Computer Simulations of Traffic on Road Intersections

Eden Delight B. Provido, Mark Adrian Ramires, Winfer C. Tabares, and Rafael P. Saldaña

Computational Science and Scientific Computing Group

Mathematics Deparment, Ateneo de Manila University

E-mail for correspondence: rsaldana@ateneo.edu

ABSTRACT

This study developed a model and implemented a simulation of a typical Metro Manila intersection. The simulation is implemented in serial programming using JAVA. Two-dimensional four-segment intersections and deterministic and stochastic cellular automata models with non-periodic boundary conditions are employed. Results show that the intersection models developed satisfy the theoretical assumptions of the classical theory of traffic dynamics.

Keywords

Traffic Dynamics, Cellular Automata, Computer Modeling and Simulation, JAVA

1.INTRODUCTION

Vehicular traffic is a major problem not only in the Philippines but also in many countries around the world. Several studies have shown that mathematical modeling and computer simulation techniques can be used to study the dynamics of vehicular traffic.

Since the year 2000, the Computational Science and Scientific Computing Group of the Mathematics Department of Ateneo de Manila University has been conducting studies on the computational aspects of vehicular traffic dynamics [2, 3, 4, 5, 6].

The present study considers the dynamics of the two-lane road intersection model based on the Cellular Automata traffic model developed by Nagel and Schreckenberg [1].

2. TWO-LANE INTERSECTION MODEL

The two-lane intersection model involves four-segment, two-lane intersections. It simulates the movement of cars in one segment of an intersection without considering the traffic light restraints.

In addition to the two-lane model in freeways, the destination of each car is considered. A tag determines the said destination. There are four possible actions: u-turn, left, right and straight. A car that will take a u-turn or a left turn should take the left lane while a car that will go straight or right should take the right lane. The whole segment is divided into three parts: the regular two-lane region where each car is to behave similar to a car in a two-lane freeway (TL), the lane-changing region (LC) where each car places itself in the lane which is dictated by its destination and the no-change region where each car behaves similar to a car in just one lane (OL). This division is assumed so as to model the behavior of cars along an intersection.

Normally, a car that is far from the intersection is in a free-drive mode. This means that the car moves from one lane to another without considering its destination. This is the reason behind the two-lane region (TL). Once it reaches a certain section of the road, the car will have to place itself in the proper lane. The lane-changing region addresses this case. After transferring to the proper lane, the car should no longer be allowed to change lanes. Otherwise, the car might miss its destination. This is why the last few cells follow one-lane neighborhood rules.

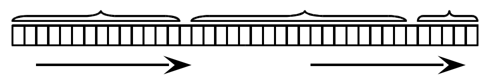

Figure 4.4: Three Divisions in the Two-Lane Intersection Model

The neighborhoods for the two-lane(TL) and one-lane division (OL) follow the same rules as the neighborhoods for the two-lane and one-lane freeways, respectively[6]. The third kind of neighborhood follows different rules. Based on the behavior of vehicles that need to transfer lanes in order to go to its destination, certain rules for lane-changing are formulated. The rules for the forced lane-change region include the following:

- Check if the car is in the proper lane. A car is in the proper lane if:
  - The tag is either straight or right turn and the car is on the rightmost lane of the road.
  - The tag is either u-turn or left turn and the car is on the leftmost lane of the road.
- If the car is in the proper lane, stay on the current lane.
- If the car is not in the proper lane, follow these rules:
  - Check back on the other lane if the car will get in the way of another car.
  - If the car will get in the way of another car, slow down and stay on the current lane.
  - If the car will not get in the way of another car, change lanes and update the speed as needed.

These rules alone, however, may cause deadlocks in the last cell of the neighborhood. This is why additional rules for the last cell are set. In case a car reaches the last cell in the region and the next cell on the other lane is empty, it transfers to the next cell on the other lane at the next time step once certain probability conditions are satisfied. This prevents deadlocks from happening.

Unlike the freeway models, the intersection model is nonperiodic. The movements of the cars at the end of the road do not depend on the cars at the beginning of the road. To make the system nonperiodic, a car-replacement scheme is formulated. This keeps the number of cars in the system constant. Once a car crosses the intersection, a car with a randomly generated destination, speed and location is placed at the beginning of the lattice. In case the section is full, the number of cars that crossed the intersection will be stored. At the next timestep, the corresponding (replacement) cars will be generated.

As in the one-lane and two-lane simulations, stochasticity is also introduced in this model. If the random number generated is less than the noise factor, then the car will stay in its lane instead of changing lanes right away. After the forced lane-changing rules have been implemented, the velocity and movement updates are done on each of the two lanes.

3.Four-segment, Two-lane Intersection Model

In the four-segment, two-lane intersection model, four two-lane intersection segments are put together to form a complete intersection. The intersection follow this diagram:

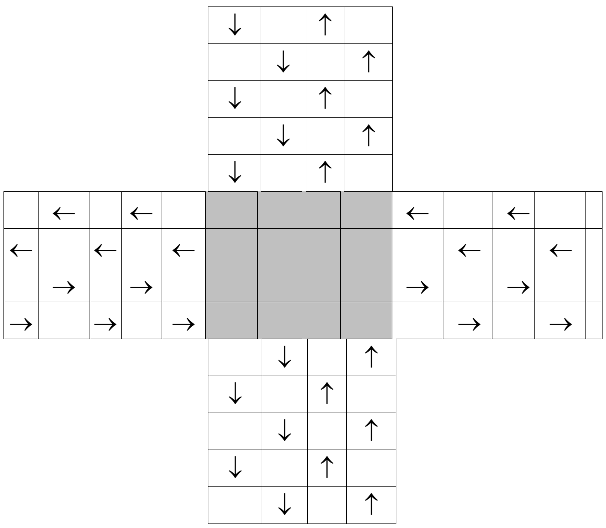

Figure 3.5: Four-Segment, Two-Lane Road Representation

Another important feature of the simulation is the implementation of traffic lights. The model follows a one-segment per traffic light system wherein each segment is given a go-time depending on the user's input. Once a go signal is given, all the cars in that particular segment are allowed to move. Otherwise, the cars on the cells near the intersection are put to a stop.

A limitation of the said simulation, though, is that it does not simulate where each specific car will go after it has crossed the intersection. In essence, it only models the behavior of cars in each segment without including the movement of cars after the intersection. The model is illustrated as follows:

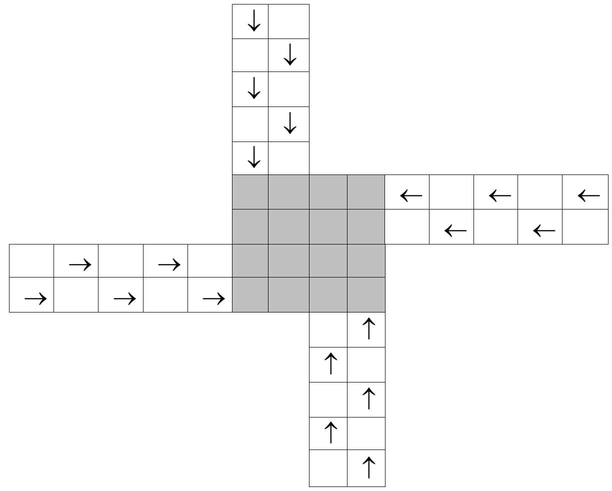

Figure 3.6: Intersection Representation

4. THE JAVA IMPLEMENTATION

The simulation for road-intersections is composed of two-parts: the two-lane intersection model and the four-segment, two-lane intersection model. Using the Java Software Development Kit, a program for the road-intersection is created. As stated earlier, three types of neighborhoods are used. The one-lane and two-lane neighborhoods follow the Nagel-Schreckenberg CA model with minor modifications for the traffic light, while the third type of neighborhood (forced-turn) is set up as a new system.

The program requires the following inputs:

maxSpeed - maximum speed

normalize - number of timesteps before data is gathered

roadLength - length of the road

timeDuration - number of timesteps in the simulation (to be stored)

turnRegionSize - size of the forced lane-changing region

turnPoint - point in the lattice where all the vehicles should be in the desired lane

noise - randomization parameter

pDest[] - probability (for each segment) that a car will go straight, left, right or back

numOfCars[] - number of Cars per segment

goTime[] - duration of the go signal per segment

5. RESULTS AND DISCUSSION

5.1Two-Lane Intersection

For the intersection simulation in the two-lane with car destination case, it is to be shown that the model follows and is consistent with qualitative observations of real traffic. This is equivalent to saying that, for the average speed versus density plot, speed increases as density increases. It also means that for the flux versus density graph, flux initially increases as density decreases but eventually decreases as the density is increased further.

Figure 5.1 and 5.2 illustrate this point. For these graphs, the system is composed of 200 cells. Each car can be allowed to run at a maximum speed of 6 cells/second. There is no noise in the set-up, which is run for 1000 time steps.

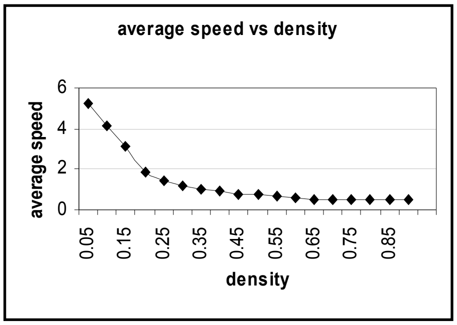

Figure 5.1: Speed vs Density

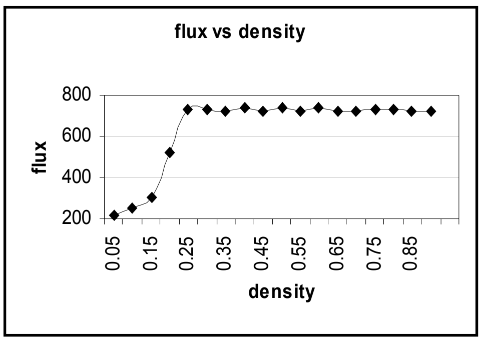

Figure 5.2: Flux vs Density

The Speed vs Density chart (Figure 5.1) shows that the two-lane intersection model follows qualitative observations of real traffic; that is, as the density of the system increases, the average speed of the cars in the system decreases. For the Flux vs Density plot (Figure 5.2), it can be seen that the curve reaches a plateau stage instead of having a decreasing flux. This is due to the fact that the model allows for a non-periodic car replacement scheme. At the point where the curve starts to plateau, the system becomes saturated. This means that no car can enter the system because there are cars blocking the entrance (the leftmost part of the road). It must also be noted that the model allows for a continuous flow of traffic. In contrast to the periodic traffic flow model, the flow of the cars at the rightmost section of the road does not depend on the traffic situation at the beginning of the road. This prevents the flux level from decreasing since the cars at the rightmost section of the road are always moving.

The next few graphs (Figure 5.3-5.6) show the relation of the number of iterations to the average speed in the system. The system uses 100 timesteps, and no noise at varying densities (0.2, 0.4, 0.6, 0.8).

From these plots, one can infer that as the density increases, the average speeds of the systems fluctuate less. This is a logical result. As the number of cars in the road decreases, there will be less room for changing the speed at every time step. It can be observed though that, unlike the Nagel-Schreckenberg model (in the case where there is no noise), the average speed fluctuates. This can be explained by the fact that the different sections of the road require different movements and speed variations.

For the next set of graphs, average speed, flux and density are plotted. This is run on a lattice with a roadlength of 200 in 2000 iterations. The randomization parameter is pegged at 0.1. The maximum speed is varied from one to six. For Figure 5.7, the topmost curve represents the data at a maximum speed of six while the curve at the bottom represents the data at a maximum speed of one.

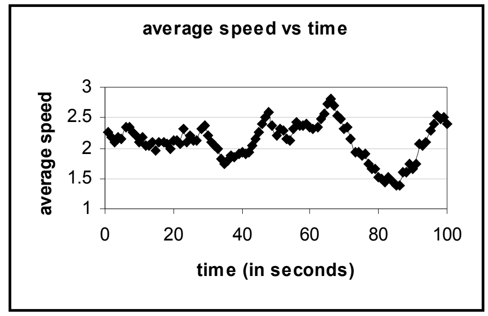

Figure 5.3: Average Speed at density=0.2

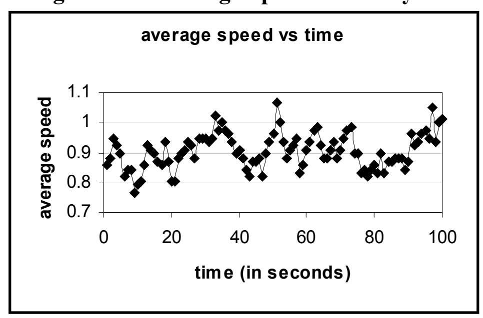

Figure 5.4: Average Speed at density=0.4

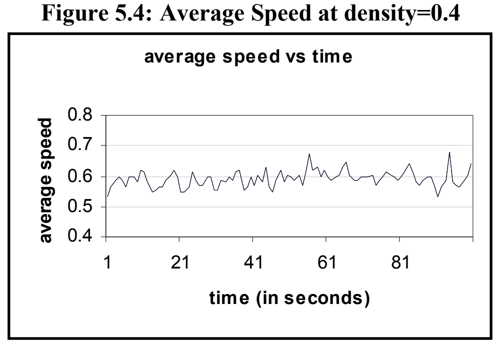

Figure 5.5: Average Speed at density=0.6

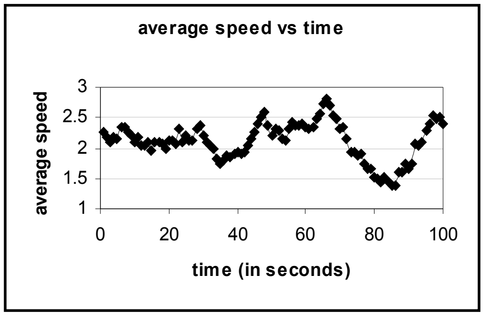

Figure 5.3: Average Speed at density=0.2

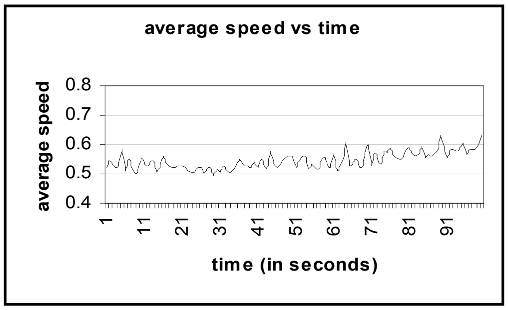

Figure 5.6: Average Speed at density=0.8

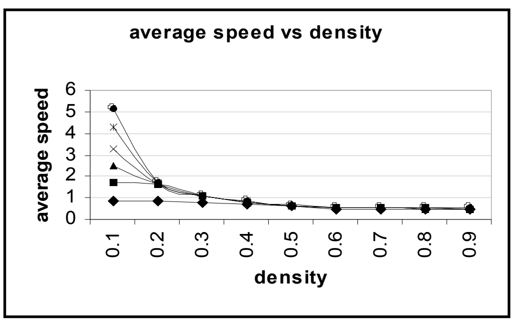

Figure 5.7: Average Speed vs Density

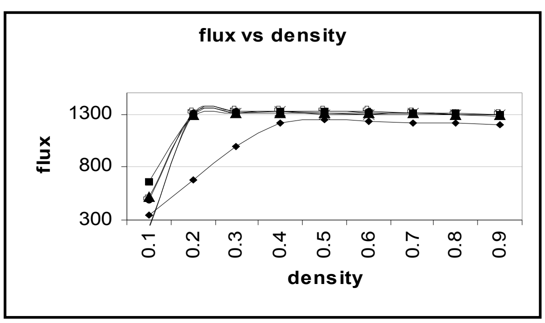

Figure 5.8: Flux vs Density

Again, this follows the qualitative behavior of real traffic [6]. At a density of 0.1, the maximum average speed is attained. This occurs at the curve where the maximum allowable speed is 6. There is continuous decrease as the density increases until a certain density level in which the maximum speed limit is no longer significant. In essence, this shows that when there is a large number of cars on the road, an increase in the maximum speed has little effect on the average speed. All cars are already travelling at low speeds. This also holds true for the flux vs density graph. Given a high density on the road, the flux of the system will no longer be affected by the maximum allowable speed.

The next two graphs show that changing the randomization parameter (noise) of the system affects the speed and flux on the road. This is run on a lattice with a roadlength of 200 in 1000 iterations. The randomization parameter is varied from 0.1-0.5. The maximum speed is set at six. For Figure 5.9, the topmost curve represents the data at a noise of 0.1 while the curve at the bottom represents the data at a noise of 0.5.

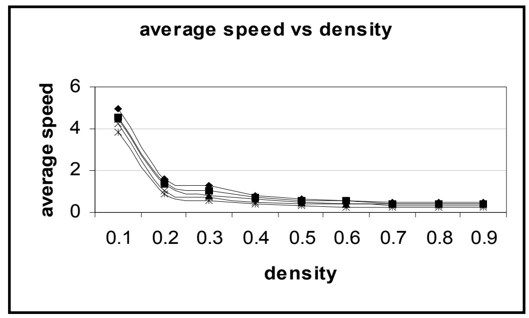

Figure 5.9: Average Speed vs Density

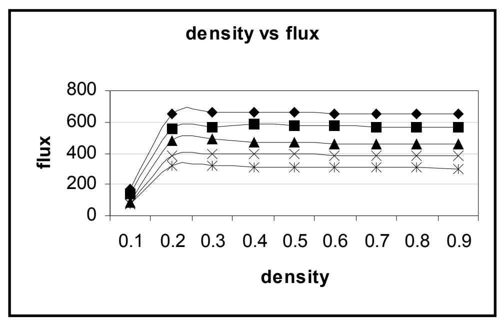

Figure 5.10: Flux vs Density

As stated earlier, these plots show that increasing the inefficiency of the cars in the system will affect the average speed of the system. Figure 5.10 shows that the flux is decreased as the noise level increases. At a noise rate of 0.5, the flux is between 200-400 cells per timestep whereas at a rate of 0.1, the flux is between 600-800.

Figures 5.11-5.14 compare the traffic dynamics at varying densities(0.2, 0.4, 0.6, 0.8). It plots time against space.

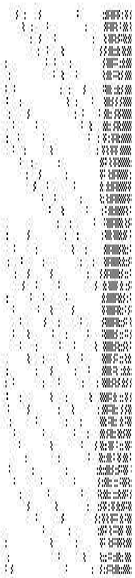

Figure 5.11

Density=0.2

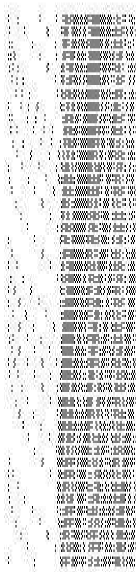

Figure 5.12

Density=0.4

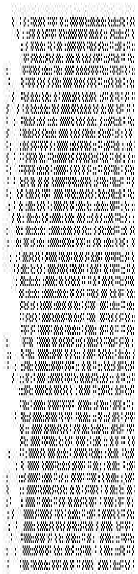

Figure 5.13

Density=0.6

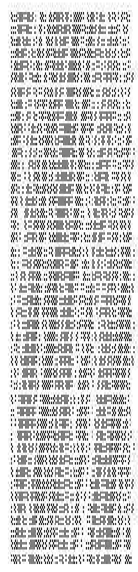

Figure 5.14

Density=0.8

Similar to previous studies, it shows the backward movement of traffic[6]. At a density of 0.2, the cars at the leftmost section of the road move forward. This can be attributed to the density of the system in that section. The cars can move freely since there are only a few cars on the said part of the road. The decrease in the speed occurs at the latter section of the road. This is the section where the cars are forced to change lanes to reach their destination. As the density increases, the traffic congestion at the rightmost section accumulates and moves towards the leftmost part of the road. At a density level of 0.6 or 0.8, one can no longer notice where the congestion is coming from (unlike in the case where density is 0.2 or 0.4).

The graphs in Figures 5.15-5.18 show the movement of traffic when the noise is changed.

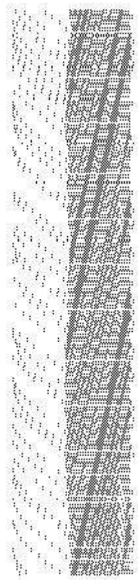

Figure 5.15

Noise=0

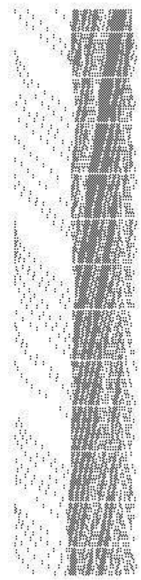

Figure 5.16

Noise=0.2

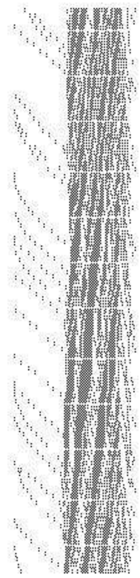

Figure 5.17

Noise=0.4

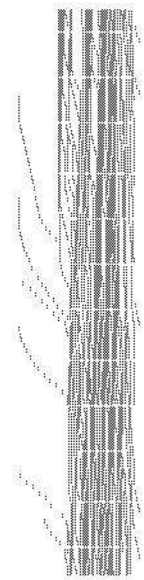

Figure 5.18

Noise=0.8

This follows the observations made in previous studies; that is, as the noise increases, more black areas are clustered together [1,6]. These graphs also show that the number of cars entering the system decreases as the noise is increased. This can be seen through the number of forward-moving lines moving toward the traffic congested regions in the system. As the noise increases, the number of lines also decreases. Another remark is that, as the noise increases, the backward movement of the cars seems more steep. This can be seen by considering the plot of Figure 5.15 and 5.18.

5.2Four-Segment Two-Lane Intersection Model

For the four-segment, two-lane intersection model, two types of relationships will be shown. For the first set-up, the density of each segment is fixed while the duration of the green light is changed. The second set-up sets a fixed traffic light duration while changing the density of each segment, leading to the creation of major roads (segments with significantly greater density than the others) in the system.

Some of the Time vs Space plots the model are shown in Figures 5.19-5.22.

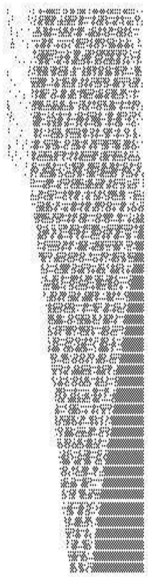

Figure 5.19

Segment 1

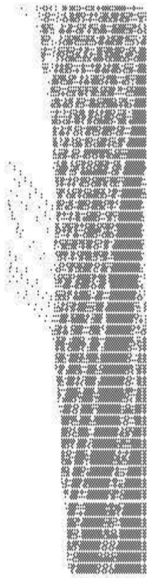

Figure 5.20

Segment 2

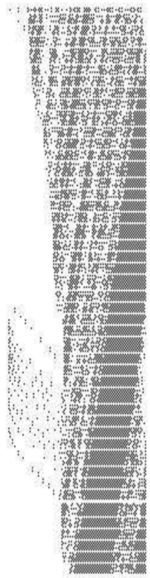

Figure 5.21

Segment 3

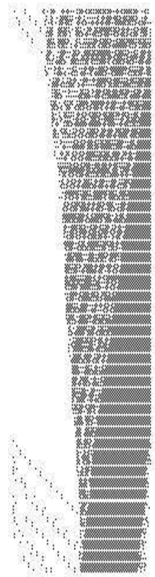

Figure 5.22

Segment 4

In these graphs, the go signal occurs at the region located right before the forward movement section of the cars (found at the leftmost section of the road). This shows that the traffic light alternates between the segments of the road. As stated earlier, the model still simulates the backward movement of traffic as seen in all the segments of the four-segment intersection.

The next few graphs are run on a four-segment, two-lane intersection with 720 iterations. Noise is set at 0.1 and the average density is fixed (0.25, 0.5, 0.75 for each graph). Speed and flux are plotted against traffic light combination. For this study, five different combinations for the traffic lights are used. Each round (each segment being given a go signal) consists of 360 seconds. Since there are 720 iterations, there will be two rounds for the simulation. This is roughly twelve minutes, allowing six minutes for each round. The combinations represent the time allowed for segment 1 and 3 (opposite roads), and for segment 2 and 4, respectively. For example, a 60-120 combination would mean that roads 1 and 3 are given a 60-second go signal whereas segments 2 and 4 are given 120 seconds. One round, then, runs as follows:

Segment #	time elapsed
1	60
2	120
3	60
4	120
Total	**360**

Table 5.1

The traffic light combinations in the system are shown in Table 5.2:

combination	section 1 & 3	section 2 & 4
1	18	162
2	36	144
3	54	126
4	72	108
5	90	90

Table 5.2

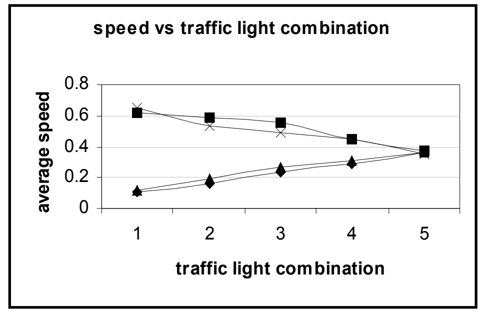

Figure 5.23: Speed vs Light at Density=0.25

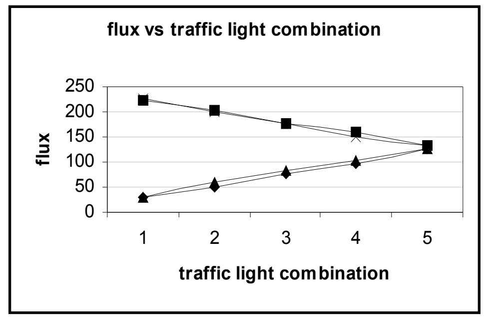

Figure 5.24: Flux vs Light at Density=0.25

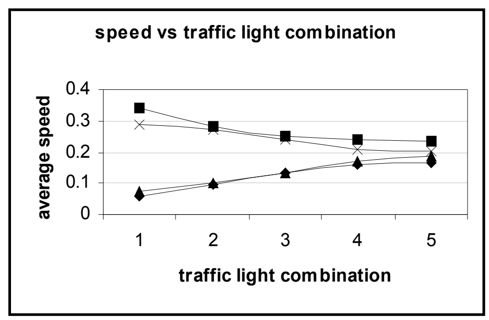

Figure 5.25: Speed vs Light at Density=0.5

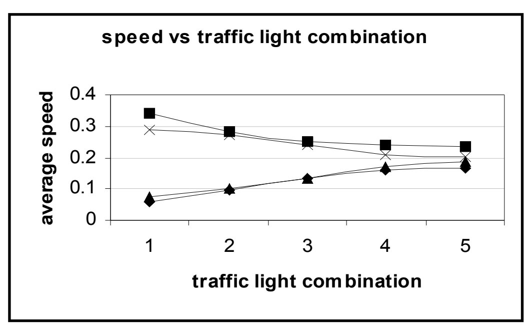

Figure 5.26: Flux vs Light at Density=0.5

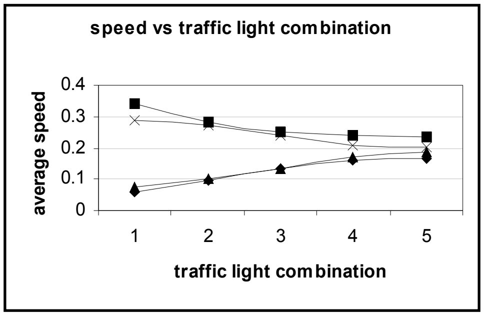

Figure 5.25: Speed vs Light at Density=0.5

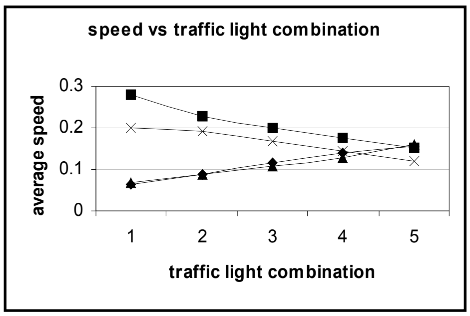

Figure 5.27: Speed vs Light at Density=0.75

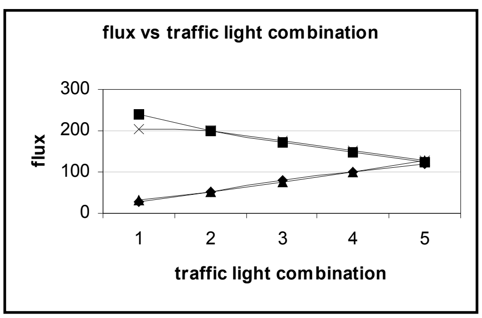

Figure 5.28: Flux vs Light at Density=0.75

For all the graphs, it can be noted that, as the traffic light combination reaches 5 (90-90 combination), the flux and the speed approach a certain range of values. In the case of the topmost curves (segments with longer go signals), as the go signal is decreased, the flux and the average speed are also decreased. This is a logical occurrence since, in real life, a decrease in the go-time in the road would also lead to slower movement of vehicles. In the case of the curves at the lower part of the graphs, as the go signal is increased, the average speed and the flux also increases.

Figures 5.29-5.30 cover the case when the traffic light combination is held constant at a rate of 90 seconds (roughly one and a half minutes) while changing the density combination. The same time frame is used (720 iterations or two rounds). The system is run on a road length of 100 cells at a noise of 0.1. This is similar to the approach earlier. In this case, the combinations are as follows:

combination	section 1 & 3	section 2 & 4
1	20	180
2	40	160
3	60	140
4	80	120
5	100	100

Table 5.3

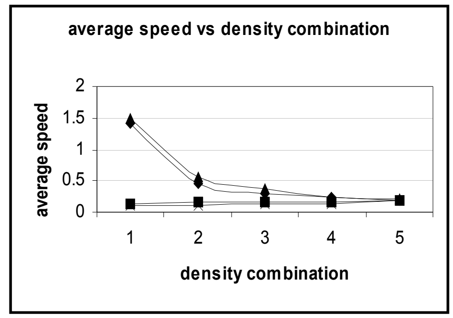

Figure 5.29: Speed vs Density at Light=.25

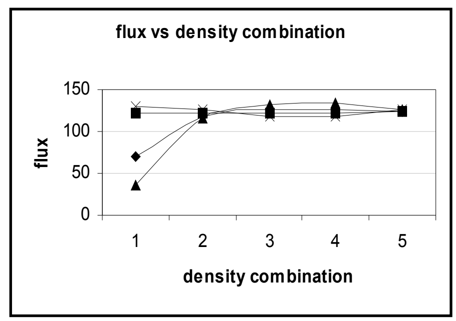

Figure 5.30: Flux vs Density at Light=.25

In this set-up, it can be noted that, at a combination of 20-180, there is a great difference in the average speed and the flux of the cars. As the density becomes equal however, the average speed and flux of the system approaches a certain range of values. This shows that, given a certain traffic light duration, one can find combinations in densities where there is great difference in the flux rate.

The last two graphs show that for the systems previously mentioned (varying the traffic light and the density), one can find a combination which can give the highest flux. For Figure 5.31 and Figure 5.32, one can see that the highest total flux occurs at a traffic light with a 90-90 combination. This means that, for an intersection with equal densities on each segment, an equal duration for the traffic light would also give the highest total flux.

In the case of Figure 5.33, one can infer that there is no clear and optimal traffic light combination. This can be attributed to the fact that the system has a large density. Varying the traffic light, in this case, will no longer give any significant and absolute increase in the flux of the system. The last graph (Figure 5.34) covers the case when the density is changed. This shows that as the density increases (given a constant traffic light), the total flux will increase. It must be observed, though, that as the density reaches a certain level, the increase in the flux will only be minimal.

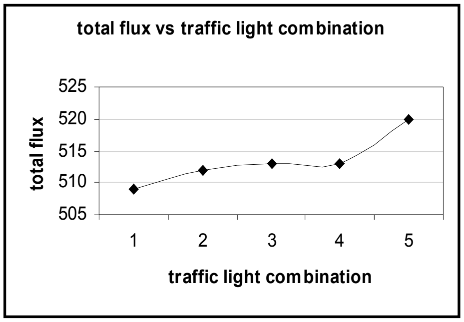

Figure 5.31: Flux vs Light at Density=0.25

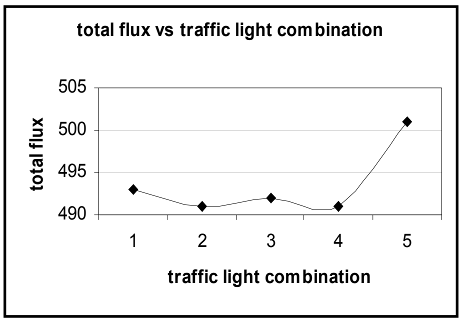

Figure 5.32: Flux vs Light at Density=0.50

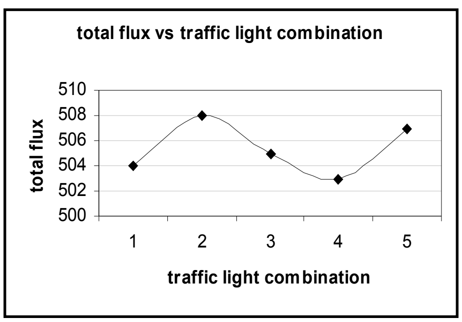

Figure 5.33: Flux vs Light at Density=0.75

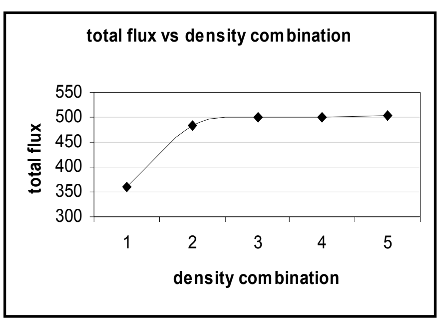

Figure 5.34: Flux vs Density at Light=0.25

6.CONCLUSION

A cellular automata-based two-lane intersection model of vehicular traffic dynamcis has been studied using computer simulations implemented in JAVA. The following is a summary of results:

- Simulations using the two-lane intersection CA model produces speed vs density plots which follow the behavior obtained using theories on vehicular traffic dynamics. Flux vs density graphs are not as similar due to the non-periodic aspect of the simulation.
- As the density of the system increases, the average speeds fluctuates less.
- When there is a large number of cars on the road, an increase in the maximum speed has little effect on the average speed.
- At low densities, the cars at the leftmost section of the roads move forward. At the same time, a decrease in the speed occurs at the latter section of the road. This is due to the forced lane-changing region.
- Changing the randomization parameter (noise) of the system affects the speed and flux on the road.
- As the density becomes equal, the average speed and flux of the system approaches a certain range of values
- For an intersection with equal densities on each segment (with low densities), an equal duration for the traffic light would also give the highest total flux
- As the density increases (given a constant traffic light), the total flux will increase. As the density reaches a certain level, the increase in the flux would be minimal.

7. ACKNOWLEDGEMENT

The authors would like to acknowledge funding support from the Commission on Higher Eduation (CHED) Center of Excellence Fund of the Mathematics Department of Ateneo de Manila University.

8.REFERENCES

[1] Nagel, K. and M. Schreckenberg (1992). "A Cellular Automaton Model for Freeway Traffic". J. Physique, Vol. 2, pp. 2221-2229.

[2] Saldaña, R. and Tabares, W. (2000). "Mathematical and Computatonal Aspects of Modeling and Simulation of Traffic Flow Dynamics". 19th Annual PAASE Meeting and Symposium. Makati City, Philippines.

[3] Saldaña, R. and Tabares, W. (2000). "Traffic Modeling on High Performance Computing Systems." Proceedings of the First Philippine Computing Science Congress (PCSCS 2000). Manila, Philippines.

[4] Saldaña, R. and Tabares, W. (2001). "A Study on Modeling Vehicular Traffic Dynamics: Comparison Between Macrosimulation and Microsimulation Methods." MODEL 2001. Mindanao State University-Iligan Institute of Technology, Iligan City, Philipppines.

[5] Saldaña, R. and Tabares, W. (2001). "A Cellular Automata-Based Study of Vehicular Traffic Dynamics." 2nd Philippine Computing Science Congress (PCSC 2001). Mindanao Stuate University-Iligan Instittue of Technology, Iligan City, Philippines.

[6] Tabares, Winfer C. (2002). "A Cellular Automata-Based Study of Vehicular Traffic Dynamics." M.S. Thesis, Ateneo de Manila University, Quezon City, Philippines.