

A Simulation on Traffic-signal Control

Xiangyu Kong GT# 903322862

Mingyi Liu GT# 903332202

Peiyuan Sun GT# 903319579

1. Introduction

Due to the increasingly number of the vehicles on road, bad traffic condition drives many citizens in big cities (like New York, atlanta., etc) crazy. There exists many vehicular traffic models on this topic. For example, the traditional Biham-Middleton-Levine Traffic Model model(BML)^[1], where the motion follows two-dimensional analogue of the simpler Rule 184 model. The Nagel-Schreckenberg model (NS)^[2], in which roads are aligned in single rows and boundaries are periodic. The main simulation includes four repeatable parts: the acceleration, slowing down, randomization and car motion. From the above basic models, many people extent their study on the optimization of the traffic lights^[3] using the language of cellular automata (CA). For example, Rui Q^[4] have the randomization probability be defined as function of the vehicle's stopped time and the result shows that shorter cycle time of traffic light leads to large saturated current so that we can adjust the cycle time of the traffic lights to enhance the road capacity. Chen S W^[5] devised and tested the genetic algorithm (GA), the particle swarm optimization (PSO) and the ant colony optimization (ACO) algorithm for obtaining the optimal solution of the traffic light setting problem, showing that GA seems to be a good strategy for setting traffic lights. Discrete event simulation model is also widely used in solving traffic problems. In Aljaafreh A's work^[6], a discrete event simulation model of traffic signal controller on a single intersection is developed using Matlab/Simulink/Simevents and results show that the proposed algorithm *AW Variable C* outperforms in reducing the average waiting time at intersections since it adapts both green intervals and cycle length. Also, Sumaryos S^[7] proposed the improved model using controller with two input switches and two output switches combined with traffic signal logic block and results show that the number of vehicles in queue and average waiting time of low traffic volume is lower than high traffic volume which means the model has functioned correctly. Our model is based on the mentioned CA models above and combined with the Discrete Event Simulation structure for better simulation.

Our goal is to simulate the traffic condition in a specific way. The main focus of our project lies on the traffic condition (i.e. whether it jams or not) under different circumstances, like heavy inflow of traffic, speed limits on road., etc, especially the duration of the traffic light time. For simplification, we only consider one lane condition and the intersected places like the crossroad and the traffic divergence will be ignored. We will analyze the global traffic light control strategies, in particular synchronized traffic lights and traffic lights with random offset.

2. Used software and the code structure

2.1 Used software

There exists many software simulating the traffic flow, like the Quadstone paramics (microscopic), VISSUM (macroscopic), VISSIM (microscopic), AIMSUN and SUMO (microscopic). We will use the VISSIM to make a microscopic simulation first and compare this to the results of the CA model we have built to see the differences. However, the disadvantages of the VISSIM is that it cannot simulate the jam condition under different traffic light time. As a result, we just use this software as a comparison.

2.2 The code structure

	Member 1	Member 2	Member 3
State	Helper File	Cars File	Traffic lights File
	length	velocity	red_on
	max number of cars	size	green_on
	number of traffic lights	location	red duration time
	number of lanes		green duration time
Event			number of wait cars
			road length
	Check Location()	Change Velocity()	Change color()
	Generate Cars()	Accelerate()	Count Wait Cars()
	Random Generator()	Slow down()	Check Jam()
		Change Line()	
		Next Location()	

Fig 2.1 The division of the project

2.3 Division of the Project:

Xiangyu Kong is responsible for the helper file which implement the functions that help the operation of the simulation such as checking the location of the cars, and random number generator, generate cars and other functions like these.

Peiyuan Sun is responsible for the cars file which implement the functions used for the cars that used for the cars such as changing velocity, accelerating and slowing down and so on.

Mingyi Liu is responsible for the traffic light file which implement the functions that used for the traffic lights such as changing color, counting waited cars and checking jam.

3. Simulation process

The simulation application models the operation of a 5km road with 8 traffic lights that handle three different types of cars travelling on the road. Incoming cars are assumed to follow a Poisson distribution with an average arrival rate of λ cars per seconds. The ratio of the three different types of cars follows this rule: (private car : bus : track = 80 : 10 : 10). Assume the velocity is different according to the type of car, set the private car's initial velocity between 15-20m/s, the initial velocity for bus 12m/s, and the initial velocity for track 10m/s.

3.1 Cars' type

Table 3.1 Different Types of Cars

Type of cars	Size	Initial Velocity	High velocity
Bus	15m	12m/s	15m/s
Track	30m	10m/s	12m/s
Private Cars	5m	15m/s -- 20m/s	25m/s

3.2 Car's Behaviors

3.2.1 Overtake and Change Lanes

Based on the figure 3.1, when the red car is following the white car within 5 meters or less, check the velocity of the white car. If the red car's velocity is larger than the white car, then the red car would check the lanes on either side.

- the d_{back} = the distance between the car following and the current car
- the d_{left} = distance between the front car in the side lane and the current car
- the d = distance between the front car and the target car in the current lane

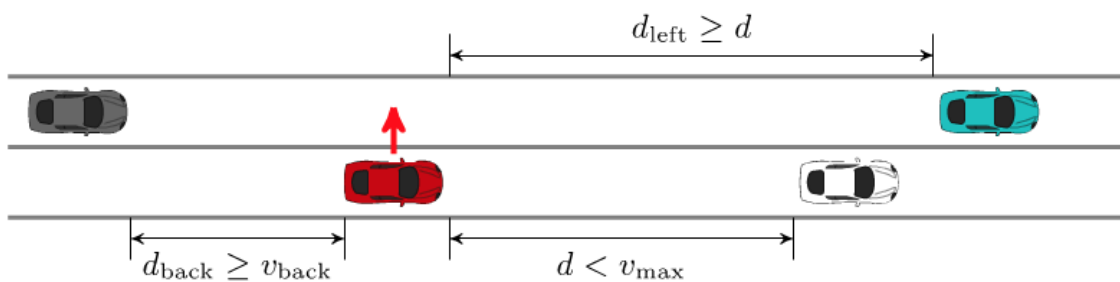


Fig 3.1 Illustration of overtaking

3.2.2 Rule of slowing down

When the distance between a car and a traffic light is smaller than 20m, check the color of the traffic light. If the traffic light is red, the car should slow down with an acceleration of 15m/s^2 . If the car is not the first car waiting for the traffic light, the car would just check the front car's behavior i.e. if the front car stop, then the car also stop.

3.2.3 Rule of accelerating

When the light becomes green, the car should have an acceleration of 10m/s^2 .

3.3 Assumption and Simplification of the Model

1. No pedestrians in our model
2. Each traffic light has only two signals, red light and green light.
3. All traffic lights have the same cycle time T , the total duration of one red light and one green light.
4. The duration of the red light is equal to that of the green light.
5. Each traffic light is implemented with a two-phase signal, where some vehicles pass through the intersection during the green light and the other vehicles pass through the intersection during the red light (their direction is green light).

6. Each intersection has two traffic lights with opposite signals. Therefore, a single traffic light is sufficient to represent these two lights.
7. Each vehicle can either move forward or stay at its current position. That is to say, vehicles are not allowed to turn left or turn right.
8. If a vehicle reaches an intersection and the traffic light is red, it must wait until the traffic light turns green.
9. Each vehicle moves one unit of distance forward in one unit of time, provided that it is not blocked by a front vehicle or a red traffic light.
10. The acceleration value for braking is 10m/s^2 and the acceleration value for accelerating is 5m/s^2

3.4 Outline of Model Steps

the following four actions are conducted in order from first to last, and all are applied to all cars. In each action, the updates are applied to all cars in parallel.

1. Acceleration: All private cars not at the maximum velocity have their velocity increased by accelerating. For example, if the velocity is 15 it is increased to 20m/s.
2. Slowing down: All cars are checked to see if the distance between it and the car in front (in units of cells) is smaller than its current velocity (which has units of cells per time step). If the distance is smaller than the velocity, the velocity is reduced to the number of empty cells in front of the car – to avoid a collision. For example, if the velocity of a car is now 20, but there are only 3 free cells in front of it, with the fourth cell occupied by another car, the car velocity is reduced to 3.
3. Randomization: The speed of all cars that have a velocity of at least 1, is now reduced by two unit with a probability of p . For example, if $p = 0.5$, then if the velocity is 10, it is reduced to 8 50% of the time.
4. Car motion: Finally, all cars are moved forward the number of cells equal to their velocity. For example, if the velocity is 10, the car is moved forward 10 cells.

It is possible that there would be some congestion situation in the simulation model:

1. When there are multiple traffic lights in a road, and the traffic light ahead is red, while the one behind is green, leaving the cars at the green light.
2. When the whole traffic lines are already filled with cars, so that incoming cars that are not able to enter because all the road are already full of cars.
3. When there is a congestion, even if the traffic light is green, the car will still wait in line instead of moving forward. If the whole road are filled with cars, the generation of the cars would stop.

References

- [1] Angel O, Holroyd A, Martin J. The jammed phase of the Biham-Middleton-Levine traffic model[J]. *Electronic Communications in Probability*, 2005, 10: 167-178.
- [2] Chowdhury D, Schadschneider A. Self-organization of traffic jams in cities: Effects of stochastic dynamics and signal periods[J]. *Physical Review E*, 1999, 59(2): R1311.
- [3] Brockfeld E, Barlovic R, Schadschneider A, et al. Optimizing traffic lights in a cellular automaton model for city traffic[J]. *Physical Review E*, 2001, 64(5): 056132.
- [4] Rui Q, Song W. A stopped time dependent randomization cellular automata model for traffic flow controlled by traffic light. *Physica A: Statistical Mechanics and its Applications*, 2006, 364: 493-496.
- [5] Chen S W, Yang C B, Peng Y H. Algorithms for the traffic light setting problem on the graph model[J]. *Taiwanese Association for Artificial Intelligence*, 2007: 1-8.
- [6] Aljaafreh A, Al-Oudat N. Adaptive traffic-signal control using discrete event simulation model[J]. *International Journal of Computer Applications*, 2014, 101(12).
- [7] Sumaryo S, Halim A, Ramli K. Improved discrete event simulation model of traffic light control on a single intersection[C]//*QIR (Quality in Research)*, 2013 International Conference on. IEEE, 2013: 116-120.