A New Agent-Based Modeling Approach to Simulate Intersection Traffic

Ali Bazghandi, Ghamarnaz Tadayon Tabrizy, and Mehdi Bazghandi

***Abstract*—The objective of this paper is to discuss a simulation model u*sing a*gents for the analysis of traffic intersection performance. Once the frequency and speed of cars is selected, the user should run the simulation and adjust the timing of the traffic light so as to minimize the amount of waiting time of cars traveling through the intersection. The primary output from the simulation model is wait time for vehicles to get through intersection. Statistical measures calculated include the mean, median, and 95th-percentileof wait times, as well as the cumulative distribution function of wait times.**

***Index Terms*—Modeling, agents, traffi, speed, watting time, netlogo.**

I. Introduction

The purpose of this paper is to provide a tool for the analysis of traffic intersection performance. In this model the turtles are cars traveling through an intersection. The user has the ability to control the frequency of cars coming from each direction; Factors considered in the study include intersection size and shape, as well as the traffic signal timing algorithm. With a discrete event simulation, the effects of varying any of these factors can be Easily monitored with performance measures such as the average throughput time or the average delay time at the intersection. The scope of this project is limited to four way signal controlled intersections.. Intersections studied are assumed to be isolated from any neighboring intersections. Consistent with this assumption, a car that reaches the front of its lane of traffic and has a green signal is assumed to be able to leave the intersection with no problem. Exit traffic from the intersection is not modeled. At present the model assumes that there is at most one lane from which cars can turn left and one lane from which cars can turn right for each direction of traffic. Lanes from which cars can go straight are assumed to be infinite in length. Lanes from which cars can turn right are also assumed to be infinite in length. That is, the number of cars that can fit in straight or right turn lanes are not limited by their size.

II. Publicly Funded Simulation Models

A large amount of work has been done in studying and trying to improve trafficintersections. For a good overview of intersection history, terminology, and current techonology, see Black and Wanat [1]. A discussion of many recent trafficanalysis tools can be found in Weiss [2].Originally, much trafficintersection research focused on analytically describing vehicles' delay in terms of the intersection characteristics. One of the most prominent early works is Webster [3]. For more recent analytical delay models and queueing theory approaches see Hurdle [4] and Hagen and Courage [5]. A discussion of more complex intersection models can be found in Meneguzzer [6]. For a detailed study of intersection characteristics and performance, the exibility of computer simulation is preferable to analytical approaches. Many privately and publicly funded simulation models exist for the purpose of studying trafficnetworks. A few examples are UTCS, TRANSYT, SCOOT, DITCS, RT-TRACS, and RHODES [7]. Most of these programs are intended as a tool to aid in setting traffic signal timing algorithms for a network of intersections. One program developed exclusively for the purpose of evaluat- ing traffic network management systems is MITSIM, described in Yang and Koutsopoulos [8].A large amount of work has been done in studying and trying to improve traffic intersections. Wherever Times is specified, originally, much traffic intersection research focused on analytically describing vehicles' delay in terms of the intersection characteristics. For a detailed study of intersection characteristics and performance, the excitability of computer simulation is preferable to analytical approaches. Many privately and publicly funded simulation models exist for the purpose of studying traffic networks. A few examples are UTCS, STARLOGO, NETLOGO SCOOT, DITCS, RT-TRACS.

III. Discussion of “Good" Intersection

As detailed in the previous section, the performance measures from the model include the mean, median, 95th-percentile, and cumulative distribution function of the wait times from cars in the simulation. It is appropriate at this point to discuss what can be implied from each of those statistics, the pros and cons of each as a measure of intersection performance, and the benefit of considering multiple of these measures. The following sections explain these points.

The sample median is another common measure of central tendancy. It has the nice feature of being inherently linked to the distribution of a popula- tion as 50% of the population is below the median and 50% is above. Similar to the sample mean, the median provides little information about the tails of a distribution. When considering a real world system, as in the case of a discrete-event simulation, the tails of a distribution are often the most impor- tant part of the model [12]. To put this in context of a trafficintersection, a driver that waits for the mean or median amount of time is pretty indi®erent about their experience at that intersection. On the other hand, a driver that experiences a wait time from one of the tails of a distribution is going to be much more excited (for better or for worse) than the previous driver that fell in

Manuscript received May 23, 2012; revised June 24, 2012.
M. Bazghandi is with school Mashhad Branch-Islanmic Azad University, Sabzevar, Iran. (e-mail: Mwhdi_bazghandi@yahoo.com).

the center of the distribution. This is why it is desireable to acquire knowledge about the tail behavior of a system, in addition to measures of central tendency.

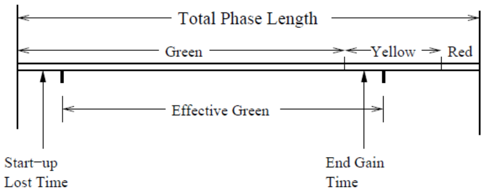

Fig. 1. Signal settings vs. effective phases

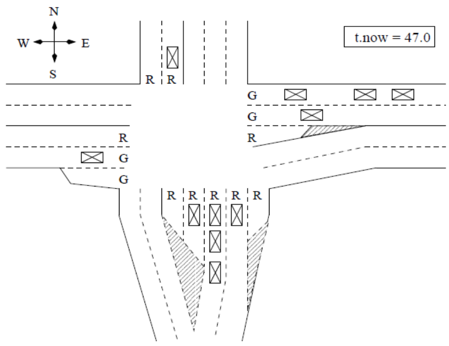

Fig. 2. Sample intersection - snapshot of dynamic information

IV. The Basic Model

In my discussion of the program implementation, the distinction is made between static and dynamic variables. Static variables are those that define the size, shape, and other qualities of the intersection that do not change over time. Other static variables describe the input model for a particular intersection. Characteristics such as the stochastic model for arrivals and turn probabilities are input once at the beginning of the study and also do not change with time. Dynamic variables include the data structures necessary to completely describe the state of the intersection at any point in time.

A. Static Variables

Static variables are those that define the size, shape and other qualities of the intersection that do not change over time. Other static variables describe the input model for a particular intersection. Characteristics such as the stochastic model for arrivals and turn probabilities are input once at the beginning of the study and also do not change with time.

The model has these components:

1) Number of lanes. The integer-valued vector lane gives the number of approaching lanes at the intersection for each of the four directions.
2) Size of lanes. The floating point-valued matrix geometry gives the size of each lane in feet measured from the corner of the intersection backwards. This matrix has dimensions 4 * max lanes and includes information about each lane in each direction of traffic. Clearly state the units for each quantity that you use in an equation.
3) Turning information, to make the distinction between lanes that are for turn only, straight only, or straight/turn, the integer-valued matrix Direction contains the appropriate information for each lane. Direction also has dimensions 4*max lanes.
4) Placement of traffic sensors, to capture the quality of data sensors used for actuated control of an intersection there is a floating point-valued matrix Sensor that store the information for each lane in the intersection.

B. Static Variables - Stochastic Input Model

Arrival Process ,To model the arrival process, 100 interarrival times were collected from each direction of approaching traffic. The distinction between arrivals to a specific lane, or arrivals with a speciffic turning intention, is not made at this level.The data of interarrival times were fitted to statistical distributions using the data fitting package in Arena [9]. The output from the Arena package consists of the statistical distribution which provided the best fit to the data,as well as the maximum likelihood estimators for the parameters of that distribution. Note that the parameterization of the log normal distribution that is used here is lognormal (μ_1, σ_1) in (1):

$$\frac{1}{\sigma t\sqrt{2\pi}} e^{-\frac{(\log t-\mu)^2}{2e^{\sigma^2}}} \quad (1)$$

For t > 0. The mean is $\mu_1 = e^{\mu+\frac{e^2}{2}}$ and the variance is:

$$\sigma^2{}_1 = e^{\mu+\sigma^2}(e^{\sigma^2}-1).$$

1) Vehicle size. Distribution for each direction of approaching traffic, there is a specific distribution of the types of vehicles that arrive.
2) Turn probabilities. For any direction of approaching traffic, a certain proportion of cars will turn left, go straight, or turn right. This information is stored in the 4×2 floating point-valued matrix TurnProb.

Light Sequence. Algorithm for any intersection that is modeled, it is necessary to describe the algorithm by which the traffic signal is updated. Before discussing the computational methods for implementing this, consider the behavior of traffic during the respective signal settings of green, yellow, and red. When a signal turns green, there is a delay of a couple seconds before the first car in line accelerates and leaves the intersection. In the civil engineering community, this is referred to as the start-up lost time. When a signal changes from green to yellow, there are a couple seconds during which cars continue to pass through the intersection. This is referred to as the end-gain time. During the time between the start-up lost time and the end-gain time, the signal is effectively green and cars leave the intersection at a steady rate. It is shown in Fig.1.

C. Dynamic Variables - System State

Traffic signal is the Boolean matrix light contains information on which lanes of traffic have red or green lights. A value of 0 indicates a red signal and a value of 1indicates a green signal. This 4 * max lanes integer-valued matrix is organized according to the approaching direction of traffic and the lanes within that direction in the same manner as the static variables. The elements of the Light matrix corresponding to lanes that do not exist will always have a value of 0. Then for the snapshot of the intersection shown in Fig.2.

1) Updating the traffic signal if a lane of traffic had a green signal in a previous phase and a red signal in an upcoming phase, the function Change2Red will turn it red. That is, it will set Light[*i*][*j*] = 0 for the appropriate direction i and lane j.
2) Cars. The state of any car in the simulation which is at the intersection is defined by the following characteristics. Definition arrival time, turning intention, current distance, vehicle lengths and queue cars.

VI. Output of Model

The intersection is modeled using a next-event simulation . That is, the simulation only considers a set of finite events that can change the state of the system.

While (t:now < STOP) { e ← { i^* | .*t*≤ event[*i*].t ;*i*= 0,1,2,..,6}
t.now ← event[e].t [Update simulation clock]
If (e < 4) {[Process an Arrival from direction e]
event[e].t ← t.now + GetArrival(*e*) [Schedule next arrival]
NewCar.arr ← t.now [Save car's arrival time]
NewCar.turn ← PickDirection(*e*) [Assign turn direction]
NewCar.len ← PickSize(e) [Assign length of vehicle]
newLane ← PickLane(Queue,NewCar) [Place in appropriate lane]
if (Light[*e*][newLane] = 1) [If green signal]
Stats[*e*][newLane].count[0] + + [Increment frequency count]
Else [If red signal]
else if (e = 4) { [Process a Departure]
PassThru() [See documentation]
For all (*i*; *j*) ∍ Light[*i*][*j*] = 1 [Check green lanes for next departure]
(approach, lane) ← (i^* , j^*) | NextDeparture[i^*][j^*]≤ NextDeparture[i^*][j^*] }
nextTime ←NextDeparture[approach][lane]
If (!Queue[approach][lane].isEmpty()) [If more waiting]
event[4].t ← nextTime [Schedule next departure]
else
event[4].x ←0 [Turn departure off]
else if (e = 5) { [Change signal to red]
For all I, j
If Light[i][j].OldPhase=1)&(light[i][j].Newphase=0)
Light[i][j] ← 0 [New red]
event[5].x ← 0 [Turn off Change to red]
event[6]:t ← t.now + blockTime [Schedule Change to green]
event[6].x ← 1 [Turn on Change to green]
}
else { [Change signal to green]
For all Ii, j
If (Light[i][j].OldPhase = 0) & (Light[i][j]. NewPhase = 1)
Light[i][j] ← 1 [New green]
event[6].x ← 0 [Turn off Change to green]
event[5].t ← t.now + phaseLength [Schedule Change to red]

The purpose of a simulation is to gain insight about the system being modeled. To that end, there are a number of performance measures reported by the program at the end of a simulation experiment. Computing these statistics is accomplished by keeping wait time data for every vehicle in the simulation. The histogram-type integer-valued vectorcount[maxTime] is used to store this information1. The integer max Time is the largest wait time recorded by the simulation and is set by the user depending on the experiment at hand. By default, any wait times that may be greater than max Time are considered equivalent to max Time for the purpose of collecting statistics. For any *i* = 0; 1; : : : ; max Time - 1, count[*i*] is the number of vehicles that had wait time, wt, such that *i* ·wt < *i*+1. In other words, each time a vehicle departs the intersection, the wait time is recorded by the statement count[min([wt], maxTime-1)]++. This discussion refers to the case of collecting statistics on every car in the simulation. The sameapproach is just as easily applied to compute statistics for more speciffic studies such as comparing the performance of the left, middle, and right lane of an approach direction, or comparing the performance of left turns versus all other movements. The specific breakdown of the analysis is dictated by-the intersection being modeled, and the questions that hoped to be answered by the study.Since it is histogram-type count data that is being recorded, the sam ple mean can be calculated according to the method of histogram means described in Park and Leemis [10]. That is:

$$\nabla = \frac{1}{n} \sum_{i=0}^{\max Time-1} i.count[i] \quad (2)$$

where *n* is the total number of observations recording in count.

The empirical cumulative distribution function of wait times can be calculated from the histogram by starting at zero and adding the contribution of 1/*n* count[*i*] for each i such that count[*i*] > 0. Let F(i) represent the empirical CDF of the wait times at any (integer) time value *i*. Then the median, t0:5, is the minimum i such that F(*i*) > 0.5. Likewise, the 95-th percentile is the minimum i such that F(*i*) > 0.95.The cumulative distribution function of wait times gives the whole picture of the proportion of cars that short wait times and long wait times. It is still bene¯cial to have the numerical measures of the mean, median, and 95th-percentile for ease of comparison between systems. The CDF of wait times will be used in some cases for visualizing the results of a simulation. Consider the CDFs of wait times for the two hypothetical Systems A andB shown in Fig.3. The median and 95th-percentile wait times for each system are sketched on the graph as well. The two systems have identical 95th-percentile wait times, but vastly different distributions aside from that point. If you were a driver with a choice between the two Systems A and B, which one would you choose? System A has a much better median wait time than system B. However, System A also has the chance of producing a much worse wait time than System B ever would. Considering that drivers who experience about the median wait time are generally apathetic about the situation, where drivers who wait the maximum amount of time are often extremely angered, it may be advantageous to play it safe and select System B.

The definition of "good" intersection performance with respect to the statistics considered in this study is not entirely clear. Ideally, each of the mean, median, and 95th-percentile of wait times should be as small as possible to get the most cars through the intersection with as little delay as possible.

Clearly, an intersection arrangement that has each of those three statistics smaller than an alternative intersection arrangement should be the preferred choice. What is unclear is which intersection is preferable if the first has superior mean and median wait times, and the second intersection has the superior 95th-percentile wait time. Or which is preferable if the first intersection has a superior median and 95th-percentile of wait times, while the second intersection has a superior mean wait time. The permutation of these three statistics among the available intersection arrangements make this a complicated question. The author suggests that a meaningful description of the trade-o®s between the mean, median, and 95th-percentile wait times may be an open research question. A detailed study of these relationships would depend on the objectives of the researcher. For example,maximizing intersection safety and maximizing driver satisfaction could lead to two different answers to the question of what is the best intersection arrangement. A philosophical discussion of these points is outside the scope of this report. The author merely intends to point out that some ambiguity does exist regarding the definition of a “good" intersection.

We that a meaningful description of the trade-offs between the mean, median, and 95th-percentile wait times may be an open research question.

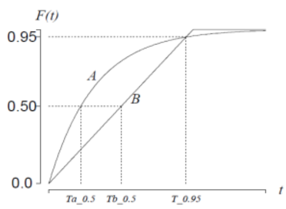

Fig. 3. Hypothetical wait time CDFs

This report follows the design and implementation of a traffic simulator written in NetLogo, an agent-based modelling environment. The system is developed with a view to analysing the likely effect of congestion-reducing schemes, in particular high occupancy vehicle lanes. The simulator is calibrated and validated using data from traffic studies, and then a section of road in South Gloucestershire is modelled to investigate the introduction of high occupancy vehicle lanes. The results suggest that the high occupancy vehicle lane fulfils its objectives by significantly reducing travel times for car sharers without adversely affecting other drivers.

Traffic simulations can be broadly classified by the type of intersection and features they 2 can simulate. Simulators supporting an intersection Try different numbers of eastbound cars while keeping all other slider values the same.it shown in Fig.4.

Try different numbers of northbound cars while keeping all other slider values the same.

LIMIT while keeping all other slider values the same. Try different values of MAX-ACCEL while keeping all other slider values the same.Try different values of GREEN-LENGTH and YELLOW-LENGTH while keeping all other slider values the same. Performamce evaluation shown Fig.5.

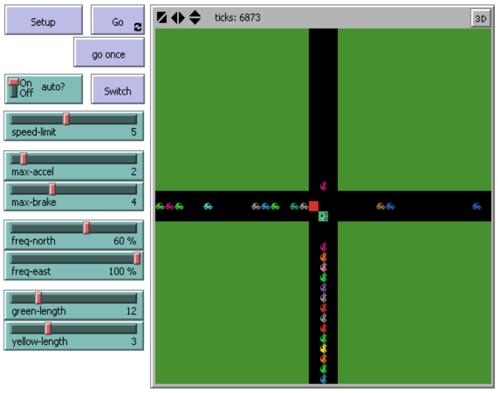

Fig. 4. A snapshot of the GUI interface of theAlgorithm simulation platform Using above

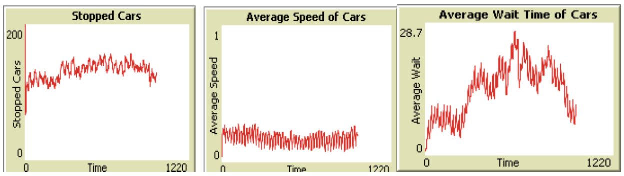

Fig.5.Performamce evaloation

References

[1] S. A. Asante, S. A. Ardekani, and J. C. Williams, “A Simulation Study of the Operational Performance of Left-Turn Phasing and Indication Sequences," *Transportation Science*, vol. 30, no. 2, pp. 112-119, 1996.

[2] J. Black and J. Wanat, “Traffic Signals and Con-trol, Ramp Metering,” and Lane Control Systems.

[3] B. Bynum, “Computer Organization and Programming Languages,” College of William and Mary, Computer Science Department, 2001.

[4] F. M. Carrano, P. Helman, and R. Veroff, “Data Abstraction and Problem Solving with C++,” Addison-Wesley, Inc., Reading, MA, 1998.

[5] L. T. Hagen and K. G. Courage, “Comparison of Macroscopic Models for Signalized Intersection Analysis," *Transportation Research Record 1225, TRB, National Research Council*, Washington, D.C., pp. 33-44, 1989.

[6] V. F. Hurdle, “Signalized Intersection Delay Models - A Primer for the Uninitiated," *Transportation Research Record 971*, TRB, National Re-search Council, Washington, D.C., pp. 96-105, 1984.

[7] A. M. Law and W. D. Kelton, “Simulation Modeling and Analysis,” McGraw-Hill, Inc., New York, 2000.

[8] L. Leemis, “Reliability: Probabilistic Models and Statistical Methods,” Prentice-Hall, Englewood Cliffs, N.J., 1995.

[9] A. D. May, Jr., “Gap Acceptance Studies," Highway Research Board Record 72, HRB, Washington, D.C., 1965.

[10] C. Meneguzzer, “Review of Models Combining Traffc Assignment and Signal Control," *Journal of Transportation Engineering*, vol. 123, pp. 148-155, 1997.

A. Bazghandi was born in Iran in 1976. I\'m masterof science in computer engineering from Iran University of Science and Technology (2001). Now I\'m member of Shahrood University of Technology.
Email: bazghandi@shahroodut.ac.ir

M. Bazghandi was born in Iran in 1986. I\'m student of master of science in computer engineering from (2010).I studing inS laumic Azad University of Mashhad.
Email:Mehdi_Bazghandi@yahoo.com.

G. T. Tabrizy is a professor of science in computer engineering from Iran. Now I\'m memberof Mashhad Branch-Islamic Azad University.
Email: gh_tadayon@yahoo.com.