

Team Control Number

For office use only

T1 _____

T2 _____

T3 _____

T4 _____

30608

Problem Chosen

A

For office use only

F1 _____

F2 _____

F3 _____

F4 _____

2014 Mathematical Contest in Modeling (MCM) Summary Sheet

(Attach a copy of this page to your solution paper.)

Type a summary of your results on this page. Do not include the name of your school, advisor, or team members on this page.

Is it a good rule to keep right except to pass?

Our model has two parts: to analyze and evaluate the performance of the keep-right-except-to-pass rule, and to extend or change our model to other situations.

Firstly, we develop a Cellular Automata traffic flow model, which consist of three progressive schemes: Scheme One (single-lane road without passing), Scheme Two (multi-lane road with free driving) and Scheme Three (multi-lane road under the keep-right-except-to-pass rule). The three schemes range from simple to complex and differ from concrete evolutionary rules.

Then, considering tradeoffs between traffic flow and safety, we formulate an evaluation system to study the performance of the three schemes in light and heavy traffic. We utilize Cellular Automata to simulate the traffic system, so that we can obtain the characteristics of traffic flow. With carrying out numerical simulations on evolutionary rules of the three schemes, we are able to specifically compare. We input the data of expected traffic flow and maximum and minimum speed limit and output the data of the number of times where there may exist risks and actual traffic flow.

As a result, the keep-right-except-to-pass rule (Scheme Three) can slightly promote traffic flow in light traffic and has high security. It will decrease traffic flow and promote security in heavy traffic. Generally speaking, Scheme Two is effective in heavy traffic; Scheme One is effective in extremely heavy traffic and Scheme Three is effective in light traffic.

Next, in order to adapt our model to right-hand traffic, we focus the difference between left-hand traffic and right-hand traffic on psychological and physiological factors of drivers, which will affect the randomization process. Then we utilize AHP to obtain weighted values of right-hand and left-hand traffic, using which we obtain the randomization deceleration probability is 0.43 in left-hand traffic and 0.4 in right-hand traffic.

Finally, we find that it will greatly change the results of our previous study by introducing intelligent system to vehicle transportation, since the former one is greatly based on random processes while the latter one is exclusively with no random process.

Is it a good rule to keep right except to pass?

February 11, 2014

Our model has two parts: to analyze and evaluate the performance of the keep-right-except-to-pass rule, and to extend or change our model to other situations.

Firstly, we develop a Cellular Automata traffic flow model, which consist of three progressive schemes: Scheme One (single-lane road without passing), Scheme Two (multi-lane road with free driving) and Scheme Three (multi-lane road under the keep-right-except-to-pass rule). The three schemes range from simple to complex and differ from concrete evolutionary rules.

Then, considering tradeoffs between traffic flow and safety, we formulate an evaluation system to study the performance of the three schemes in light and heavy traffic. We utilize Cellular Automata to simulate the traffic system, so that we can obtain the characteristics of traffic flow. With carrying out numerical simulations on evolutionary rules of the three schemes, we are able to specifically compare. We input the data of expected traffic flow and maximum and minimum speed limit and output the data of the number of times where there may exist risks and actual traffic flow.

As a result, the keep-right-except-to-pass rule (Scheme Three) can slightly promote traffic flow in light traffic and has high security. It will decrease traffic flow and promote security in heavy traffic. Generally speaking, Scheme Two is effective in heavy traffic; Scheme One is effective in extremely heavy traffic and Scheme Three is effective in light traffic.

Next, in order to adapt our model to right-hand traffic, we focus the difference between left-hand traffic and right-hand traffic on psychological and physiological factors of drivers, which will affect the randomization process. Then we utilize AHP to obtain weighted values of right-hand and left-hand traffic, using which we obtain the randomization deceleration probability is 0.43 in left-hand traffic and 0.4 in right-hand traffic.

Finally, we find that it will greatly change the results of our previous study by introducing intelligent system to vehicle transportation, since the former one is greatly based on random processes while the latter one is exclusively with no random process.

Contents

1	Introduction	3
1.1	Defining the Problem	3
1.2	Model Overview	3
2	Conventions	4
2.1	Terminology	4
2.2	Variables	5
3	Assumptions and Justifications	6
4	Motivation and sub-schemes	7
4.1	Scheme One: Single-lane Road without Passing	8
4.2	Scheme Two: Multi-lane Road with Free Driving	10
4.3	Scheme Three: Multi-lane Road under the Keep-Right-Except-To-Pass Rule	13
4.4	Evaluation System of Three Schemes	15
4.5	Simulation	17
5	Discussions based on the Driving-On-The-Left Rule	21
5.1	Determination of Weights	22
5.2	How to change	25
6	Discussions based on the Intelligent System	25
7	Strengths, Weaknesses, and Extensions	27
	References	27
	Appendices	28
	Appendix A output data of our algorithm	28
	Appendix B <i>two – laneCAmodelbasedonright – handtraffic</i> MATLAB finish	28

1 Introduction

The keep-right-except-to-pass rule is today hotly debated among people who have to drive on the right. People who oppose the rule do not think that vacant left lanes and more vehicles in right lanes will make them a bit safer [1]. On the contrary, people who support the rule argue that failure to keep right has a negative effect on traffic safety, traffic congestion, emergency response, aggressive driving, air pollution, fuel consumption, direct and indirect medical costs and much more [2].

1.1 Defining the Problem

With regard to the keep-right-except-to-pass rule, we ought to:

- Analyze and evaluate the performance of the keep-right-except-to-pass rule in light and heavy traffic. We may consider such factors as tradeoffs between traffic flow and safety, the role of maximum or minimum speed limits, and other possible factors.
- Examine the effectiveness of the keep-right-except-to-pass rule in promoting better traffic flow. If our answer is "no", we also need to put forward a better alternative.
- Accommodate our solution to the conditions in countries where people are required to drive on the left.
- Discuss the possible change of our results with the existence of an intelligent system.

1.2 Model Overview

In order to analyze and evaluate the performance of the keep-right-except-to-pass rule, we need to put forward some comparisons. With the large number of complexities involved in the vehicle transportation system, we begin by analyzing the problem in small parts. We begin by discussing about Scheme One: single-lane highway without passing behaviors. By looking at the simple situation, we formulate a series of updating rules. Next, we develop Scheme Two: multi-lane road with free driving, which only add lane-changing rules to Nasch one-lane model. Then, we study Scheme Three: Multi-lane Road under the Keep-Right-Except-To-Pass Rule and we need to add the lane-changing rules corresponding to the case that vehicles return to the travel lane from the passing lane. In order to analyze the performance of the keep-right-except-to-pass rule in light and heavy traffic, we formulate an evaluation system of three schemes above-mentioned.

Finally, we carry out numerical simulations on evolutionary rules of the three schemes. We don't need to give concrete formulas when utilizing Cellular Automata model. We only have to provide the corresponding evolutionary rules. In this way, we can reproduce various complicated traffic phenomenon through the micro interaction among cells. Thus CA model is effective for us to study the characteristics of traffic flow. As we previously analyzed, the randomization procession is affected by various uncertainties, which will finally affect the randomization deceleration probability. Uncertainties include over-braking, changes in road conditions, psychological factors and delay in accelerating. Thus, we can find out significant differences in the respect of the above-mentioned factors between left-hand traffic and right-hand traffic. Then we can utilize Analytic Hierarchy Process (AHP) to figure out how left-hand traffic and right-hand traffic differently affect the randomization deceleration probability, which will be expressed by an array of weights. Finally, we can determine how to change our solution accordingly.

2 Conventions

This section defines the basic terms used in this paper.

2.1 Terminology

- **The keep-right-except-to-pass rule:** Drivers should keep driving in the right-most side lane unless they are passing another vehicle, in which case they move to the left lane, pass, and return to their former travel lane. **Figure 1** shows how it works.

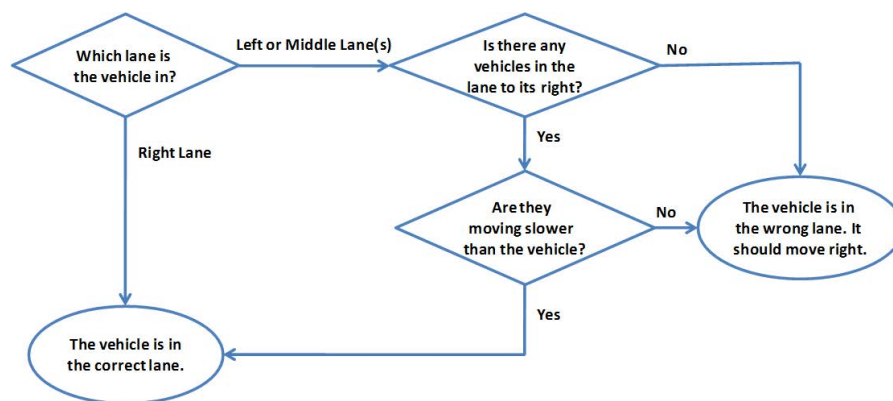


Figure 1 .Flow chart of the keep-right-except-to-pass rule [3]

- **Passing lane and travel lane:** Passing lanes are provided for drivers to pass another vehicle under the keep-right-except-to-pass rule. Both left

lane and middle lane(s) are passing lanes. In our study, we ignore the differences between left lane and middle lane(s), which means that they can be treated equally. Travel lane is provided for drivers who don't want to pass any vehicles and right lane is the travel lane. Moreover, we can say that the speed of vehicles varies from slowest to fastest from right lanes to left lanes. The sketch of passing lane and travel lane is shown in **Figure 2**.

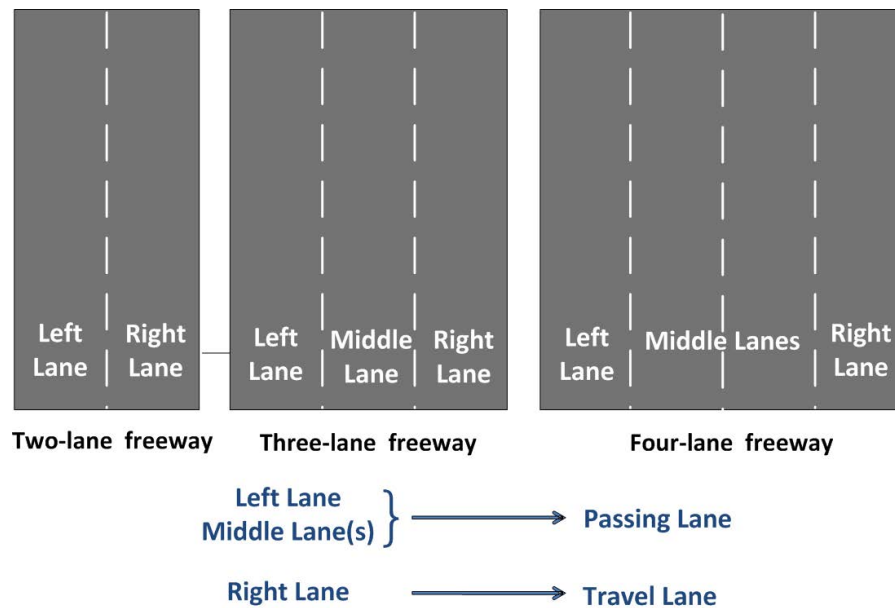


Figure 2 . Sketch of passing lane and travel lane

- **Light and heavy traffic:** Light and heavy traffic are two traffic states, which can be represented by the possible number of vehicles that arrive at a certain cross section per unit time. Different traffic states result from different probability of vehicles' arrival. Therefore, we use the expected traffic flow to describe traffic states. High probability corresponds to heavy traffic while low probability corresponds to light traffic.
- **Actual traffic flow:** Actual traffic flow is the actual number of vehicles that arrive at a certain cross section in a certain period of time. In our paper, we use it to evaluate the performance of a certain traffic rule in terms of traffic flow.

2.2 Variables

We will define the following variables here as they are used widely throughout our paper. Additional variables may be defined later, but will be confined to a particular section.

Variable Definition

$V_i(t)$	Speed of the i^{th} vehicle at the time of t
V_{min}	Minimum speed limit
V_{max}	Maximum speed limit
x_i	Position of the i^{th} vehicle at the point of t
p	Randomization deceleration probability
d_i	Number of null cells between the i^{th} vehicle and the vehicle ahead (located in the current lane)
d_i'	Number of null cells between the i^{th} vehicle and the back vehicle in the travel lane
D_i	Number of null cells between the i^{th} vehicle located in the current lane and the vehicle ahead which is located in the adjacent lane
D_i'	Number of null cells between the i^{th} vehicle located in the current lane and the back vehicle which is located in the adjacent lane
μ	Security coefficient
q	Expected traffic flow
Q	Actual traffic flow
M	Number of times where there may exist risks
$\bar{V}$	Average speed of vehicles
ρ	Vehicle density

3 Assumptions and Justifications

We make the following assumptions about vehicle transportation in this paper.

- Flux is a single function of density at every point of the road.

This assumption means that the traffic flow of stationary state lays on a curve that passes through the origin and has at least one maximum. This is a basic assumption of traffic flow theory, which is formulated based on Fundamental Diagram Approach.

- We assume that all drivers drive conservatively.

This assumption ensures that each cell (vehicle) accelerates with an acceleration of $1\text{cell}/s^2$ and slows down with an acceleration of $-1\text{cell}/s^2$. That is $a = \pm 1\text{cell}/s^2$

- Drivers always expect to drive at the minimum speed under safe conditions.

In reality, it is the most basic driving behavior that drivers will drive at a maximum safe speed.

- All the drivers will obey traffic rules, which particularly refer the keep-right-except-to-pass rule.

This assumption ensures that all the passing vehicles will definitely return to the travel lane after passing another vehicle.

- We assume that there exists no minimum speed limit on single-lane highway.

Under normal condition, vehicles cannot drive under the minimum speed limit on the highway. However, it is impossible to pass another vehicle on single-lane highway. In this case, vehicles can only slow down in the case of automobile accidents or other special situation. Therefore, vehicles have to drive under the minimum speed or even stop. Speaking from this angle, this assumption is reasonable.

- We assume that there are ideal traffic conditions and road conditions for vehicles to move forward, including that each vehicle drive continuously on the highway; road conditions are good and vision is wide and clear; traffic capacity will not be affected by intersections and toll booths.

This is a simplification which allows our model to produce more meaningful results.

- We assume that traffic flow, safety, maximum and minimum speed limits are the most significant factors affecting the traffic system

Additional assumptions are made to simplify analysis for individual sections. These assumptions will be discussed at the appropriate locations.

4 Motivation and sub-schemes

With the large number of complexities involved in the vehicle transportation system, we begin by analyzing the problem in small parts. We begin by discussing about Scheme One: single-lane highway without passing behaviors. By looking at the simple situation, we formulate a series of updating rules. Next, we develop Scheme Two: multi-lane road with free driving, which only add lane-changing rules to Nasch one-lane model. Then, we study Scheme Three: Multi-lane Road under the Keep-Right-Except-To-Pass Rule and we need to add the lane-changing rules corresponding to the case that vehicles return to the travel lane from the passing lane.

In order to analyze the performance of the keep-right-except-to-pass rule in light and heavy traffic, we formulate an evaluation system of three schemes above-mentioned.

Finally, we carry out numerical simulations on evolutionary rules of the three schemes. We don't need to give concrete formulas when utilizing Cellular Automata model. We only have to provide the corresponding evolutionary rules. In this way, we can reproduce various complicated traffic phenomenon through the micro interaction among cells. Thus CA model is effective for us to study the characteristics of traffic flow.

4.1 Scheme One: Single-lane Road without Passing

Although single-lane highway is not common in reality, it does provide guidance for us to gain insights into the nature of the problem.

Time, space and speed are discretized in Cellular Automata traffic flow model. Generally speaking, CA model basically consists of four parts: cell, cellular space, cellular neighbor and updating rules.

1. Cell

Highway is divided into several discrete grids (cells) and one cell corresponds to one vehicle. The state of each cell is null or represents the speed of the contained vehicle, and the speed of each vehicle can take integer values between the minimum speed limit and maximum speed limit. That is

$$V_{min} \leq V_i \leq V_{max} V_i \in Z$$

Where V_{max} is the minimum speed limit and V_{min} is the maximum speed limit.

2. Cellular space

Cellular space is the set of all lattice points that contain cells and cellular space particularly refers to the overall road system in which all the vehicles locate. Theoretically, cellular space can extend infinitely in each dimension. In order to implement cellular space through computer, we define the following boundary conditions:

(a) Entry:

- Entry condition: $x_{last} > V_{max}$, where x_{last} is the position of the last vehicle;
- Entry speed: V_{max}
- Entry position: $\min(x_{last} - V_{max}, V_{max})$
- Entry probability: α

(b) Departure

If $x_{lead} > L_{road}$, then vehicle will depart with a certain probability β . Note that x_{lead} is the position of the lead vehicle

3. Cellular neighbor

Each grid (cell) is consisted of an actual state or a null state, which is

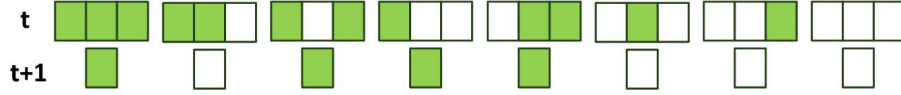


Figure 3 .Flow chart of the keep-right-except-to-pass rule [3]

represented by a green grid and white grid, relatively. See Figure 3 Each cell will update itself according to the state of the cells on the left and right sides. As shown in **Figure 3**, the combination state of the cells on the left and right of any cell can be summarized as follows:

$$111, 110, 101, 011, 100, 010, 000$$

Where "1" represents an actual state and "0" represents a null state.

4. Updating rule

NaSch traffic flow model is a random model. The state of each vehicle is represented by its speed and position. Each vehicle will concurrently update its state according to the following evolutionary rules:

As we previously assumed, all drivers drive conservatively, which means that each cell (vehicle) accelerates with an acceleration of $1\text{cell}/s^2$ and slows down with an acceleration of $-1\text{cell}/s^2$. That is $a = \pm 1\text{cell}/s^2$

(a) Acceleration process:

As the previous assumption states, Drivers always expect to drive at the minimum speed under safe conditions, then we have

$$V_i(t+1) = \min(V_i(t) + 1, V_{max})$$

Where $V_i(t)$ is the speed of the i^{th} vehicle at the time of t .

(b) Safe braking process:

This process accounts for the deceleration measures taken to avoid crashing into another vehicle. So we have

$$V_i(t+1) = \min(V_i(t), d_i - 1)$$

Where d_i is the number of null cells between the i^{th} vehicle and the vehicle ahead (the $(i+1)^{th}$ vehicle). Its calculating formula is

$$d_i = x_{i+1}(t) - x_i(t) - l_{veh}$$

Where x_i represents the position of the i^{th} vehicle at the point of t , and l_{veh} is vehicle length

(c) Randomization process:

Randomization is a deceleration phenomenon caused by various uncertainties, which can be expressed as

$$V_i(t+1) = \max(V_i(t) - 1, V_{\min}) \text{ with a probability } p$$

Where p is the randomization deceleration probability, and $V_{\min}$ is the minimum speed limit, whose role is to stop the randomization process while the vehicle has currently reach the minimum speed limit.

Moreover, various uncertainties that affect the randomization process are listed as follows:

- Over-braking
- Changes in road conditions
- Changes in road conditions
- Delay in accelerating

(d) Update in position

Since vehicles will move forward at the adjusted speed, then we have

$$x_i(t+i) = x_i(t) + V_i(t)$$

Where x_i represents the position of the i^{th} vehicle at the point of t .

Of the four steps, the first three account for the update in the speed of vehicles while the fourth step accounts for the update in the position of vehicles. In one dimension cellular automation model, once we have defined the next moment's state of each possible current cell in eight situations, which are shown in Figure 3, we can completely define the rule of such a cellular automation. Thus, each vehicle is able to move simultaneously in line with the given rule.

We use an example to show how the four evolutionary rules work in **Figure 4**.

Note: This figure demonstrates the operation of the four evolutionary rules. Take Step1 (Acceleration process) as an example: the vehicle has a maximum speed of two *cells/s* and acceleration of one *cell/s*², so the vehicle firstly moves forward two grids and then move forward three grids.

4.2 Scheme Two: Multi-lane Road with Free Driving

Compared with the scheme of single-lane road without passing, lane-changing rules are added to Scheme Two because of the existence of passing behaviors. Therefore, with regard to the analysis of multi-lane road with free driving, we only need to determine the lane-changing rules to Nasch one-lane model. Here,

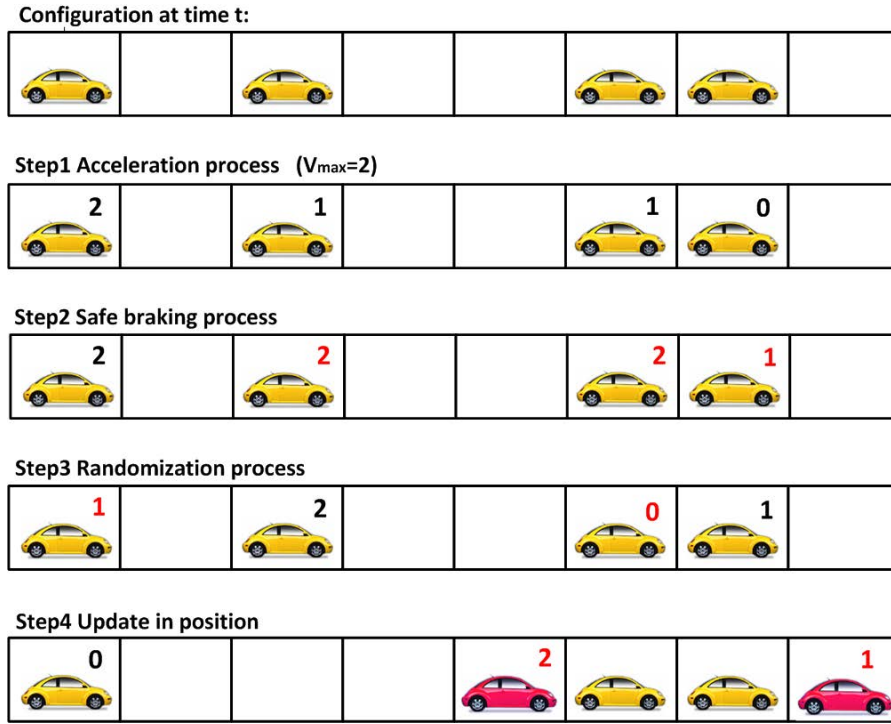


Figure 4 .Operation of the four evolutionary rules

lane-changing rules particularly refer to those in the case that vehicles change their lane from right to left.

In Nasch multi-lane model, vehicles driving in various lanes should comply with not only updating rules but also lane-changing rules. We set lane-changing rules according to the preconditions for changing lanes, which consist of changing-lane motivations and safety condition [5].

1. Changing-lane motivations:

Changing-lane motivations refer to the intension that whether a certain vehicle wants to change lanes or not.

- (a) **Motivation One:** The driving conditions of the adjacent lane are better than the current one. This motivation can be expressed as

$$D_i > d_i$$

Where d_i is the number of null cells between the i^{th} vehicle and the vehicle ahead (located in the current lane); D_i is the number of null cells between the i^{th} vehicle located in the current lane and the vehicle ahead which is located in the adjacent lane. We use an example to show the meaning of Motivation One in **Figure 5**.

Note: This figure demonstrates the meaning of Motivation One. As shown in Figure 5, we set Green Car as the main subject. The number

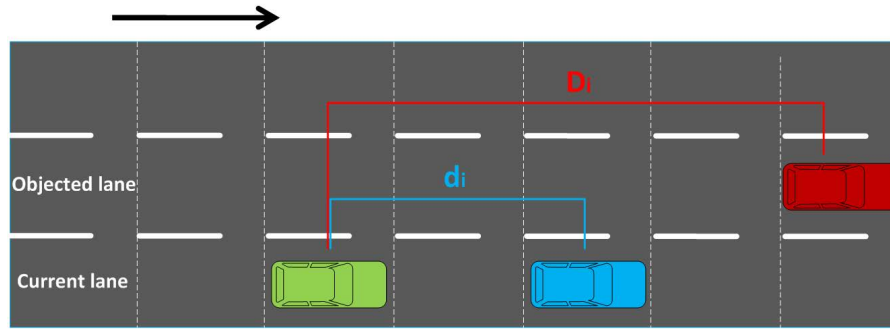


Figure 5 .Meaning of Motivation One

of null cells between Green Car and Blue Car is two, and the number of null cells between Green Car and Red Car is four. In this case, we can say that there exists Motivation One.

- (b) **Motivation Two:** A certain vehicle cannot drive at its expected speed in the current lane. This motivation can be expressed as

$$d_i < \min(V_i(t) + 1, V_{max})$$

We use an example to show the meaning of Motivation Two in **Figure 6**.

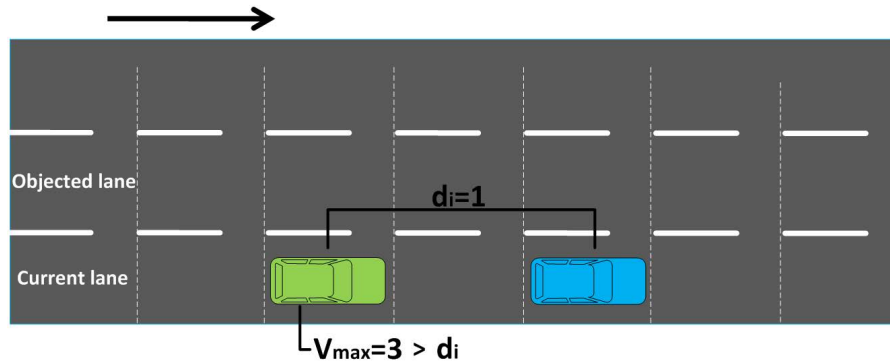


Figure 6 .Meaning of Motivation Two

Note: This figure demonstrates the meaning of Motivation Two. As shown in Figure 6, the number of null cells between Green Car and Blue Car is one, which is smaller than the maximum speed. In this case, we can say that there exists Motivation Two.

2. Safety condition:

Driver will consider the safety of both themselves and other drivers while changing their lanes. Safety condition is used to ensure that a certain vehicle would not crash into another vehicle after changing its lane. We can

express the safety condition as

$$D'_i > V_{max}$$

Where D'_i represents the number of null cells between the i^{th} vehicle located in the current lane and the back vehicle which is located in the adjacent lane.

We use an example to show the meaning of the safety condition in **Figure 7**.

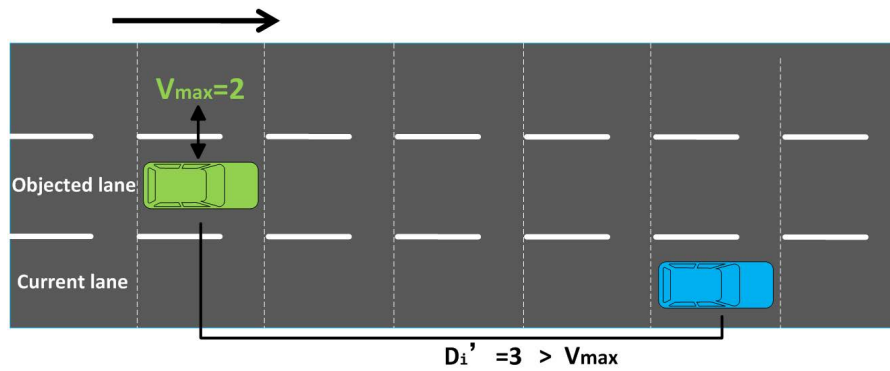


Figure 7 .Meaning of Motivation Two

Note: This figure demonstrates the meaning of the safety condition. As shown in Figure 7, the number of null cells between Green Car and Blue Car is three, which is greater than the maximum speed. In this case, we can say that there exists the safety condition. To sum up, lane-changing rules are

$$\begin{cases} D_i > d_i \\ d_i < \min(V_i(t) + 1, V_{max}) \\ D'_i > V_{max} \end{cases}$$

If all the rules listed above exist, the certain vehicle will change lane with a certain probability, which represents how willing the driver is when there exists the lane-changing rules.

Vehicles in each lane need to update simultaneously after changing its lane according to the updating rules.

4.3 Scheme Three: Multi-lane Road under the Keep-Right-Except-To-Pass Rule

As the precious terminology stated, under the keep-right-except-to-pass rule the passing vehicle needs to move from the travel lane to the passing lane, and finally return to the travel lane. In this case, there exist two sets of lane-changing

rules. One corresponds to the case that vehicles move from the travel lane to the passing lane, the other corresponds to the case that vehicles return to the travel lane from the passing lane.

As for the vehicles that change their lane to the passing lane, the lane-changing rules are the same as those of Scheme Two. Thus, with regard to the analysis of multi-lane road under the keep-right-except-to-pass rule, we only need to add the lane-changing rules corresponding to the case that vehicles return to the travel lane from the passing lane.

The lane-changing rules corresponding to the case that vehicles return to the travel lane from the passing lane are listed below: As we previously assumed, all the passing vehicles will definitely return to the travel lane after passing another vehicle.

1. Driving motivation

We should ensure that the vehicle can drive at its expected speed after returning to its original lane. We can express this as

$$d_i > \min(V_i + 1, V_{max})$$

We use two examples to show the meaning of the driving motivation in Figure 8.

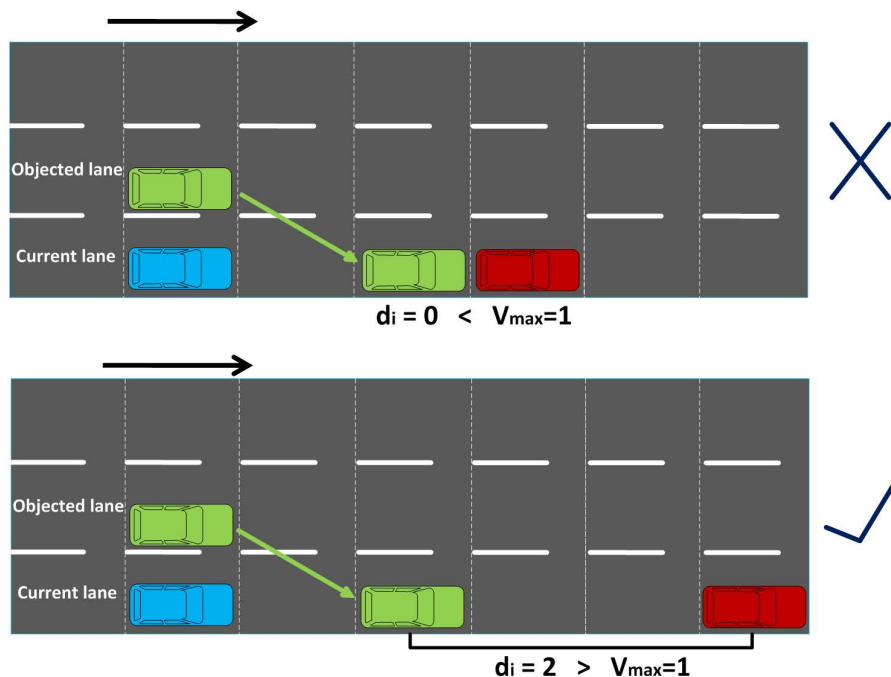


Figure 8 .Meaning of the driving motivation

Note: This figure demonstrates the meaning of driving motivation. As shown in Figure 8, the upper sketch shows that the driving motivation has

not been satisfied while the lower sketch shows that the driving motivation has been satisfied.

2. Safety condition

We should ensure that the vehicle wouldn't crash into another vehicle in the travel lane. We can express this as

$$d'_i > V_{max}$$

Where d'_i is the number of null cells between the i^{th} vehicle and the back vehicle in the travel lane.

We use an example to show the meaning of the safety condition in **Figure 9**.

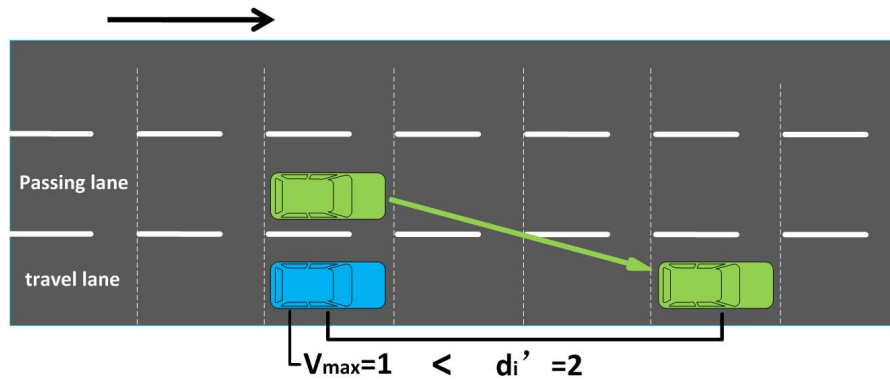


Figure 9 .Meaning of the safety condition

Note: This figure demonstrates the meaning of the safety condition. As shown in Figure 9, after changing its lane, the number of null cells between Green Car and Blue Car is two, which is greater than the maximum speed. In this case, we can say that there exists the safety condition.

Once the rules above are satisfied, the passing vehicle will return to the travel lane immediately after passing another vehicle. Next, vehicles in each lane should update themselves according to the updating rules. Note that this procedure doesn't contain the randomization process since the vehicle wants to return to the original lane as soon as possible.

If the rules above cannot be satisfied, vehicles in each lane should update themselves (this procedure doesn't contain the randomization as well) according to the updating rules until the two rules are satisfied.

4.4 Evaluation System of Three Schemes

In order to analyze the performance of the keep-right-except-to-pass rule in light and heavy traffic, we formulate an evaluation system of three schemes above-mentioned.

As the previous terminology defines, light and heavy traffic are two traffic states. We use the expected traffic flow to describe traffic states. Therefore, traffic states can be represented by the possible number of vehicles that arrive at a certain cross section per unit time. We set the expected traffic flow as q .

1. Evaluation in terms of traffic flow

As we mentioned in the terminology section, we use the actual traffic flow to evaluate the performance of a certain traffic rule in terms of traffic flow. Specifically, the actual traffic flow is the actual number of vehicles that arrive at a certain cross section per unit time. We set the actual traffic flow as Q .

2. Evaluation in terms of safety

In reality, traffic accidents happen when the driver is so careless that he hasn't kept a safe distance from the car in front, or the driver didn't react correctly and timely.

- **As for single-lane highway without minimum speed limit:**

According to our previous assumption, there exists no minimum speed limit on single-lane highway.

Yang et al. proposed some conditions to judge whether traffic accidents will happen or not based on the consideration of effects of randomization process on traffic accidents [10]. There is also no minimum speed limit in their study, which can fit in with our study.

As for the case of single-lane highway, we define the following events:

$$A_1 = \{V_i(t+1) = d_i(t)\} \quad (1)$$

$$A_2 = \{V_{i+1}(t) > 0\} \quad (2)$$

$$A_3 = \{V_{i+1}(t+1) = 0\} \quad (3)$$

If event A represent the case that the above-mentioned events coexist simultaneously, then we have

$$A = A_1 \cap A_2 \cap A_3$$

- **As for multi-lane highway with minimum speed limit:**

Vehicles can keep their speed over the minimum speed limit even if traffic accidents happen, since they can change lanes on multi-lane highway.

As for the case of multi-lane highway, we define event B as: a certain vehicle pass another vehicle.

If event represents traffic accident happens.

As for single-lane highway, we set p_1 as the probability that the i^{th} vehicle will crash into the vehicle ahead in the case of event A . Then we have

$$p_1 = P(C|A)$$

As for multi-lane highway, we set as the probability that the i^{th} vehicle will crash into the vehicle ahead in the case of event B . Then we have

$$p_2 = P(C|B)$$

We define the security coefficient μ as the probability of no traffic accident for a certain time, upon satisfying the conditions of the happening of traffic accidents. That is

$$\mu = (1 - p_k)^{N_k} \text{ for } k = 1, 2$$

Where N_K is the number of times where event A or B appeared during a certain period of time T . $k = 1$ represents the case of single-lane highway while $k = 2$ represents the case of multi-lane highway.

4.5 Simulation

We don't need to give concrete formulas when utilizing Cellular Automata. We only have to provide the corresponding evolutionary rules. In this way, we can reproduce various complicated traffic phenomenon through the micro interaction among cells. Therefore, we utilize Cellular Automata to simulate the traffic system, so that we can obtain the characteristics of traffic flow. We carry out numerical simulations on evolutionary rules of the three schemes. Finally we compare and analyze the results we obtain.

1. Preparation of our simulation

In our simulation, we set that each lane is consisted of 1000 grids (cells). Each cell's actual length is 7.5 m and the actual length of corresponding highway is 7.5 km. one vehicle corresponds to one grid. We simulate the changing process of the vehicles on the highway in 60 minutes (equal to 4200 time steps). In order to eliminate the influences of initial configuration's randomness, we only adopt the results of the last 50 minutes. See the flow chart of our simulation in Figure 10. In our simulation, concrete values of the parameters are listed in Table 1. Note: The probability of lane-

Table 1 Concrete values of the parameters

Entry probability α	0.1-1
Departure probability β	0.9
Randomization deceleration probability p	0.4
Probability of lane-changing in Scheme Two	0.4
Probability of lane-changing in Scheme Three	0.6
Probability of risks in Scheme Two	0.0003
Probability of risks in Scheme Three	0.000001

changing of Scheme Three is greater than that of Scheme Two. It is because

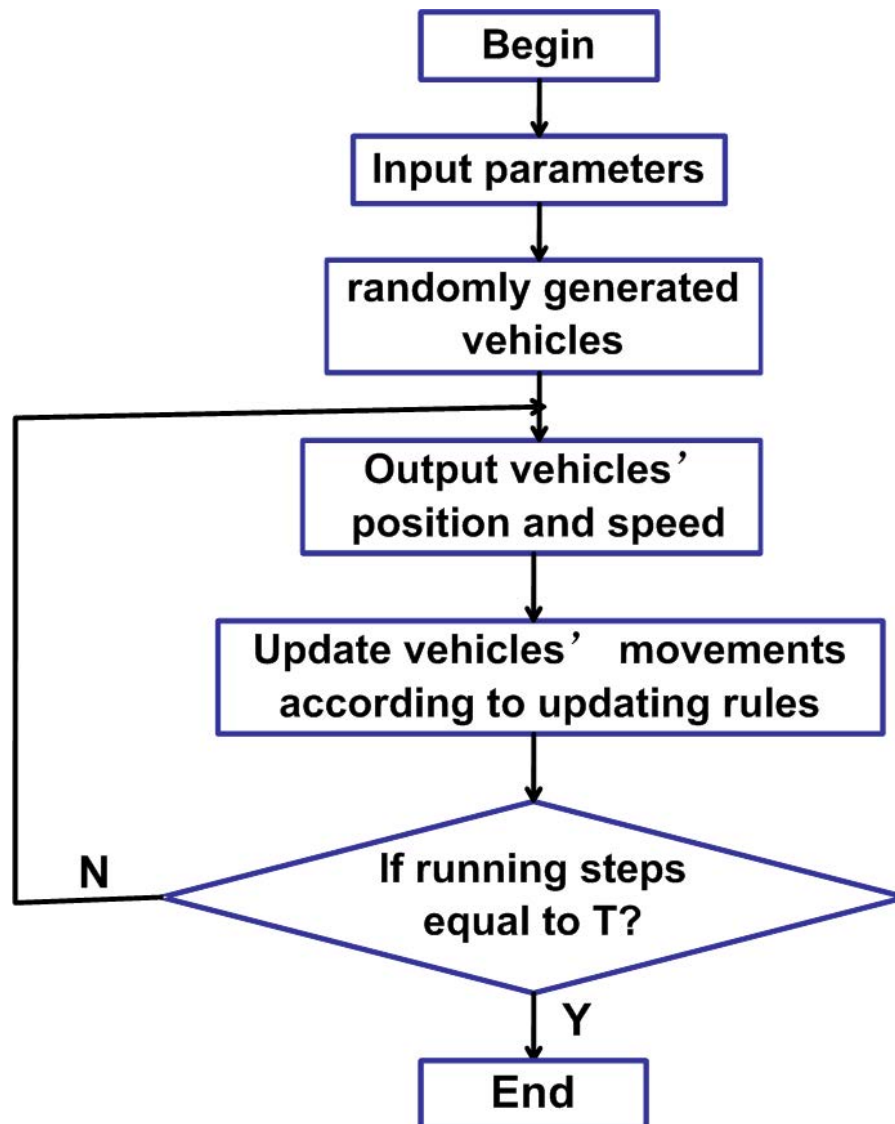


Figure 10 Flow chart of our simulation

that the security of overtaking is higher under the keep-right-unless-to-pass rule. Thus, drivers' desire for lane-changing will be stronger.

Apart from the data listed in Table 1, we also input arrays of data into the simulation system, including the data of the expected traffic flow and the combined data of minimum and maximum speed limit. For concrete values, see Table 2 and Table 3

Table 2 Concrete values of the expected traffic flow

Expected traffic flow	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
-----------------------	-----	-----	-----	-----	-----	-----	-----	-----	-----	---

Table 3 Concrete values of minimum and maximum speed limit

Sequence Number	minimum and maximum (cell/s)	maximum speed limit (cell/s)
1	0	5
2	0	6
3	1	5

2. Results of our simulation

We can output numeral results of four important variables: the number of times where there may exist risks M , actual traffic flow Q , average speed of vehicles $\bar{V}$ and vehicle density ρ . Through our simulation we finally output the numeral results of the variables above-mentioned, which are listed in the appendix.

According to the data we obtain, we plot four charts, which are shown in Figure 11,12,13,14

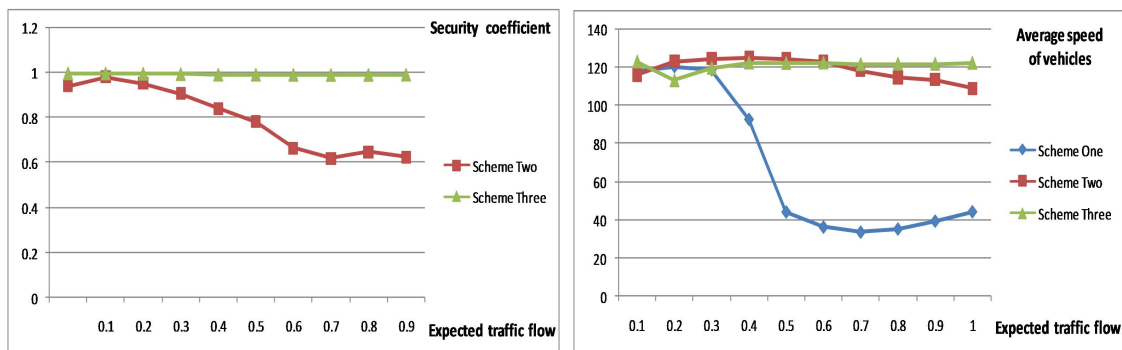


Figure 11 .expected traffic flow-security co-efficient Figure 12 . expected traffic flow-average speed of vehicles

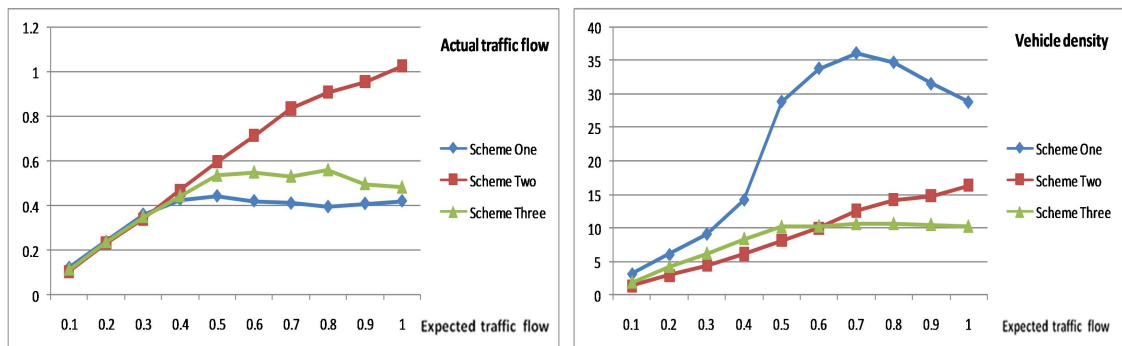


Figure 13 .expected traffic flow-actual traffic flow Figure 14 .expected traffic flow-vehicle density

We compare the performance of Scheme Three with that of Scheme One and Scheme Two. The results are listed in Table 4

Table 4 Concrete values of minimum and maximum speed limit

Item	security coefficient	average speed of vehicles	actual traffic flow
Light traffic	Better than S2	No significant difference	No significant difference
Heavy traffic	Better than S2	Better than S1 and S2	Better than S1 Worse than S2

Scheme One: single-lane road without passing (abbreviated S1)

Scheme Two: multi-lane road with free driving (abbreviated S2)

Scheme Three: multi-lane road under the keep-right-except-to-pass rule (abbreviated S3)

In order to analyze the role of speed limit, we change the values of maximum and minimum speed limit. The numeral data is listed in the appendix an we also plot four charts, which are shown in Figure 15,16,17,18.

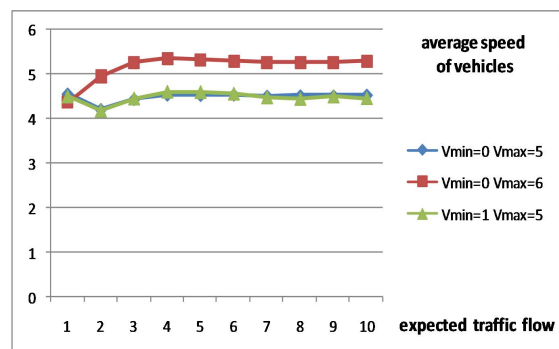
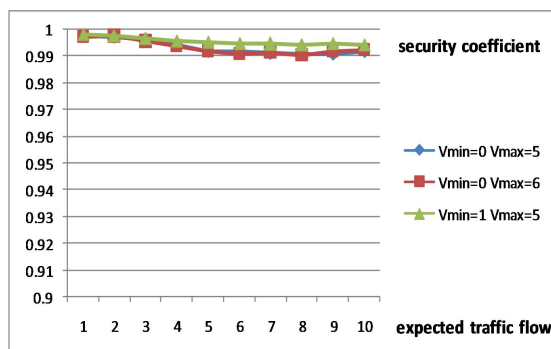


Figure 15 .Effect of speed limit on security coefficient Figure 16 .Effect of speed limit on average speed of vehicles

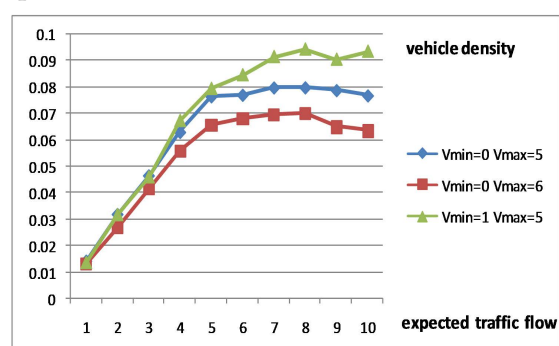
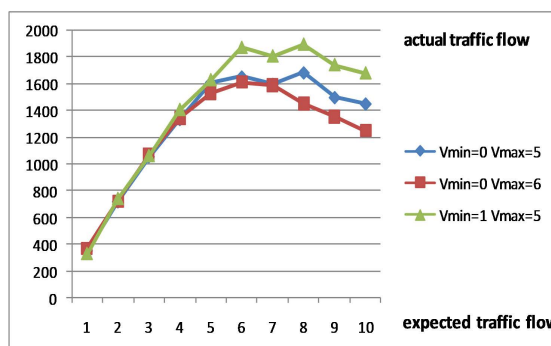


Figure 17 .Effect of speed limit on average actual traffic flow Figure 18 .Effect of speed limit on vehicle density

From the charts showing above, we can find that the minimum speed limit helps to promote the actual traffic flow, safety coefficient and average speed of vehicles. Moreover, with the increase of the maximum speed

limit, the actual traffic flow will slightly increase and the vehicle density will be smaller, but the safety coefficient will decline as well.

Generally speaking, the keep-right-except-to-pass rule can slightly promote traffic flow in light traffic and has high security. It will decrease traffic flow and promote security in heavy traffic, in which case the highway cannot be used effectively. Therefore, we put forward some ideas:

- Scheme Two (multi-lane road with free driving) is effective in heavy traffic
- Scheme One (single-lane road without passing) is effective in extremely heavy traffic
- Scheme Three (multi-lane road under the keep-right-except-to-pass rule) is effective in light traffic

5 Discussions based on the Driving-On-The-Left Rule

It is reported that there are 34% of the world's countries where driving automobiles on the left is the norm and 28% of the world's passable roads are the left-hand traffic [6]. In this respect, it is still very important to make appropriate rules of the road for left-hand traffic and we will discuss them based on our previous study.

According to the manufacturing rules of international auto makers, the position of the cockpit must be close to the road centerline in automotive design, so that drivers can check traffic with a wider field of view while moving, especially in the case of passing another vehicle [6]. According to the driving-on-the-left rule and design principles of the cockpit, we can know that it is completely symmetry between the traffic scenarios under the driving-on-the-left rule and the driving-on-the-right rule. That is, there are no additional cases in real scenarios when vehicles are passing another vehicle. For the operation of overtaking rules, see **Figure 19**.

Note: Green circles represent the position of the cockpit

As we previously analyzed, the randomization procession is affected various uncertainties, which will finally affect the randomization deceleration probability. Uncertainties include over-braking, changes in road conditions, psychological factors and delay in accelerating. Thus, we can find out significant differences in the respect of the above-mentioned factors between left-hand traffic and right-hand traffic. Then we can utilize Analytic Hierarchy Process (AHP) to figure out how left-hand traffic and right-hand traffic differently affect the randomization deceleration probability, which will be expressed by an array of weights.

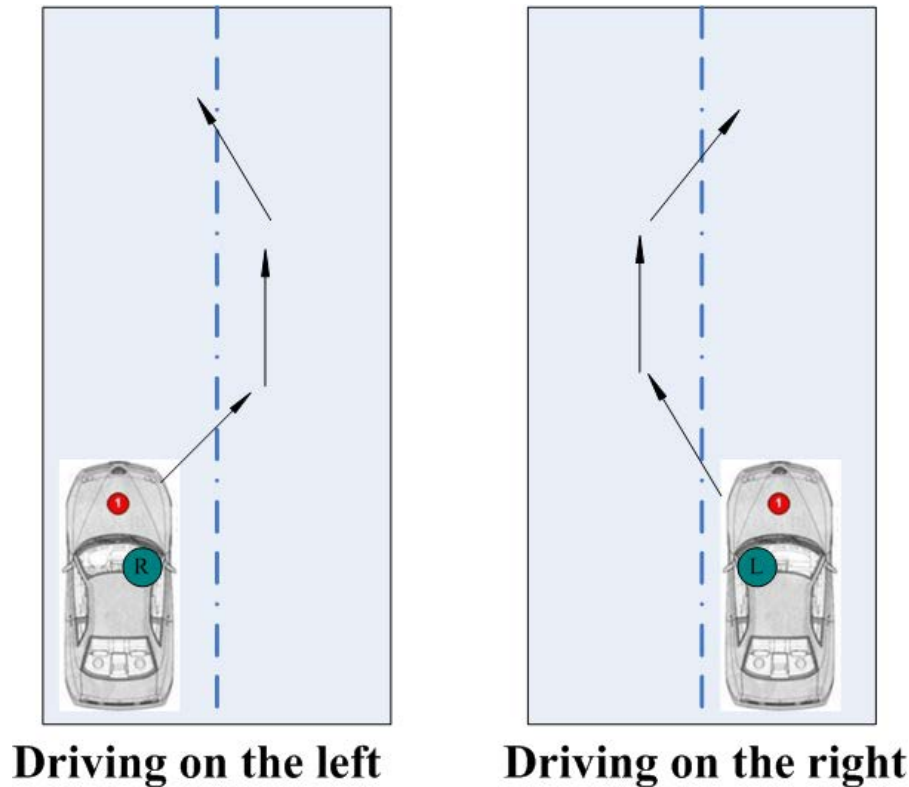


Figure 19 .operation of overtaking rules

5.1 Determination of Weights

In the case that road conditions are the same, the differences between left-hand traffic and right-hand traffic mainly lie in the change of people's sensory systems on account of the differences of driving directions. Relevant information shows that driving habits, people's instinct of avoiding risks [7], benefits for right-handed individuals [8] and benefits for checking traffic [9] are the significant factors affecting people's sensory systems. Next, we utilize Analytic Hierarchy Process (AHP) to figure out how left-hand traffic and right-hand traffic differently affect the randomization deceleration probability, which will be expressed by an array of weights.

1. **Building the Hierarchic Model** Based on the principle of hierarchical analysis and our previous analysis, we develop the following hierarchical structures:

At the top end is the destination layer: identifying speed of people's sensory systems. The second layer is the criterion layer, consisting of four aspects: driving habits, people's instinct of avoiding risks, benefits for right-handed individuals and benefits for checking traffic. The third layer is the project layer, consisting of two approaches: right-hand traffic and left-hand traffic. See the hierarchy diagram in **Figure 20**.

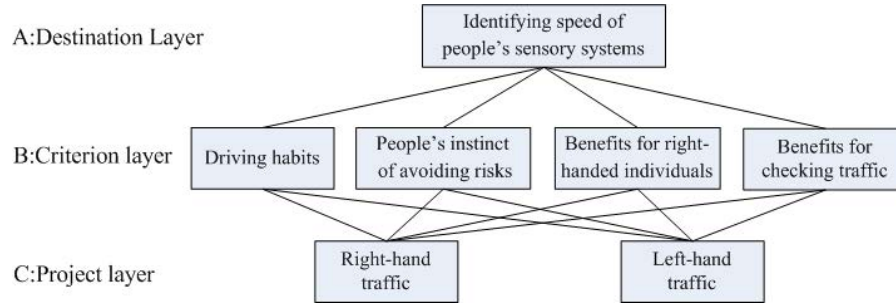


Figure 20 .Hierarchy diagram

2. Establishing Judgment Matrixes

- (a) The judgment matrix of factors affecting identifying speed of people's sensory systems:

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix}$$

According to the importance of each specific factor, the judgment matrix can be set as:

$$A = \begin{bmatrix} 1 & 1/2 & 1/5 & 1/5 \\ 2 & 1 & 3/5 & 2/5 \\ 5 & 5/3 & 1 & 4/5 \\ 5 & 5/2 & 5/4 & 1 \end{bmatrix}$$

- (b) The judgment matrix of factors affecting the criterion layer:

$$B_k = \begin{bmatrix} b_{11}^{(k)} & b_{12}^{(k)} \\ b_{21}^{(k)} & b_{22}^{(k)} \end{bmatrix}, k = 1, 2, 3, 4$$

Where B_1, B_2, B_3, B_4 refer to the judgment matrix of driving habits, people's instinct of avoiding risks, benefits for right-handed individuals and benefits for checking traffic, relatively.

Most countries of the world adopt the driving-on-the-right rule, so drivers who are used to drive on the right don't need to adapt to the new traffic rule in most countries of the world. This is an advantage of right-hand traffic. Thus, the judgment matrix B_1 can be set as:

$$B_1 = \begin{bmatrix} 1 & 3 \\ 1/3 & 1 \end{bmatrix}$$

Human beings have the instinct of avoiding risks: people will instinctively lean to the left or turn to the left, when they are faced with the

danger ahead and move fast at the same time. Thus, if a driver suddenly find himself in danger, which comes from the front, he will be more likely to turn left. Moreover, it will take less time for him to turn left. This is an advantage of left-hand traffic. Therefore, the judgment matrix B_2 can be set as:

$$B_2 = \begin{bmatrix} 1 & 2/3 \\ 3/2 & 1 \end{bmatrix}$$

It is another advantage of right-hand traffic that drivers can use his left hand to keep control of the steering wheel and use his right hand to finish the gear shifting and the operation of dashboard in the meantime. This is beneficial to right-handed drives. Thus, the judgment matrix B_3 can be set as:

$$B_3 = \begin{bmatrix} 1 & 3 \\ 1/3 & 1 \end{bmatrix}$$

Right eyes are always taken as dominant eyes in people's visual system. Consequently, if the driver's seat is placed on the right, it will be easier for the driver to check traffic. Thus, the judgment matrix can be set as:

$$B_4 = \begin{bmatrix} 1 & 1/2 \\ 2 & 1 \end{bmatrix}$$

3. **Consistency Check and Weight Calculation** After simple calculation, we obtain the consistency indexes, which are listed in Table 1.

Table 5 Concrete values of minimum and maximum speed limit

Consistency index	λ_A	CI	CR
Value of the consistency index	4.0159	0.0053	0.0060

We can see in Table 1 that $CR = 0.0006 < 0.1$, so the the judgment matrix A has passed the consistency check. Moreover, a second order matrix is definitely a consistent judgment matrix. Therefore, we don't need to check the consistency of B_1, B_2, B_3, B_4 .

After calculation, we have

$$W_A = (0.07840.17340.33540.4127)^T$$

and

$$W_B = \begin{bmatrix} 0.7597 & 0.4006 & 0.7597 & 0.3333 \\ 0.2403 & 0.5994 & 0.2403 & 0.6667 \end{bmatrix}$$

Thus we get

$$W = W_B \cdot W_A = (0.52, 0.48)$$

Thus the weighted values of right-hand traffic and left-hand traffic are 0.52 and 0.48, relatively.

5.2 How to change

As we previously analyzed, the randomization deceleration probability is caused by various uncertainties, such as drivers' psychological and physiological factors. The randomization process can weaken the drivers' ability to respond to road conditions, thus affecting the deceleration process.

It is completely symmetry between the driving-on-the-left rule and the driving-on-the-right rule in terms of traffic scenarios. Therefore, we don't need to drastically change our solution. We just need to change the randomization deceleration probability slightly.

According to our previous analysis, right-hand traffic has a higher identifying speed of people's sensory system, which means a higher security when overtaking and thus a smaller randomization deceleration probability. We assume that the randomization deceleration probability is inversely proportional to identifying speed of people's sensory system. Then according to the weighted values of right-hand traffic and left-hand traffic, we obtain that the randomization deceleration probability is 0.43. in left-hand traffic when the randomization deceleration probability is 0.4 in right-hand traffic.

6 Discussions based on the Intelligent System

If vehicle transportation is completely under the control of the intelligent system, then we don't have to consider human factors. With the implementation of intelligent system, human judgments will no longer work. There exists no random event at all, including the randomization process. We assume that all vehicles drive at the maximum speed limit and the entry speed of each vehicle is . Therefore, the maximum traffic flow that a certain lane can afford is

$$\frac{V_{max}}{2L}$$

The vehicle density is

$$\frac{1}{V_{max}}$$

The safety of vehicle will not change along with the change of maximum speed limit under the intelligent system. We consider the case of no maximum speed limit, so vehicles will accelerate all the time but not with a constant acceleration.

We assume that the resistance of the vehicle only derives from the friction drag of the road.

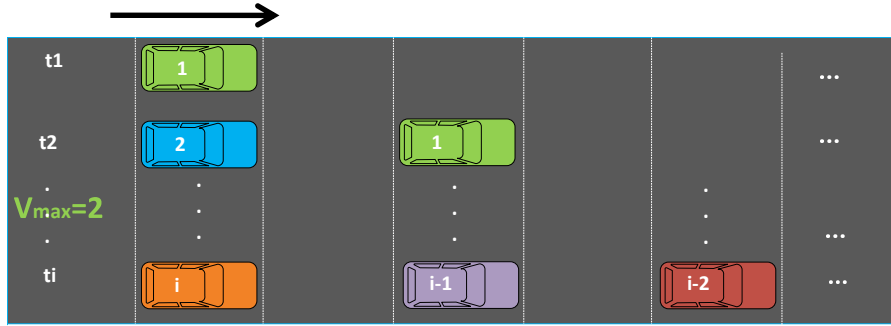


Figure 21 .vehicle transportation under the control of the intelligent system

The maximum acceleration of the i^{th} vehicle at the time of t is

$$a = (\frac{p_i}{v_{it}} - \mu m_i g) / m_i$$

Where

- p_1 is the biggest power of the i^{th} vehicle
- μ is the friction coefficient on road surface
- v_{it} is the speed of the i^{th} vehicle at the moment of t
- m_i is the mass of the vehicle

The minimum acceleration of the i^{th} vehicle at the time of t is

$$a' = (f_i + \mu m_i g) / m_i$$

Where f_1 is the braking resistance of the vehicle.

Then the safety distance between a certain vehicle and the vehicle ahead of it can be expressed as:

1. If $v_{it} < v_{(i+1)t}$ and $a'_i > a'_{i+1}$, then $s = 0$
2. If $v_{it} < v_{(i+1)t}$ and $a'_i < a'_{i+1}$, then $s = \frac{v_{it}^2}{2a'_i}$
3. If $v_{it} > v_{(i+1)t}$ and $a'_i > a'_{i+1}$, then $s = v_{it} \cdot \left(\frac{v_{it} - v_{(i+1)t}}{a'_i - a'_{i+1}} \right) + \frac{1}{2} a'_i \left(\frac{v_{it} - v_{(i+1)t}}{a'_i - a'_{i+1}} \right)^2$
4. If $v_{it} < v_{(i+1)t}$ and $a'_i > a'_{i+1}$, then $s = \frac{v_{it}^2}{2a'_i}$

Since vehicles can reach the maximum acceleration and the minimum safety distance in this model, the traffic flow will reach its maximum under the intelligent system.

7 Strengths, Weaknesses, and Extensions

With the large number of complexities involved in the vehicle transportation system, we begin by analyzing the problem in small parts. Therefore, the model we formulate is progress and easy to understand. As mentioned above, these models taken together have the advantage of simplifying and discretizing the complex continuous traffic flow, so that we successfully simulate the traffic system and obtain the characteristics of traffic flow. However, our model focuses more on the input, output and evolutionary rules, so its reliance on reality is not that high, thus resulting in a certain degree of error.

Our previous study shows that the keep-right-except-to-pass rule is only effective in light traffic. Therefore, we put forward a new rule: the right-most lane is special for low-speed vehicles. Vehicles that drive over a certain speed cannot drive in the right-most lane. In particular, drivers can drive under without rules in rush hours.

References

- [1] "Drivers not abiding by the 'keep right, pass left' rule may be in for some big fines: Joan Quigley." Joan Quigley. The Jersey Journal. Web. 7 Feb. 2014 <http://www.nj.com/hudson/voices/index.ssf/2013/07/life_in_the_fast_lane_may_get.html>
- [2] "How best to educate the public as to the importance of the Slower Traffic Keep Right laws in respect to public safety?" Jesse Breedlove. Web. 7 Feb. 2014 <http://www.ted.com/conversations/14800/how_best_to_educate_the_public.html>
- [3] "KEEP RIGHT Except to Pass". Web. 9 Feb. 2014 <<http://keep.right.except.to.pass.fuuck.ca/>>
- [4] Cleveland D E. Capalle D G. Queuing Theory Approaches: an Introduction to Traffic Flow Theory [R]. Washington D.C.: Highway Research Board. 1964:49-98
- [5] Nagel, KaiWolf, Dietrich EWagner, PeterSimon, Patrice. Two-lane traffic rules for cellular automata: A systematic approach [J]. Physical Review. E.1997(No.2).
- [6] "Right- and left-hand traffic". Wikipedia. Web. 10 Feb. 2014 <http://en.wikipedia.org/wiki/Right-_and_left-hand_traffic>
- [7] "Right- and left-hand traffic". Baidupedia. Web. 10 Feb. 2014 <http://baike.baidu.com/link?url=y7Ks6oQV1N8bPjTTvMnDUqEs2rCH_iNnTWliCElP8HRNRseYWOuO5IaAejvf8X2El9g-83dbiJbEMGxaP13zza>

- [8] "Handedness". Wikipedia. Web. 10 Feb. 2014 <<http://en.wikipedia.org/wiki/Handedness>>
- [9] ZHANG Yong. Research on dominant eyes based on 345 cases of young people with normal sight. Journal of Practical Ophthalmology [J]. 1992(No.5).
- [10] X.Q.Yang, Y.Q.Ma. Car accidents determined by stopped cars and traffic flow. Journal of Physics A. 2002,35(49): 10539-10547

Appendices

Appendix A output data of our algorithm

data of our algorithm,

Table 6 output data of our algorithm

Expect traffic	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
Actual flow(vah/s)(1)	0.123	0.24	0.36	0.43	0.45	0.42	0.41	0.40	0.41	0.42
v(km/h)(1)	118.0112738	120.23	118.65	92.65	44.20	36.30	33.66	35.28	39.41	44.30
Density(vah/km)(1)	3.09	5.97	9.07	14.17	28.86	33.82	36.12	34.72	31.50	28.781
Security (1)	0.5467	0.8177	0.6258	0.3801	0.1751	0.0864	0.0168	0.0084	0.0130	0.0092
Actual flow(vah/s)(2)	0.11	0.23	0.34	0.47	0.60	0.72	0.84	0.91	0.96	1.03
v(km/h)(2)	116.28	123.24	124.88	125.36	124.89	123.22	118.51	114.87	113.68	109.15
Density(vah/km)(2)	1.36	2.916	4.46	6.126	8.116	9.966	12.57	14.18	14.9	16.31
Security(2)	0.9975	0.9973	0.9960	0.9942	0.9919	0.9918	0.9912	0.9909	0.9911	0.9920
Actual flow(vah/s)(3)	0.12	0.24	0.35	0.45	0.54	0.55	0.53	0.56	0.50	0.48
v(km/h)(3)	123.06	113.45	119.78	122.51	122.31	122.53	121.72	121.93	121.88	122.38
Density(vah/km)(3)	1.90	4.26	6.21	8.40	10.19	10.27	10.64	10.65	10.51	10.24
Security(3)	0.9975	0.9973	0.9960	0.9942	0.9919	0.9918	0.9912	0.9909	0.9911	0.9920
Security(3)	0.9973	0.9975	0.9956	0.9938	0.9919	0.9909	0.9914	0.9905	0.9919	0.9923
Actual flow(vah/s)(3)	0.12	0.24	0.37	0.44	0.51	0.54	0.53	0.48	0.45	0.42
v(km/h)(3)	118.09	133.75	142.35	144.41	143.48	142.66	141.88	142.02	142.28	143.02
Density(vah/km)(3)	1.78	3.60	5.57	7.44	8.78	9.08	9.26	9.33	8.66	8.46
Security	0.9984	0.9977	0.9967	0.9957	0.9952	0.9949	0.9945	0.9944	0.9948	0.9944
Actual flow(vah/s)(3)	0.11	0.25	0.35	0.47	0.54	0.62	0.60	0.63	0.58	0.56
v(km/h)(3)	121.45	113.05	120.24	124.21	124.00	123.08	120.90	120.00	121.43	120.46
Density(vah/km)(3)	1.84	4.24	6.14	8.98	10.59	11.28	12.19	12.57	12.06	12.466

Appendix B *two-laneCAmodelbasedonright-handtraffic* MATLAB finish

bordercontrol

```
function [new_matrix_cells,new_v,Traffic]=...
border_control(matrix_cells,a,b,v,vmax,Traffic,q)
```

```

%Boundary conditions, the opening border, vehicle access control

%Outlet boundary, if the first car in the road boundary, then the
%probability of 0.9 leave
n=length(matrix_cells);
if a==n
    rand('state',sum(100*clock)*rand(1));%Definition of random seed
    p_1=rand(1);%-Random probability
    if p_1<=1
        %If the random probability is less than 0.9, the vehicle
        %left the road, otherwise not to leave

        matrix_cells(n)=0;
        v(n)=0;
    end
end

%Inlet boundary, the average number of vehicles arrive is q veh/s
if b>vmax
    rand('state',sum(100*clock)*rand(1));
    p_2=rand(1);
    if p_2<=q
        Traffic=Traffic+1;
        m=min(b-vmax,vmax);
        matrix_cells(m)=1;
        v(m)=m;
    end
end
new_matrix_cells=matrix_cells;
new_v=v;

```

LeftToRight

```

function [isLToR]=LeftToRight(cells_right, v, currPos, vmax, n)
%Key levels based on the rules of right way to the right way of lane changing rules
if currPos<1 || currPos==n
    isLToR =0;
    return;
end

k2=searchfrontcar(currPos,cells_right);
%Search the first non-empty cellular location in front of left cell
if k2==0;%Determine the number of empty cells between the forward car
    d_front=n-currPos;
    %d The space between forward cars
else d_front=k2-currPos-1;
end

k3=searchbackcar(currPos,cells_right);
% Search the first non-empty cellular location behind the left cell
if k3==0;
    % Determine the number of empty cellular between later cars
    d_back=n-currPos;%d The later car with the current spacing

```

```

else d_back=currPos-k3-1;
end

if d_front>min(v(currPos)+1,vmax) && d_back>vmax
    isLToR=1;
else isLToR=0;
end

```

RightToLeft

```

function [isRToL]=RightToLeft(cells_right,cells_left, v, currPos, vmax, n)
%Key levels based on the rules of right way to the right way of lane changing rules
if currPos<1 || currPos==n
    isRToL =0;
    return;
end
k1=searchfrontcar(currPos,cells_right);
%search the first non-empty cellular location
if k1==0;%Determine the number of empty cells between the forward car
    d1=n-currPos;% d 1 The space between forward cars
else d1=k1-currPos-1;
end

k2=searchfrontcar(currPos,cells_left);
%search the first non-empty cellular location in front of the left cell
if k2==0;
%Determine the number of empty cellular between later cars
    d_front=n-currPos;
%The forward car with the current spacing
else d_front=k2-currPos-1;
end

k3=searchbackcar(currPos,cells_left);
%Search the first non-empty cellular location behind the left cell
if k3==0;
%Determine the number of empty cellular between later cars
    d_back=n-currPos;%he later car with the current spacing
else d_back=currPos-k3-1;
end

P=0.6;%The probability of lane changing
if d1<min(v(currPos)+1,vmax) && d_front>d1 && d_back>vmax && rand()<P
    isRToL=1;
else isRToL=0;
end

```

randslow

```

function [new_v]=randslow(v, vmin)
p=0.4;%Probability of slowing down
rand('state',sum(100*clock)*rand(1));%Definition of random seed
p_rand=rand;%Random probability
%In the lane vmin=0
if p_rand<=p && v>vmin

```

```
    v=max(v-1,0);  
end  
new_v=v;
```

leadcarupdate

```
function [new_matrix_cells,new_v]=leadcarupdate(matrix_cells,v)  
  
%The update the rules of the first car  
n=length(matrix_cells);  
if v(n)~=0  
    matrix_cells(n)=0;  
    v(n)=0;  
end  
new_matrix_cells=matrix_cells;  
new_v=v;
```

roadstart

```
function [matrix_cells_start]=roadstart(matrix_cells,n)  
  
%Initialization state road vehicles, random cellular matrix is 0 or 1,  
%matrix_cells is initial matrix, n is the initial number of vehicles  
k=length(matrix_cells);  
z=round(k*rand(1,n));  
for i=1:n  
    j=z(i);  
    if j==0  
        matrix_cells(1)=0;  
    else  
        matrix_cells(j)=1;  
    end  
end  
matrix_cells_start=matrix_cells;
```

searchfrontcar

```
function [location_frontcar]=searchfrontcar(current_location,matrix_cells)  
i=length(matrix_cells);  
if current_location==i  
    location_frontcar=0;  
else  
    for j=current_location+1:i  
        if matrix_cells(j)~=0  
            location_frontcar=j;  
            break;  
        else  
            location_frontcar=0;  
        end  
    end  
end
```

searchlastcar

```

function [location_lastcar]=searchlastcar(matrix_cells)
%Search the last car Location
for i=1:length(matrix_cells)
    if matrix_cells(i)~=0
        location_lastcar=i;
        break;
    else
        %If no cars on the road, set the number of empty cellular be the road length
        location_lastcar=length(matrix_cells);
    end
end

```

searchleadcar

```

function [location_leadcar]=searchleadcar(matrix_cells)
i=length(matrix_cells);
for j=1:i
    if matrix_cells(i-j+1)~=0
        location_leadcar=i-j+1;
        break;
    else
        location_leadcar=0;
    end
end

```

speedstart

```

function [v_matixcells]=speedstart(matrix_cells,vmax)

%The initial installation of road vehicle speed initialization state
v_matixcells=zeros(1,length(matrix_cells));
for i=1:length(matrix_cells)
    if matrix_cells(i)~=0
        v_matixcells(i)=round(vmax*rand(1));
    end
end

```

Main

%This is two-lane model, we let the maximum speed be 5 cells, the Open Boundary's %Co
 %entry probability is $q=0.1:0.1:1$, the leave probability is 0.9, acceleration is 1 %c
 %is 1 cell, Random probability of slowing down is 0.4.

```

clc
clear

```

```

%CA setup
step = 3600;

```

```

n=1000;%Data initialization
z=zeros(1,n);%The number of cells
z_right=roadstart(z,150);%

```

```

Initialization right road condition, road section randomly distributed 150 cars
z_left=roadstart(z,3);
%Initialization left road condition, road section randomly distributed 3 cars
cells_right=z_right;
cells_left=z_left;
vmax=5;%maximum speed
vmin=0;%minimum speed
v_right=speedstart(cells_right,vmax);%Speed initialization
v_left=speedstart(z_left,vmax);
x=1;%Record the speed and vehicle position
memor_cells_right=zeros(step,n);
memor_v_right=zeros(step,n);
memor_cells_left=zeros(step,n);
memor_v_left=zeros(step,n);
mum=1;
for p=0.1:0.1:1
    Traffic=0;%The actual number into the vehicle.
    Adn=0;%The number of actual overtaking.
for x=1:step
        %Boundary condition processing, search at the end and the first car,
        % control access, use the open boundary conditions
        a=searchleadcar(cells_right);
        b=searchlastcar(cells_right);
        [cells_right,v_right,Traffic]=border_control...
(cells_right,a,b,v_right,vmax,Traffic,p);
        i=searchleadcar(cells_right);%Search first bus location
        OldV=0;
%Used to record v_n + 1 previous didn't update in front of the vehicle speed
        NewV=0;%Used to record v_n + 1 last updated speed of the vehicle

        %To the right lane renewal process
        if i-j+1==n
            [z_right,v_right]=leadcarupdate(z_right,v_right);
            continue;
        else
            oldv = v_right(i-j+1);
            %Change lanes, acceleration, deceleration, stochastic moderated
            if cells_right(i-j+1)==0;
%To judge whether the current location is not empty
                continue;
            else
                %Determine whether to change
                [isRToL]=RightToLeft(cells_right,cells_left,v_right,i-j+1,vmax,n);
                if isRToL
                    cells_right(i-j+1)=0;
                    v_left(i-j+1)=v_right(i-j+1);
                    v_right(i-j+1)=0;
                    cells_left(i-j+1)=1;
                    if rand()<0.1
                        Adn=Adn+1;
                    end
                end
                continue;%
            end
        end
    end

```

```

        end
        %Determine whether to change

        v_right(i-j+1)=min(v_right(i-j+1)+1,vmax);%Accelerate
        %Decelerate
        k=searchfrontcar((i-j+1),cells_right);
% Search in front of the first non-empty cellular location
        if k==0;
%Determine the number of empty cells between the forward car
            d=n-(i-j+1);
%The number of empty cells between the forward car and the retrospective car
        else d=k-(i-j+1)-1;
        end
        v_right(i-j+1)=min(v_right(i-j+1),d);
        %=====Decelerate
        %Stochastic moderated
        v_right(i-j+1)=randslow(v_right(i-j+1),vmin);
        new_v=v_right(i-j+1);
%acceleration, deceleration, stochastic moderated

        %Update the vehicle position
        z_right(i-j+1)=0;
        z_right(i-j+1+new_v)=1;
        %Update speed
        v_right(i-j+1)=0;
        v_right(i-j+1+new_v)=new_v;
    end
end

end

%To the left lane renewal process
a=searchleadcar(cells_left);
if a==n
    cells_left(n)=0;
    v_left(n)=0;
end
i=searchleadcar(cells_left);%Search first bus location
OldV=0;
NewV=0;%Used to record v_n + 1 last updated speed of the vehicle
for j=1:i
    if i-j+1==n
        [z_left,v_left]=leadcarupdate(z_left,v_left);
        continue;
    else
        oldv = v_left(i-j+1);
        %Change lanes, acceleration, deceleration
        if cells_left(i-j+1)==0;
%To judge whether the current location is not empty
            continue;
        else
            %Determine whether to change

```

```

[isLToR]=LeftToRight (cells_right,v_left,i-j+1,vmax,n);
if isLToR
    cells_left(i-j+1)=0;
    s=v_left(i-j+1);
    v_left(i-j+1)=0;
    cells_right(i-j+1+s)=1;
    v_right(i-j+1)=min(s+1,vmax);
    if rand()<0.1
        Adn=Adn+1;
    end
    continue;%
end
%Determine whether to change

v_left(i-j+1)=min(v_left(i-j+1)+1,vmax);%Accelerate
%Decelerate
k=searchfrontcar((i-j+1),cells_left);
%Search in front of the first non-empty cellular location
if k==0;
%Determine the number of empty cells between the forward car
    d=n-(i-j+1);
%The number of empty cells between the forward car and the retrospective car
    else d=k-(i-j+1)-1;
    end
    v_left(i-j+1)=min(v_left(i-j+1),d);
    %====Accelerate
    new_v=v_left(i-j+1);
    %====Accelerate,Decelerate
    %Update the vehicle position
    z_left(i-j+1)=0;
    z_left(i-j+1+new_v)=1;
    % Update speed
    v_left(i-j+1)=0;
    v_left(i-j+1+new_v)=new_v;
end
end
end
cells_left=z_left;
memor_cells_left(x,:)=cells_left;%Record the speed and vehicle position
memor_v_left(x,:)=v_left;
cells_right=z_right;
memor_cells_right(x,:)=cells_right;%Record the speed and vehicle position
memor_v_right(x,:)=v_right;
end
CA=sum(memor_cells_right,2);
CB=sum(memor_cells_left,2);
VA=sum(memor_v_right,2);
VA=VA./CA;
VB=sum(memor_v_left,2);
VB=VB./CB;
dele=600;
for i=1:dele%In order to precise calculation, 1000 data before removal

```

```

    CA(1)=[];
    CB(1)=[];
    VA(1)=[];
    VB(1)=[];
end
C=(CA+CB)/2;
V=(VA+VB)/2;
Cvege=sum(C)/(step-dele);
den(mum)=Cvege/n;
avageV(mum)=sum(V)/(step-dele);%The average speed of vehicles to walk
den; %The density of road vehicles
Adcident(mum)=Adn; %The number of actual overtaking
P=0.000001;
%Under the condition of meet the accident, the probability of accidents do occur
P=1-P;
safety(mum)=P^Adn;
Carmum(mum)=Traffic;
mum =mum+1;
end
p=[0.1:0.1:1];
disp('Under the different needs of the number of cars (veh/s)');disp(p);
disp('The average speed of road vehicles');disp(avageV);
disp('The safety of the highway=');disp(safety); %The safety of the highway
disp('The safety of the highway');disp(Carmum);

```

OverTaking

```

function [isOT]=OverTaking(cells1,cells2, v1, currPos, vmax, n)
%No rule of the two-lane lane changing rules

if currPos==0 || currPos==n
    isOT =0;
    return;
end
k1=searchfrontcar(currPos,cells1);
%Search the first non-empty cellular location in front of cell
if k1==0;%Determine the number of empty cells between the forward car
    d1=n-currPos;%d is the space between forward cars
else d1=k1-currPos-1;
end

k2=searchfrontcar(currPos,cells2);
%Search another cell in front of the first non-empty cellular location
if k2==0;%Determine the number of empty cellular between forward cars
    d_other=n-currPos;%d is the ahead car with the current spacing
else d_other=k2-currPos-1;
end

%Search another cellular behind the first non-empty cellular location
k3=0;
j=currPos-1;
while j>0
    if cells2(j)~=0

```

```
        k3=j;
    break;
else
    k3=0;
end
j=j-1;
end
if k3==0;%Determine the number of empty cellular to the later car
    d_back=n-currPos;%The later car with the current spacing
else d_back=currPos-k3-1;
end
P=0.4;%The probability of lane changing
if d1<min(v1(currPos),vmax) && d_other>d1 && d_back>vmax
    if rand()<P
        isOT=1;
    else isOT=0;
    end
else isOT=0;
end
end
```
