Civil and Environmental Engineering
University of Michigan, Ann Arbor, MI 48109, USA
email: vkamat@umich.edu
http://pathfinder.engin.umich.edu

SUMMARY: *This paper illustrates how general-purpose simulation and visualization tools primarily designed to study construction operations can be effectively used to model and animate vehicular traffic flows in urban transportation networks. A high-fidelity simulation model and a corresponding 3D visualization describing traffic flow at a busy 3-way signalized intersection are presented. The model and animation demonstrate how simulation and visualization can be used to study vehicle interaction in a transportation network, given data on the expected volume of traffic over intervals of time. The characteristic that distinguishes the presented work is the capability to model and animate vehicular traffic operations with high fidelity using general-purpose discrete event simulation and visualization tools typically used to model and animate construction operations. The focus of the research has been on evaluating the capabilities of state-of-the-art construction simulation and visualization tools in being able to accurately model and animate traffic operations. The presented model simulates several different scenarios and traffic control strategies, and can be a valuable tool in the efficient design of signalized traffic intersections. The solution to the problem is modelled in detail using the simulation tool STROBOSCOPE (MARTINEZ, 1996) and has been animated in 3D using VITASCOPE (Kamat, 2003). The obtained results highlight the potential that general-purpose tools, even if originally designed for specific purposes, can be effective in studying operations involving the interaction of entities in multiple domains such as construction and traffic flow, thereby enabling their use in studying mixed operations such as the effect of roadway construction (e.g. lane closures, slow construction vehicles, etc.) on existing vehicular traffic.*

KEYWORDS: *Animation, Discrete-Event Simulation, Traffic Operations, Verification, Validation, Visualization.*

1. INTRODUCTION

Intersection design and traffic signalization are important areas of practice for city planners and transportation engineers (Chang and Su 1995). City planning officials plan and analyze the design of intersections in order to optimize the flow of traffic and minimize time lost in traffic discharge (Ceylan and Bell, 2004). The advantages of such optimization include a decrease in gasoline usage and harmful emissions from idling vehicles, and reductions in lost time from traffic deadlock. In complex, dynamic areas of high traffic flow, these negative effects from an inefficient design are more pronounced, and the optimization and design process is more critical and complicated (Mirchandani and Head, 2001).

Simulation and 3D animation systems are widely used tools in the process of intersection design and analysis (Bayarri et al 1996, Yang and Koutsopoulos, 1996). Specialized tools exist which can execute and even optimize simulations of both single intersections and multiple nodes in a larger network. These tools also provide the capability of displaying the simulations in a three-dimensional environment, allowing the designer to visualize the flow of traffic exactly as it would be at the real world intersection. This paper evaluates the extent to which urban traffic operations can be modelled and animated using general-purpose simulation and visualization tools typically used to study construction operations. A three-way signalized traffic intersection is studied by modelling traffic flow with several intersection control methods, beginning with stop signs, and extending to more complex intersections governed by both pre-timed and actuated traffic signals. The simulation models were implemented using the STROBOSCOPE simulation system (Martinez, 1996), and were then animated in 3D using the VITASCOPE visualization system (Kamat, 2003).

2. INTERSECTION DESIGN AND MODELING

Stop signs are a well known method of control at an intersection, often used in areas with low traffic volume (Naylor and Graham, 1997). Intersections can also be designed with several different types of automated traffic controllers: pre-timed, semi-actuated, and fully actuated. Pre-timed traffic controllers operate with a fixed cycle length, with the red, yellow, and green lights each taking a predefined interval of the cycle. Traffic volume is different at various periods in the day, and different cycle lengths can be programmed to take effect throughout the day. The operation of the lights will, however, not change in response to the current volume of traffic. Actuated traffic controllers, on the other hand, use vehicle-sensing equipment to determine when right-of-way is being demanded by other vehicles. Semi-actuated controllers typically monitor side streets, interrupting a continuous green light on the main street when vehicles approach, while fully actuated controllers monitor all approaches at the intersection. These actuated controllers are considered more efficient because they can lengthen or omit cycles in response to current traffic volume.

Intersection design, in addition to traffic signalization, is a necessary and desirable step of transportation planning to assure that vehicle and pedestrian traffic is efficiently coordinated, resulting in minimal congestion and delays. Inefficiently designed intersections and the resulting congestion create numerous problems for citizens, both socially and economically, including time lost waiting in traffic, high vehicle emissions that harm the environment, and increased gasoline consumption. Since networks of intersections can become very complex in populated areas, it is advantageous for city planners to be able to simulate and visualize the flows of traffic through these networks under various conditions. The task of intersection modelling is useful for the planning phases of new roadways, but it can also be used to optimize existing networks which may be problematic or congested.

The efficiency of an intersection is generally determined by several different measurements. Examples include the average delay/waiting period of a vehicle at an intersection, which may include time spent decelerating on the approach, time queued at a stop, and time spent waiting to discharge on the departure. Other important factors are the average length of the queue, whether that is the number of vehicles stopped at a red light which all discharge on the next green light, or the number of vehicles stopped at a stop sign, each discharging individually. These measurements must be accurately recorded by the simulation as it executes in order to determine the optimal parameters for real world operation.

Numerous factors must thus be considered when designing a simulation to accurately reflect real world conditions. Vehicle flow rate (vehicles/hour), probable direction of departure (left, straight, right), discharge headway (time between successive vehicles reaching intersection), and average time to clear the intersection are all important characteristics which may vary within each day. Specialized programs exist which execute traffic intersection simulations using real-world data, either observed or estimated. Programs like SYNCHRO, SimTraffic, and 3D Viewer, all designed by Trafficware, simulate and visualize individual intersection operations and their effects on the local traffic network. For each approach, the quantity and direction of approach lanes, estimated vehicle flow rate, and phasing data (cycle length, clearance times) are some of the necessary parameters that are used to create a virtual model of intersection activity. However, these systems have some limitations. Although they are very valuable and effective for traffic operations modelling, they are special-purpose tools. They are designed to exclusively model vehicular traffic performance and behaviour. They cannot model the interaction of operations in different domains, for example traffic flow and construction operations simultaneously.

The models that this paper discusses have been created using general-purpose simulation and visualization tools. The STROBOSCOPE simulation language is used to define the rules governing the intersection, allowing the user to set the necessary input parameters. The simulation records statistics on the average and maximum wait times, queue length, and number of vehicles on each path. The running simulation prints a chronological trace of statements corresponding to each event in the simulation. This trace file is then used by VITASCOPE, along with CAD models of the objects, to create a visualization of the simulated operations. This allows the user to confirm that the simulation model is accurate and operating in a way that reflects real-world conditions.

3. SIMULATION OF TRAFFIC OPERATIONS: TECHNICAL APPROACH

The presented simulations recreate a 3-way intersection, with traffic that approaches the intersection travelling North, West, and East, and departs travelling South, West, or East. The inbound and outbound directions are characteristic properties of each distinct vehicle object, defined as a subtype of the Characterized Resource Type *Vehicle* in STROBOSCOPE. This simulation divides the three incoming paths into distinct simulation sequences. In this discussion, the events for the Northbound lane will be highlighted, while analogous activities execute in each of the other two paths. The current model is configured to continue simulating for ½ a day (*VARIABLE SecsToSimulate 43200*) in units of seconds. This operation time can be changed to accommodate shorter or longer periods of observation.

All simulations include *TotalFlow*, a user-defined variable for total traffic flow (vehicles/hour) at the intersection. In addition, the user can define parameters for vehicle flow from each of the 3 approaching directions (*NorthBoundFrac, EastBoundFrac, WestBoundFrac),* and can dictate the likelihood of a vehicle leaving the intersection in any of the two valid directions of departure (*NBEastRight, NBWestLeft, EBSouthRight, EBEastStraight, WBSouthLeft, WBWestStraight*). These values are used as the parameters for a uniformly distributed random variable which dictates the interval between any two consecutive vehicles on the same path, as will be described later. These parameters, when gathered by observing the real-world intersection, make the simulated counterpart a more accurate model, yielding more applicable results.

Figure 1 presents the initial model representing a stop-sign scenario. The progression of each vehicle object through this network is linear, moving through events in its respective branch until it departs the environment. This simulation begins with the approach of each vehicle from its respective direction. Assuming no U-turns, each vehicle has 2 available directions of departure. A cycle exists between the *CQueueNBE* and *CCombNBE*, for Northbound vehicles turning right (East), and another cycle exists between *CQueueNBW* and *CCombNBW*, for Northbound vehicles turning left (West). Each directional Combi draws a vehicle from the corresponding directional Queue, and executes for a uniformly distributed duration. A single vehicle is placed in each Queue at the start of the simulation, and is replaced at the completion of the Combi event. The uniform distributions for the *CCombNBE* event and the *CCombNBW* event can be adjusted so that the correct ratio of vehicles depart in each direction, and the total number of vehicles in all three paths equal the full volume of traffic flow. This formula accounts for varying traffic flow volumes in the appearance of approaching vehicles:

$$3600/(TotalFlow*NorthBoundFrac*NBEastRight) \sim\sim \left\{ \frac{\frac{\sec}{hr}}{\frac{vehicles}{hr}*(\%NB)*(\%NBE)} \right\}$$

(%NB) is the percent of total vehicles on a Northbound approach, and (%NBE) is the percent of Northbound vehicles that depart to the East. Therefore, the duration of the Combi is in units of X *sec/vehicle*, distributed uniformly, indicating that there are generally X seconds between the appearance of consecutive Northbound, East-departing vehicles. All of these values are defined by the user, based on an analysis of real-world traffic flows, to approximate the existing traffic conditions. Alternatively, a model could be created which offers a single approaching branch of traffic, with vehicles probabilistically selecting a direction of travel according to the observed ratios.

These preliminary cycles govern the intervals between approaching vehicles. After exiting the *CCombNBE* or *CCombNBW* Combis, the vehicle is considered to be starting its approach to the intersection. At this point in the simulation, both departing vehicle subtypes merge into the single Northbound traffic stream. Since each subtype is generated independently, they must merge in an orderly fashion. The *NBFilterQ* Queue accepts a vehicle at the start of its approach, ordering multiple vehicles by the time of entry into the Queue. The *NBFilterC* Combi, with a duration of 1 second, draws vehicles from the Queue. The statement (*SEMAPHORE NBFilterC '!NBFilterC.CurInst';*) guarantees that only 1 instance of the *NBFilterC* is active at a time, providing a 1 second buffer between two vehicles to prevent them from beginning their approach at the same instance. At this point, the simulation differs depending on the method of control being simulated.

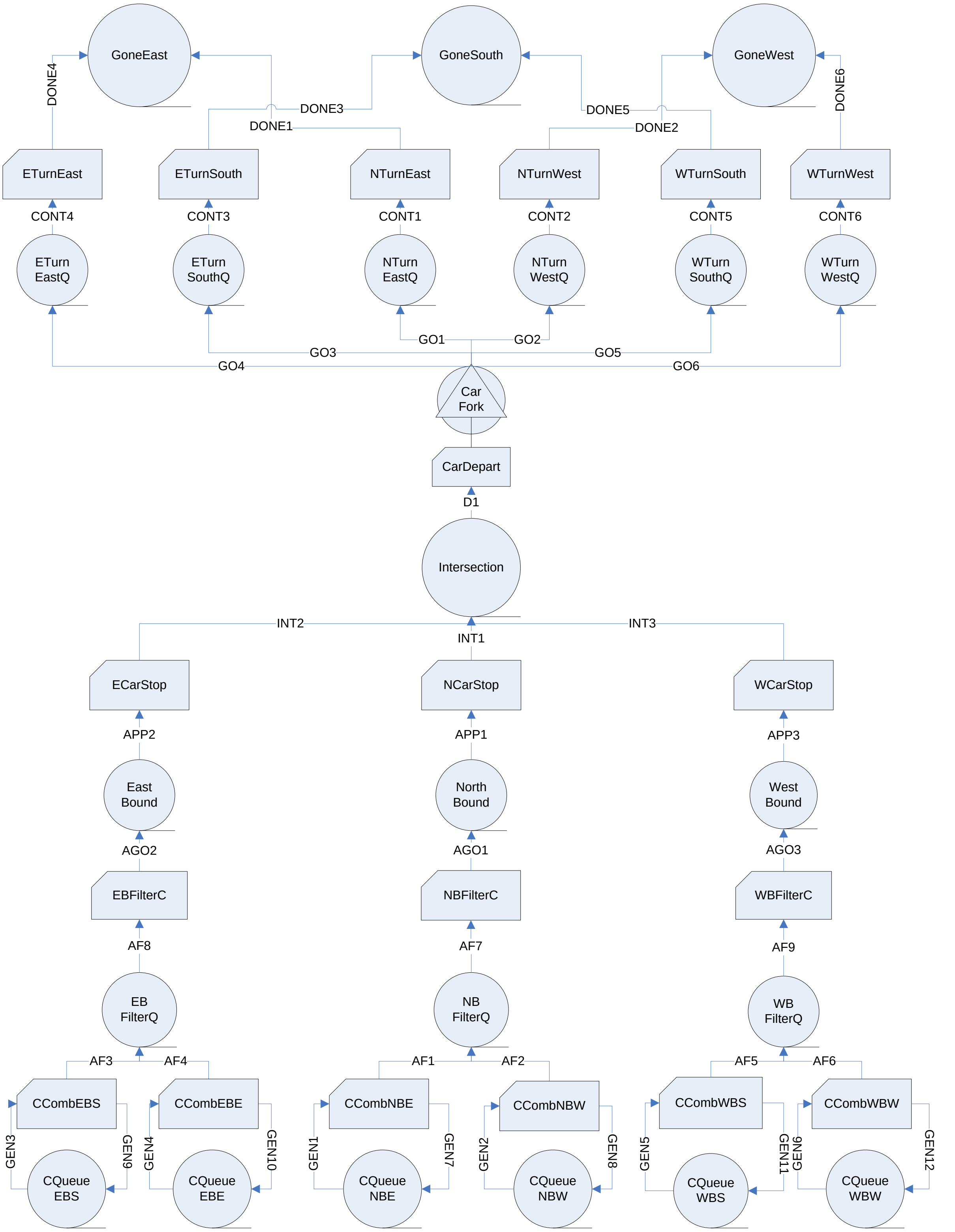

FIG. 1: Stroboscope Network for Stop Sign Scenario

3.1 Stop Sign Model

The initial stop sign model simulates activity at a three-way intersection governed by stop signs. Each vehicle comes to a complete stop at the sign, with right-of-way being given to the vehicle with earliest time of arrival at the sign. After Northbound vehicles have been filtered into a single approaching stream of traffic, they enter the *Northbound* Queue, and proceed to the *NCarStop* Combi. This event, with a 2-second duration, forces the vehicle to pause at the stop sign, even if all other directions are clear.

At this point, the simulation must account for right-of-way among multiple vehicles stopped at the sign, coming from one or more of the inbound branches. All three Combi activities (*NCarStop, ECarStop, WCarStop*) feed into the *Intersection* Queue, where the vehicles are ordered chronologically by the time of their entrance into the Queue (or equivalently, their release from the *CarStop* event). The *CarDepart* Combi activity draws the first vehicle from the *Intersection* Queue, holding it for a duration of 1 second to account for the time to leave the stop sign and safely clear the intersection. The command (*SEMAPHORE CarDepart '!CarDepart.CurInst';*) ensures that only one *CarDepart* activity is active at a time, so no vehicle can enter the intersection while another vehicle is passing through. The other cars are held in the *Intersection* Queue until the intersection is clear, when the *CarDepart* Combi draws the next vehicle until empty. Figure 2 graphically depicts a snapshot of the running animation describing this scenario.

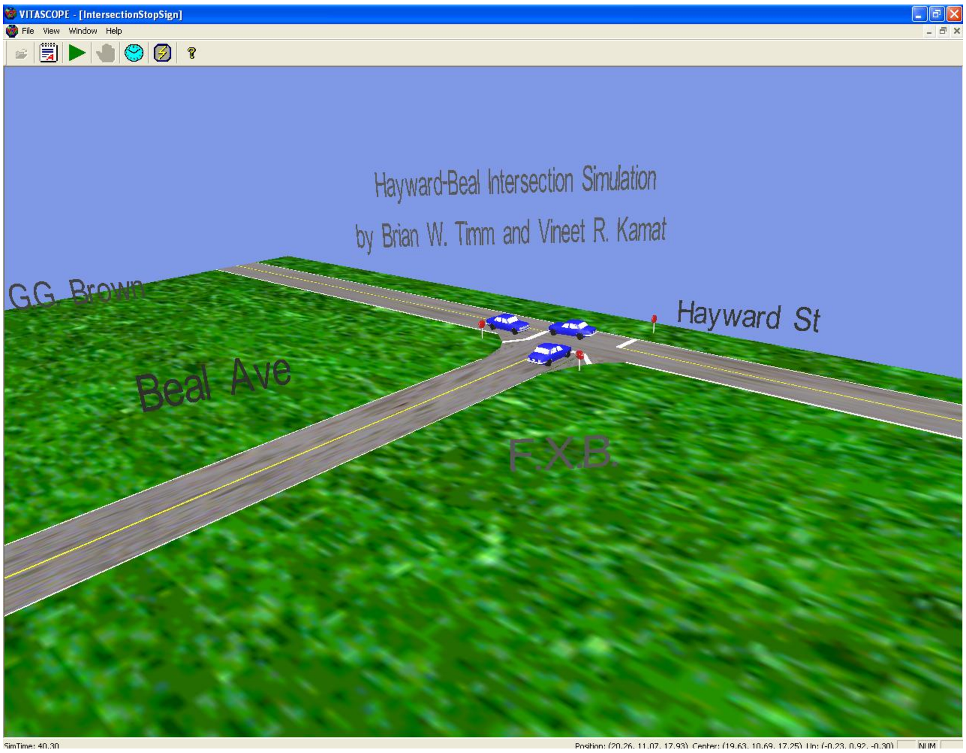

FIG. 2: Vehicles at Stop Sign Yielding to Westbound Car

After clearing the intersection, vehicles enter the *CarFork* Fork, where there are 6 possible paths to follow. The flow control splits back into three distinct branches, as determined by the vehicle's characteristic properties: *To* (Direction of Departure) and *From* (Direction of Approach). For example, the Northbound vehicle will leave the *CarFork*, and enter either the *NTurnEastQ* Queue or the *NTurnWestQ* Queue, displayed in the statements:

```
STRENGTH GO1 'CarDepart.Vehicle.From==1 & CarDepart.Vehicle.To==2';

STRENGTH GO2 'CarDepart.Vehicle.From==1 & CarDepart.Vehicle.To==3';
```

In this case, the *STRENGTH* will evaluate to 1 if both conditions are true, and that path will be taken by the simulation. If either comparison is false, it will evaluate to 0, and that path will not be taken. The *NTurnEast* Combi will draw the vehicle from the *NTurnEastQ*, and the *NTurnWest* Combi will draw a vehicle from the *NTurnWestQ*. These Combi events account for a 2-second duration, from the time a vehicle clears the intersection until it clears the vicinity of the intersection on its outbound path. Finally, the vehicle enters either the *GoneEast* or *GoneWest* Queue, where they are held until the simulation finishes executing.

3.2 Pre-Timed Traffic Light Model with Protected Left Turn

This model contains a set of automated traffic lights which govern the right-of way for vehicles at the intersection, and is graphically depicted in Figure 4 on the next page. The right-of-way cycle is as follows:

1) All Northbound vehicles (Departing East and West)

2) Protected left-turn for Westbound vehicles (Departing South)

3) All Eastbound vehicles (Departing East and South), Remaining Westbound vehicles (Departing West)

Cycle times for most lights are controlled by a user-defined variable *LightLength.* Each light remains green for a period of (*0.85*LightLength*), and the yellow lasts for (*0.15*LightLength*). At all other times, the lights remain red. The protected left turn light for Westbound traffic has a full cycle time of (*0.5*LightLength*), but the green and yellow lights occupy the same fraction of that cycle. Figure 3 shows the sub-network of the model that describes the right of way cycle for the protected left turn scenario.

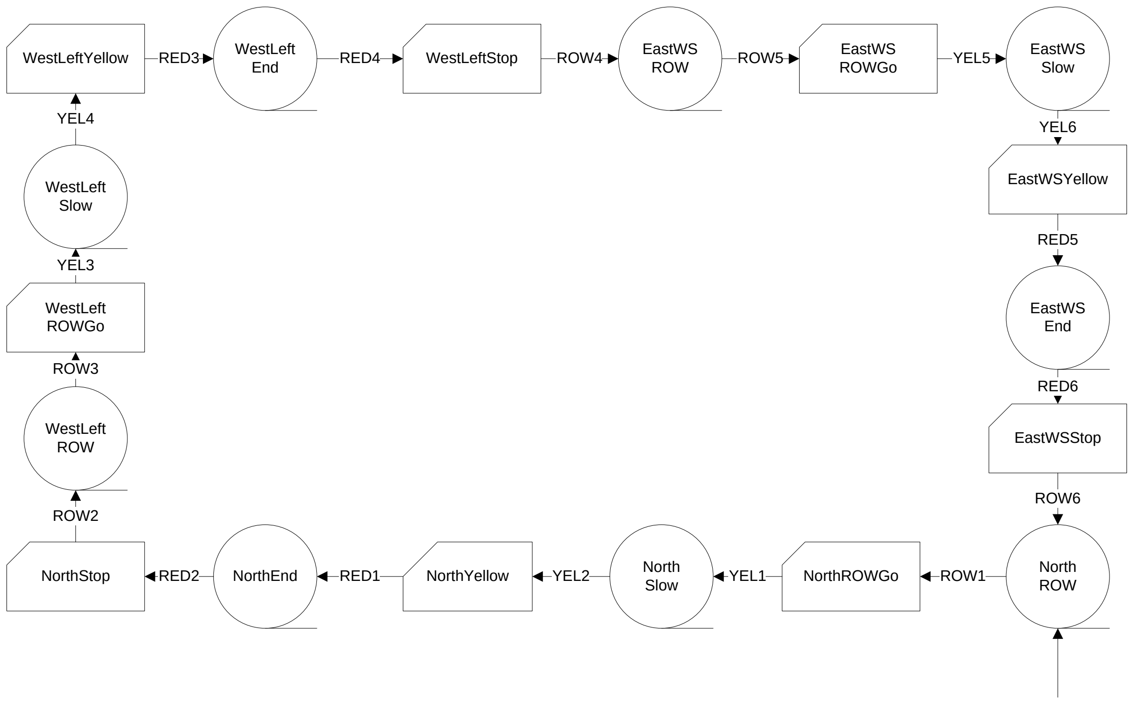

FIG. 3: RightOfWay Cycle for Pre-Timed Model with Protected Left Turn

Northbound vehicles approaching the intersection enter the *NBLight* Queue, where they are held until an *NBCarDepart* Combi event is initialized. The vehicle departing events (*NBCarDepart, EBCarDepart, WBSCarDepart, WBWCarDepart*) are governed by a *RightOfWay* object cycle. This cycle, shown in Figure 3, is a path of alternating Queue and Combi events meant to mirror the lights. The each **ROWGo* Combi corresponds to a green light for that lane, the **Yellow* Combi reflects a yellow light, and the **Stop* Combi is a 1.5 second period when all lights are red, allowing the intersection to clear.

FIG. 4: Stroboscope Network for All Light-Controlled Intersection Models

This independent cycle governs vehicle departures in the following way. A vehicle on Northbound approach passes through the *NorthBound* Queue, entering the *NBApp* Combi with a 2-second duration for the vehicle moving toward the light. After completing the approach, the vehicle is held in the *NBLight* Queue until the *NBCarDepart* event is able to initialize. Just as before, only one *NBCarDepart* Combi can be active at a time, but the semaphore statement is updated with two additional constraints:

```
SEMAPHORE NBCarDepart '(NorthROWGo.CurInst | NorthYellow.CurInst) &
          !NBCarDepart.CurInst';
```

The *NBCarDepart* Combi can begin only if a *NorthROWGo* or *NorthYellow* Combi is active. Therefore, a vehicle can only pass through the intersection when the *RightOfWay* cycle signifies either a green or yellow light. At all other times, vehicles will accumulate in the *NBLight* Queue until the next green light, when the *RightOfWay* object gets back to the correct point in the cycle. Figure 5 graphically depicts a snapshot of the running animation describing one of these scenarios where vehicles make a protected left turn.

FIG. 5: Vehicles Making a Protected Left Turn

3.3 Pre-Timed Traffic Light Model without Protected Left-Turn

This simulation maintains an identical vehicle flow diagram, seen in Figure 4, but removes left turn protection for Westbound vehicles. This allows the *RightOfWay* cycle to be shortened, as shown in Figure 6. The right-of-way cycle is now:

1) All Northbound vehicles (Departing East and West)

2) All Eastbound and Westbound vehicles

In this scenario also, the user-defined variable *LightLength* determined the length of each light cycle. In these cases, the *North** and *EastWest** paths represent the lights for all Northbound traffic, and the light for all Eastbound and Westbound traffic, respectively.

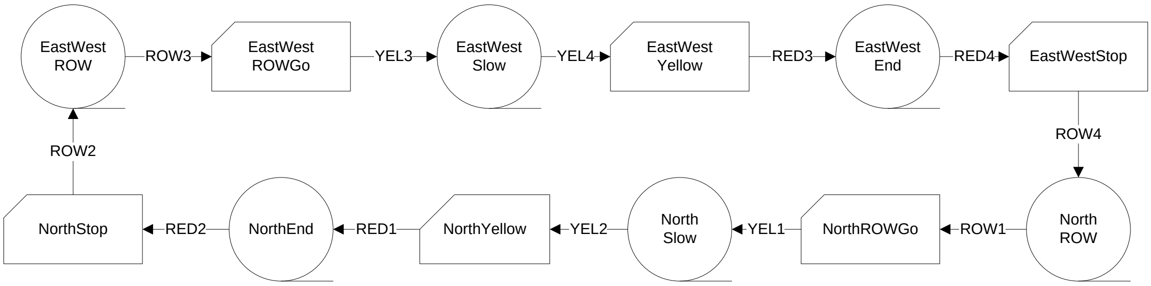

FIG. 6: RightOfWay Cycle for Pre-Timed Model with Permitted Left Turn

Since the Westbound, left turn lane is no longer protected, these vehicles will always yield to any Eastbound vehicles present at the light when it is green or yellow. Similarly, if a Westbound vehicle is turning left and has not cleared the intersection, any approaching Eastbound vehicle must wait until they have done so. This interaction is governed by adding constraints to both the *WBSCarDepart* and *EBCarDepart* Combis:

```
SEMAPHORE  WBSCarDepart  '(EastWestROWGo.CurInst  |EastWestYellow.CurInst)
      & !WBSCarDepart.CurInst &  !EBCarDepart.CurInst & !EBLight.CurCount';

SEMAPHORE  EBCarDepart  '(EastWestROWGo.CurInst  |  EastWestYellow.CurInst)
                         & !EBCarDepart.CurInst & !WBSCarDepart.CurInst';
```

The first three conditions do not change from other branches, requiring the vehicles to have *RightOfWay* in the form of either a Green or Yellow light, and ensuring a safe gap between consecutive vehicles entering the intersection. The additions are different for each of the Combi events. The *EBCarDepart* semaphore restricts Eastbound vehicles from departing if a Westbound vehicle is turning left (*!WBSCarDepart.CurInst*). Similarly, the *WBSCarDepart* semaphore assures that Westbound vehicles will not enter the intersection if any Eastbound vehicle is passing through (*!EBCarDepart.CurInst*), or if any Eastbound vehicles are at the light, waiting to discharge (*!EBLight.CurCount*). Note that vehicles turning left will not wait an unreasonable length of time for vehicles that are "approaching" from the East (*EBApp.CurInst*), but will yield to vehicles that have already reached the intersection (*EBLight.CurCount*). Figure 7 portrays vehicles waiting to make an unprotected left turn.

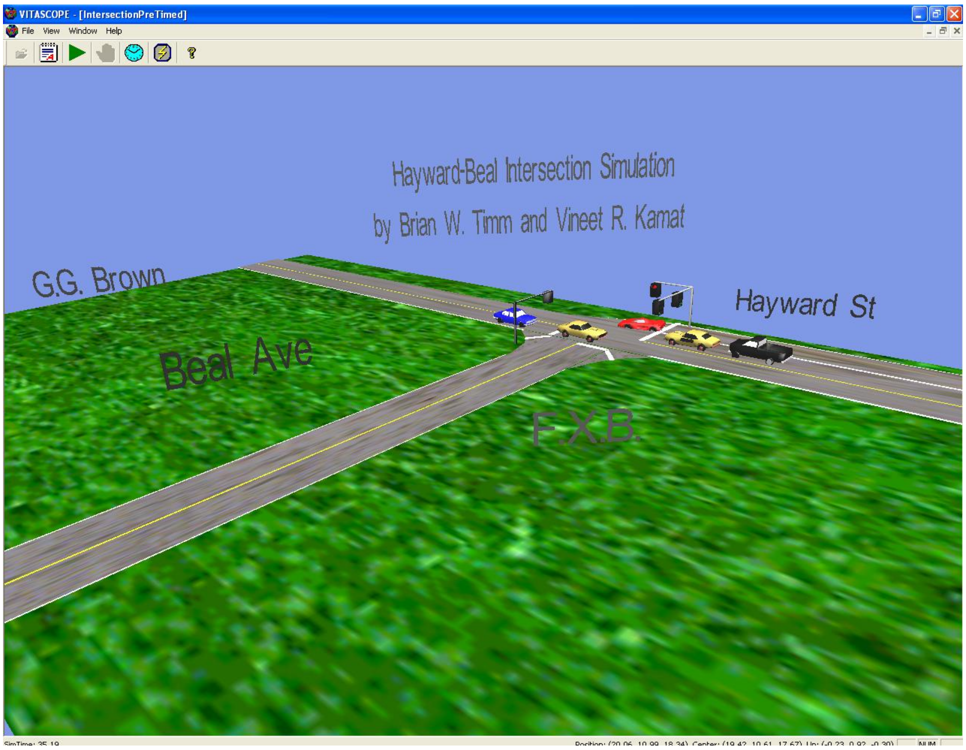

FIG. 7: Vehicles Wait to Make Unprotected Left Turn

3.4 Actuated Traffic Light Model

In this simulation, the permitted left turn lane remains, operating under the same controlling dynamics as described in the previous section. Rather than designating a single *LightLength* variable, this actuated model allows the user to define the *MaxQueueLength* variable. In place of a predefined duration, the green light Combi events are replaced by a set of semaphore statements for the yellow light Combi events:

```
SEMAPHORE NorthYellow     'EBLight.CurCount + WBSLight.CurCount +
                           WBWLight.CurCount) >= MaxQueueLength';

SEMAPHORE EastWestYellow 'NBLight.CurCount >= NBQueue';
```

An example is as follows: A Northbound vehicle is presently in the *NBLight* Queue, meaning it is waiting at a red light. The *RightOfWay* object is presently in the *EastWestROW* Queue, which represents a green light for those lanes. This change was made because the light must remain green for an undefined length of time. Instead, when the number of vehicles waiting at the light exceeds the value of *MaxQueueLength*, the *EastWestYellow* Combi draws the *RightOfWay* object out of the Queue. This Combi, with a defined duration, corresponds to the yellow light, followed by the initialization of an *EastWestStop* Combi to allow the intersection to clear. Then, right-of-way passes to the Northbound lane (*NorthROW* Queue), which remains green until the same set of criteria are met and the yellow light is initialized. Figure 8 graphically depicts the sub-model that mimics this behaviour. Figure 9 depicts accumulating cars in the east-west corridor that will trigger the reclaiming of the right of way from the Northbound vehicles.

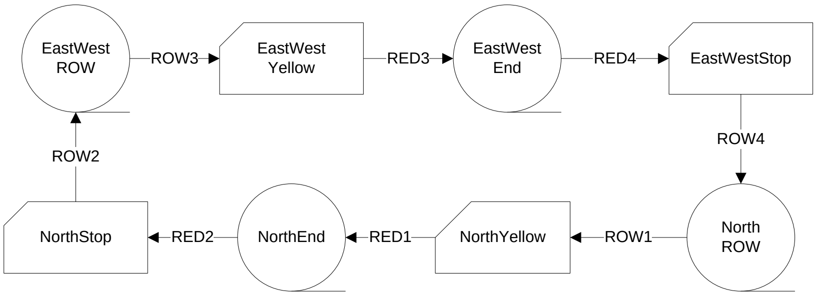

FIG. 8: RightOfWay Cycle for Model with Actuated Lights

4. VALIDATING SIMULATED OPERATIONS USING 3D ANIMATION

The use of a discrete-event simulation tool often leaves the user unable to fully authenticate the model and the validity of the results (Khoury et. al. 2007). Visualizing the execution of the model allows the user to verify that the model is free of errors, and to assure that the simulation accurately reflects real-world conditions. In addition, the ability to witness operations in a virtual representation of the real-world environment may expose additional complications which need to be considered or included in the model. The STROBOSCOPE model was programmed to generate a statement for each discrete event in the simulation, creating a trace file which can be executed and replayed in 3D with VITASCOPE.

VITASCOPE is a general-purpose, user-extensible 3D animation system designed for visualizing simulated processes in a continuous virtual environment. Given an ASCII trace file and 3D CAD models of various entities in the environment, VITASCOPE recreates the simulation with accurate chronological and spatial accuracy. The VITASCOPE system maintains an independent system clock, allowing the user to start and pause at any time within the simulation, or jump to any point in the execution, in addition to the ability to navigate in virtual space and change the user viewpoint. Together, these capabilities allow VITASCOPE to create valuable visualizations of simulated processes.

Commands to build the trace file are integrated into the STROBOSCOPE simulation file. At any discrete stage, such as the initialization or completing of an event, a command can be printed to the trace file along with the simulation time when it occurred. These commands include CREATE, PLACE, MOVE, or DESTROY, allowing the visual models to perform an action at the exact time as it occurred in the simulation, creating a chronologically accurate sequence of events. These printed commands include references to the CAD models associated with each object, and any actions with a specified duration are executed smoothly and continuously.

FIG. 9: Traffic Lights Change when Queue Reaches MaxQueueLength

5. SIMULATION RESULTS AND INTERPRETATION

Each of the 4 models were simulated under varying traffic flow volumes, seeking to quantify which method of intersection control results in minimal average waiting time and traffic backup. STROBOSCOPE maintains system variables for each Queue. The variables of interest are *AveWait* (Avg. time spent in Queue (secs)), *AveCount* (Avg. number of vehicles in Queue), and *MaxCount* (Max. number of vehicles in Queue). This data reflects an equal traffic distribution in each approach, and an equal distribution of departure in each direction.

For the Stop Sign simulation, all measurements were taken from the *Intersection* Queue, since all vehicles passed through this Queue, ordered chronologically by the time of arrival. These values do not differentiate between the 3 approaching lanes, so that the maximum queue size is the sum of waiting vehicles from all 3 directions. (Note that lengthening the time to clear the intersection would reduce discharge time and increase the number of vehicles building up in the *Intersection*.) In any case, Table 1 indicates that low traffic volumes are handled efficiently with a system of Stop Signs, but higher volumes eventually propagate into large delays.

Table 1: Stop Sign Simulation Results

Traffic Volume (vehicles/hour)	Intersection.AveWait (seconds)	Intersection.AveCount (vehicles)	Intersection.MaxCount (vehicles)
2400	0.12	0.12	2
3200	0.16	0.21	3
4000	0.22	0.34	4
4800	0.29	0.51	4
5600	0.52	0.97	7
5800	0.78	1.55	9
5900	1.34	2.68	16
6000	66.51	86.94	284
6100	321.14	656.99	1318

The stop light simulations allow data collection from each individual direction of approach. Tables 2, 3, and 4 provide the *AveWait* and *AveCount* for each of the 4 traffic lanes, in the corresponding **Light* Queue where vehicles are held until their lane has right-of-way through the intersection. The pre-timed simulation with a protected left-turn is initially examined (Table 2). With this model, the Westbound left-turn lane has a longer average waiting time, since it remains red while the two longer light cycles change, but also has a generally lower number of vehicles because Westbound traffic is divided into two lanes. This protected left turn method shows a general increase in queue length and average waiting time across all lanes as the total traffic volume increases. As total traffic increases, the delay for vehicles making left turns grows consistently and remains relatively stable. However, at high volumes of traffic, other lanes are unable to discharge completely in one cycle, leading to significant traffic backup.

The pre-timed simulation with permitted left turn, shown on Table 3, eliminates this backup in the other lanes. By removing the protected left turn, the resulting red time for the other lanes decreases, so the Northbound and Eastbound traffic experience shorter queue length and wait times. Vehicles turning left also experience shorter average waiting time, even though the average vehicle count remains the same. However, for higher volumes of traffic, the large delays are seen in the left turn lane. The visualization confirms that the high volume of Eastbound traffic prevent left turning vehicles from entering the intersection, causing the large average queue size for *WBSLight*.

Table 2: Pre-Timed Simulation with Protected Left Turn Results

Volume	NBLight	NBLight	EBLight	EBLight	WBWLight	WBWLight	WBSLight	WBSLight
(veh/hr)	AveWait	AveCount	AveWait	AveCount	AveWait	AveCount	AveWait	AveCount
	(secs)	(veh)	(secs)	(veh)	(secs)	(veh)	(secs)	(veh)
1600	7.31	1.89	7.18	1.73	6.72	0.8	10.75	1.34
2000	7.56	2.35	7.39	2.14	6.78	0.98	10.84	1.68
2400	7.7	2.79	7.56	2.53	6.86	1.14	10.99	1.97
3200	8.23	3.77	7.93	3.27	7.02	1.43	11.23	2.55
4000	8.7	4.74	8.24	3.99	7.15	1.69	11.49	3.11
4800	9.36	5.82	8.62	4.66	7.28	1.93	11.91	3.63
5000	9.69	6.2	8.69	4.83	7.28	1.98	12.06	3.79
5200	11.74	7.34	8.78	4.95	7.29	2.03	12.38	4.02
5400	85.43	83.38	8.89	5.12	7.34	2.08	13.77	4.53

Table 3: Pre-Timed Simulation with Permitted Left Turn Results

Volume	NBLight	NBLight	EBLight	EBLight	WBWLight	WBWLight	WBSLight	WBSLight
(veh/hr)	AveWait	AveCount	AveWait	AveCount	AveWait	AveCount	AveWait	AveCount
	(secs)	(veh)	(secs)	(veh)	(secs)	(veh)	(secs)	(veh)
1600	4.06	1.84	4.07	1.73	3.77	0.81	5.06	1.37
2000	4.28	2.32	4.19	2.13	3.88	0.96	5.41	1.68
2400	4.39	2.79	4.32	2.54	3.86	1.14	5.73	1.97
3200	4.6	3.77	4.52	3.27	3.95	1.43	6.6	2.57
4000	4.9	4.71	4.75	3.97	4.02	1.68	7.52	3.09
4800	5.17	5.83	5.05	4.64	4.08	1.92	9.3	3.64
5000	5.23	6.36	5.32	4.79	4.11	1.97	68.53	3.77
5200	5.32	7.35	5.39	4.97	4.13	2.04	1416.84	4.02
5400	5.38	59.2	5.44	5.12	4.12	2.09	2839.67	4.38

Finally, the simulation was studied using the actuated stop light model, where right-of-way changes when the traffic queue reaches a certain number. This causes the average waiting time to decrease as traffic volume increases, since cars accumulate more quickly, triggering a green light for that lane. The simulation results indicate that the actuated model becomes more effective for higher volumes of traffic.

Table 4: Actuated Simulation Results

Volume (veh/hr)	NBLight AveWait (secs)	NBLight AveCount (veh)	EBLight AveWait (secs)	EBLight AveCount (veh)	WBWLight AveWait (secs)	WBWLight AveCount (veh)	WBSLight AveWait (secs)	WBSLight AveCount (veh)
1600	8.94	2.73	6.61	1.6	6.14	0.62	7.83	0.84
2000	7.94	2.92	5.83	1.69	5.18	0.63	7.37	0.96
2400	7.28	3.1	5.39	1.8	4.71	0.66	7.08	1.06
3200	6.51	3.48	4.78	1.97	4.12	0.72	6.68	1.29
4000	6.13	3.87	4.45	2.13	3.74	0.78	6.75	1.56
4800	5.95	4.26	4.29	2.32	3.51	0.82	7.01	1.95
5000	5.94	4.39	4.27	2.36	3.43	0.82	7.2	1.96
5200	5.94	4.5	4.25	2.41	3.36	0.83	7.38	2.06
5400	6.66	4.61	4.21	2.51	3.71	0.87	6.13	2.16

Based on the overall results from these simulations, the traffic volume provides a good indication of the best method of intersection control. A system of actuated lights works well for the higher volumes of traffic. For intermediate levels of traffic, a pre-timed system is efficient, but further analysis may be done on the volume of Westbound traffic making a left turn. This allows the user to decide whether the delays caused by a protected left turn signal on other lanes of traffic are acceptable, or if it is more efficient for a permitted left turn to shift those delays to the left turn lane. Finally, for lower levels of traffic, a stop sign may prove to be the more efficient and easily implemented solution to control the flow of traffic through the intersection. In any case, the ability of the user to define the simulation parameters allows for in-depth analysis of the simulated environment which can be made to reflect any real-world intersection.

6. CONCLUSIONS

This paper provided an overview of the challenges and advantages of traffic engineering in designing and optimizing urban signalized intersections. It discussed a number of current methods that are used to model and visualize traffic flow and intersection operation, focusing on computer simulation tools designed especially for intersection analysis. The study modelled urban traffic operations using a general-purpose simulation tool typically used in construction, in order to determine the simulation tool's ability to effectively model operations in the traffic flow domain. Previous work on urban traffic simulation was examined, followed by a description of how traffic flow through a three-way signalized intersection was modelled and animated using general-purpose simulation and visualization tools, that have been successfully used before to model and animate construction processes (e.g. Martinez and Ioannou, 1999, Kamat and Martinez, 2003).

Four scenarios based on different traffic flow control conditions were examined and respective STROBOSCOPE models with all required parameters were presented. Then VITASCOPE was used as a 3D animation tool to ensure the credibility and validity of the models, and to communicate the simulated operations. At the end, results were generated from STROBOSCOPE models. The obtained results highlighted the potential that general-purpose tools, even if originally designed for specific purposes, can be effective in studying operations involving the interaction of entities in multiple domains such as construction and traffic flow, thereby enabling their use in studying mixed operations such as the effect of roadway construction (e.g. lane closures, slow construction vehicles, etc.) on existing vehicular traffic.

Since the effectiveness of general-purpose simulation and visualization tools in modelling and animating construction operations has already been demonstrated before (e.g. Kamat and Martinez, 2003), only traffic operations were modelled in this study in order to keep the discussion concise and focused, and to isolate any modelling or visualization limitations that might have arisen due to the used tools' inability to accurately model traffic operations. Future work will include mixed construction-traffic operations to study the impact of road maintenance projects on urban traffic. The STROBOSCOPE simulation system and its documentation are available at http://www.ezstrobe.com, and the VITASCOPE system and its documentation can be found at http://pathfinder.engin.umich.edu. The latter website also includes several videos depicting modelled and animated operations in construction, transportation, and other domains.

7. ACKNOWLEDGMENTS

The presented work has been supported by the United States National Science Foundation (NSF) through grant CMS-0448762. The authors gratefully acknowledge NSF's support. The authors would also like to acknowledge the Undergraduate Research Opportunities Program (UROP) at the University of Michigan, which fosters collaboration between undergraduate students and faculty by presenting research opportunities to gain valuable experience outside the classroom. The authors are also grateful to Professor Amir Behzadan at the City University of New York (formerly Ph.D. Student at the University of Michigan) for helping conduct some of the background research and for reviewing this manuscript. Any opinions, findings, conclusions, and recommendations expressed by the authors in this paper do not necessarily reflect the views of the NSF, UROP, or the individual mentioned here.

8. REFERENCES

Bayarri, S., Fernandez, M., and Perez, M. (1996), "Virtual Reality for Driving Simulation", *Communications of the ACM*, Volume 39, Issue 5, ACM, New York, NY, 72-76.

Ceylan, H., and Bell, M. G. H. (2004), "Traffic Signal Timing Optimization Based on Genetic Algorithm Approach, Including Driver's Routing", *Transportation Research Part B: Methodological*, Volume 38, Issue 4, Elsevier Science, New York, NY, 329-342.

Chang, J. L., and Su, C. C. (1995), "Predicting Intersection Queue with Neural Network Models", *Transportation Research Part C: Emerging Technologies*, Volume 3, Issue 3, Elsevier Science, New York, NY, 175-191.

Kamat, V. R. (2003), "Vitascope: Extensible and scalable 3D visualization of simulated construction operations", PhD Dissertation, Virginia Polytechnic Institute and State University, Blacksburg, VA.

Kamat, V. R., and Martinez, J. C. (2003). "Validating Complex Construction Simulation Models Using 3D Visualization", *Systems Analysis Modelling Simulation*, Vol. 43, No. 4, Taylor & Francis Group, London, United Kingdom, 455-467.

Khoury, H. M., Kamat, V. R., and Ioannou, P. G. (2007). "Evaluation of General-Purpose Construction Simulation and Visualization Tools for Modeling and Animating Airside Airport Operations", Simulation: Transactions of the Society for Modeling and Simulation International, Vol. 83, Issue 9, *The Society for Modeling and Simulation International (SCS)*, San Diego, CA, 663-679.

Martinez, J. C. (1996), "Stroboscope: State and Resource Based Simulation of Construction Operations", PhD Dissertation, University of Michigan, Ann Arbor, MI.

Martinez, J. C., and Ioannou, P. G. (1999). "General Purpose Systems for Effective Construction Simulation", *Journal of Construction Engineering and Management*, ASCE, 125(4), Reston, VA, 265-276.

Mirchandani, P., and Head, L. (2001), "A Real Time Traffic Signal control System: Architecture, Algorithms, and Analysis", *Transportation Research Part C: Emerging Technologies*, Volume 9, Issue 6, Elsevier Science, New York, NY, 415-432.

Naylor, D. W., and Graham, J. R. (1997), "Intersection Design and Decision-Reaction Time for Older Drivers", *Transportation Research Record*, Volume 1573, TRB, Washington, DC, 68-71.

Yang, Q., and Koutsopoulos, H. (1996), "A Microsoft Traffic Simulator for Evaluation of Dynamic Traffic Management Systems", *Transportation Research Part C: Emerging Technologies*, Volume 4, Issue 3, Elsevier Science, New York, NY, 113-129.