**Project One Checkpoint Report**

**CSE 6730 Group 15**

**Zongwan Cao**

**Rahul Nihalani**

**Yaohong Xi**

**Section 1 Describe the problem**

Our object for this project is to develop a simulation model to compare the average travel time for vehicles to traverse a portion of Peachtree Street (from 10th to 14th) when using synchronized compared to unsynchronized traffic signals. Traffic signal synchronization is a method of timing groups of traffic signals along an arterial to provide for the smooth movement of traffic with minimal stops. By calculating the arrival time for a group of vehicles at specific speed, we can time the traffic signals to turn green just as those vehicles arrive at each intersection.

A more realistic application for this project is we can optimize the travel process and come out with the most effective traffic light pattern to reduce the traffic congestion.

The NGSIM Dataset provides Peachtree Street Trajectory and Signal Time data. We can use it to get the geometric information about the Peachtree. Figure 1 shows the geometric structure of the study area. In this area, there are five intersections that divide the street into four sections.



Figure 1 Study Area Schematic

Source: NGSIM Peachtree Street Dataset

**Section 2 Conceptual model**

This traffic modeling system can be abstracted as queueing network conceptual model. A lane loading cars are regarded as a queue. Each car is an element of a queue. The time cars need to cross an intersection depends on the traffic signal status and cars’ direction. When a car crosses an intersection, it will enter a new queue and leave the old one.

**1.Input and Output**

The first thing we need to figure out is the input and output. Based on our objects,

we need input attributes for each car including carID, starting point, terminal point,

direction and starting time. After running the model, ending time will be obtained from the model. Ending time minus starting time is each car's traveling time. All the input data are produced randomly.

**2.Assumptions**

Also, some assumptions are made during the establishment of conceptual model to neglect some details

(1)The car can start and stop immediately

(2) For all cars, the time for crossing each intersection is fixed

(3) Each lane has unlimited capacity to hold vehicles

(4) There are two lanes on each side

(5) There is no stop sign in each direction

(6) The traffic light has three states, namely red (all cars need to stop), green

(cars can only go straight and turn right) and left (cars can only turn left)

(7) Time for cars driving on the lanes is ignored, which means we just care about time consumed on the intersections

**3.Entities and Events**

Then we move to the content of our conceptual model.

For entities, we consider cars, lanes, traffic lights and junctions.

After analyzing this modeling system, we decide four events

(1) Enterlane, occurring when a car cross an intersection and enter a new lane

(2) Exitlane, standing for a car cross an intersection and leave the old lane

(3) Checksignal, in order to get the status of individual traffic light

(4) Changesignal ,to change the state of individual signal

**Section 3 Literature survey**

Traffic simulation is modeling of transportation system that can help in plan design and operate transportation. The traffic simulation models can be classified according to discrete and continuous time, state and space [1]. The queuing model has long been used to study discrete event simulation [2~5]. Priority queue is a queue data structure for each element has an associated priority. In a priority queue, an element with high priority can be popped before an element with low priority. Queuing model had good performance in simple traffic micro-simulation [6]. It is possible to use such a dynamic, event-based approach to simulate transportation with affordable data and hardware. For example, Vandaele and his team developed some queuing models based on traffic counts and modeled the behavior of traffic flows as a function of some of relevant factors [7]. Furthermore, for the computation of the large-scale it can also be speed up by parallel implementation [8]. In general the queue model is simpler and easier to improve than the flow models of DYNASMART(Dynamic Traffic Assignment and Simulation for Advanced Network Informatics)[9], DynaMIT[10], and the cell transmission model[11]. The disadvantage of queuing model is that the speed in traffic jam condition cannot be realistically modeled; the advantage is higher speed in computation.

[1] Pursula, Matti. "Simulation of traffic systems-an overview." Journal of Geographic Information and Decision Analysis 3.1 (1999): 1-8.

[2] Henriksen, James O. "An improved events list algorithm." Proceedings of the 9th conference on Winter simulation-Volume 2. Winter Simulation Conference, (1977).

[3] Kingston, Jeffrey H. "Analysis of tree algorithms for the simulation event list."Acta Informatica 22.1 (1985): 15-33.

[4] McCormack, William M., and Robert G. Sargent. "Analysis of future event set algorithms for discrete event simulation." Communications of the ACM 24.12 (1981): 801-812.

[5] Chi, Sung-Do, Ja-Ok Lee, and Young-Kwang Kim. "Discrete event modeling and simulation for traffic flow analysis." Systems, Man and Cybernetics, 1995. Intelligent Systems for the 21st Century., IEEE International Conference on. Vol. 1. IEEE, 1995.

[6] Simon, Patrice M., Jörg Esser, and Kai Nagel. "Simple queueing model applied to the city of Portland." International Journal of Modern Physics C 10.05 (1999): 941-960.

[7] Vandaele, Nico, Tom Van Woensel, and Aviel Verbruggen. "A queueing based traffic flow model." Transportation Research Part D: Transport and Environment5.2 (2000): 121-135.

[8] Cetin, Nurhan, Adrian Burri, and Kai Nagel. "A large-scale agent-based traffic microsimulation based on queue model." IN PROCEEDINGS OF SWISS TRANSPORT RESEARCH CONFERENCE (STRC), MONTE VERITA, CH. 2003.

[9] Mahmassani, H. S., T. Hu, and R. Jayakrishnan. "Dynamic traffic assignment and simulation for advanced network informatics (DYNASMART)." Urban traffic networks: Dynamic flow modeling and control. Springer, Berlin/New York(1995).

[10] Ben-Akiva, Moshe, et al. "DynaMIT: a simulation-based system for traffic prediction." DACCORS Short Term Forecasting Workshop, The Netherlands. 1998.

[11] Daganzo, Carlos F. "The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory." Transportation Research Part B: Methodological 28.4 (1994): 269-287.

**Section 4 Simulation Model**

The diagram shown below is a simplified version of peachtree street, which we simulate. The lane numbers and signal numbers are marked. Our model consists of four events, which are briefly described as follows:

1) EnterLane: EnterLane is called whenever a car tries to enter a lane. If the car is the first car in the lane (i.e. the lane was empty before that), it schedules SignalCheck for the front signals, which then remains active till the queue again becomes empty.

EnterLane (Car c, Time t, Lane l) {

If l was empty before c entered {

Schedule event SignalCheck for front signal at time t

}

Push car c in Lane l

}

2) SignalCheck: It is fired by the first car in any lane, and then remains active for that lane till it becomes empty again. It's job is to see if the front car can pass to it's next lane. If yes, it schedules the entry and exit for that car by giving a delay t\_d, which is the time a car takes to pass any intersection. If not, it schedules another SignalCheck after some wait.

SignalCheck (Signal s, Time t, Lane l) {

If l is empty {

exit()

}

If the front car in l can pass {

Schedule entry for that car at time t + t\_d

Schedule exit for that car at time t.

Schedule SignalCheck at time t+t\_d

}

else {

Find the time for next green light.

Schedule SignalCheck at that time

}

}

3) ExitLane: Very similar to entry lane, it is called when a car wants to leave a lane

4) SignalChange: It is scheduled at times when the signals at an intersection need to be changed. It is called periodically.

The implementation for the first checkpoint contains the entry, exit and traversal of cars through the road system. The changing of signals however is not implemented yet. The system for now thus behaves as if all the lights are green. Implementations of various signalling schemes is kept as part of final submission.

For running the code, please refer to the readme file.

