# A Middling-Resolution Model of Urban Traffic Network Based on Road Units

**Nation: People's Republic of China**

**Province: Guangdong**

**School: Guangzhou No.6 Middle School**

**Students: Yichao Li, Jinde Yang, Changren Jian**

**Supervisor: Bin Qu**

# Contents

# Abstract

Traffic system is a complex, large-scale system. In view of the high-cost traditional micro model and unsatisfactory macro model, we set up our own middling-resolution model of urban traffic network.

Based on the traditional cellular automata [1] and queue model [2], we combine the advantages of both into our model, introducing new units in traffic network: "Road Unit" and "Generalized Intersection", so as to simplify a complex, irregular traffic network by using simple bipartite graphs as the structure of “Road Units”, and easily build our model of the transport system based on this structure.

At small scales, state transition of the Road Units is the focus of our research. With reasonable assumptions and inferences, we use differential equations and other mathematical tools to build a set of Road Unit state transition rules.

On larger scales, flow distribution of the Generalized Intersections is central to our work. Based on the design of our innovative driver’s psychology model, we also use mathematical methods to build a detailed flow distribution rule and give some interesting details of the transport system in some special conditions.

Based on the study above, we are able to build a complete traffic network model. To test the consistency of this model, we used a specific example to evaluate the entire model.

Building such a complex model is a difficult and tiring task. However, during our research, we gradually feel the charm of mathematics and traffic. This not only encourages us to stick to our goal and overcome difficulties, but also brings us closer to the gate of math and science.

Nowadays, as the traffic system has gone through great changes in our country, prediction and optimization of the system becomes a very important issue. Limited to time as well as our level, we did not conduct deeper research in this area, but we believe that with this new model, we are also able to contribute our own efforts to traffic system researches.


**Key words：** Urban traffic network; Cellular automaton; Road Unit; Generalized Intersection; State transition; Flow distribution.

# 摘要

交通系统是复杂、庞大的.鉴于传统的微观模型求解的高昂代价和宏观模型不尽人意的效果，我们以中观的分辨率建立了我们自己的城市交通网络模型.

基于传统的元胞自动机模型[1]和队列模式[2]，将两者的优点结合，将交通网络以我们独创的方式元胞化，引入了全新的交通网络组成单位：“道路元”和“广义交叉路口”，并且借此以较为规则的二分图作为道路元的组织方式，将复杂的、不规则的交通网络简化，在其基础上简便地进行交通系统的模拟.

在小尺度上，道路元的状态转移是研究的重点.借助合理的假定和推理，我们用微分方程等数学工具建立了道路元的状态转移规则.

在大尺度上，广义交叉路口的流量分配则是我们工作的核心.在我们设计的独具新意的驾驶员心理模型的基础上，同样利用数学方法建立了详尽的流量分配规则，并给出了交通系统中一些特殊情况的有趣细节.

在以上研究的基础上，我们得以构建出一个完整的交通网络模型.为了检验这个模型的合理性，我们利用具体的例子对整个模型进行了评估.

建立这样一个模型是复杂且困难的。但在我们研究的过程中，我们逐渐感受到了数学和交通理论的魅力。这不仅鼓励我们坚持向前、克服苦难，也让我们离数学与科学的大门更近一步。

在城市交通高度发达的今天，交通系统的预测与优化是一个十分重要的课题.限于水平、时间，我们在这方面没有进行更深入的研究，但是我们相信，通过这个新颖的模型，我们能够给交通系统的研究贡献一份自己的力量.

**关键词：** 城市道路交通网络；元胞；道路元；广义交叉路口；状态转移；流量分配.

# Symbols

| Symbol | Significance | Unit |
|---|---|---|
| $k$ | Traffic flow density | $v \cdot m^{-1}$ |
| $k_j$ | Density when traffic is completely blocked | $v \cdot m^{-1}$ |
| $q$ | Number of vehicles passed through a section of road in a unit time | $v \cdot s^{-1}$ |
| $u$ | Speed | $m \cdot s^{-1}$ |
| $u_s$ | Average speed | $m \cdot s^{-1}$ |
| $u_f$ | Free-flow speed | $m \cdot s^{-1}$ |
| $K_r$ | Average output flow | $v \cdot s^{-1}$ |
| $S_f$ | Number of vehicles of queue part | $v$ |
| $S_q$ | Number of vehicles of queue part | $v$ |
| $O$ | Largest capacity of Road Unit | $v$ |
| $I$ | Largest output of Road Unit | $v$ |
| $L_f$ | Length of free part | $m$ |
| $L_q$ | Length of queue part | $m$ |
| $L$ | Length of Road Units | $m$ |
| $d$ | Distance to the mandatory lane change line | $m$ |
| $t_{unit}$ | Unit time | $s$ |
| $t_f$ | Traffic condition | N/A |
| $Dir_k$ | Lane set of driving direction $k$ | N/A |
| $f(x)$ | Significance of lane change of direction of destination in drivers' eyes who are in that position. | N/A |
| $W_{ijk}$ | Attraction of direction of destination of a single lane | N/A |
| $B_{ik}$ | Conservatism coefficient | N/A |
| $A_{ijk}$ | The unified attraction of a single lane | N/A |
| $y_{ijk}$ | The unified attraction after superposition | N/A |
| $G_{ijk}$ | Unified attraction | N/A |
| $H_{ij}$ | Physical limit coefficient matrix of traffic condition | N/A |
| $C_x$ | Physical limit based on the distance between the current lane and the target lane | N/A |
| $F_{ijk}$ | Driver's Psychological tendencies coefficient(without standardization) | N/A |
| $T_{ijk}$ | Psychological tendency coefficient of drivers after standardization | N/A |
| $Q_{F_{ij}}$ | Volume proportional matrix | $v$ |
| $O_{R_{ik}}$ | Proportion coefficient of instantaneous output volume of the vehicles | N/A |

# 1 The theory basis of the traffic system

## 1.1 Introduction to traffic system and its basic characteristics

Traffic system, as the name suggests, is a system composed of roads (including all facilities), vehicles (including drivers) and their surroundings. The system appears to be simple, but it is actually a huge system consists of many elements together influencing each other. And because of the dynamics and uncertainty of drivers, it's difficult to describe the interaction between vehicles by using simple mathematical model accurately.

In recent years, with the improvement of people's living standards, urban residents showed a rapid increase in vehicle ownership situation. Problems caused by urban traffic are also more serious. In addition to time and economic losses caused by traffic jam, there are air pollution, waste of fossil fuels and other environmental problems. Therefore, the study of the traffic network has practical significance particularly. In this paper, we set up our own middling-resolution model of urban traffic network, which is really an attempt.

From the micro level, traffic network can be abstracted as a collection of roads and vehicles on them. Because of the directionality of traffic, it's natural to consider traffic flow has **non-aftereffect property** [1]: the rear traffic has no basic effect on the judgments of drivers in front .

The limited attention and limited sight in traffic system make drivers can only see traffic situation in a small distance. Meanwhile, the distance is a critical factor for drivers to assess whether the event in front is important. And the degree of attention to the event is monotonically decreasing with distance rapidly. Therefore, traffic flow is **short-sighted**: the distant traffic has little effect on drivers' judgments. Of course, with the development of intelligent traffic systems, driver's ability to handle more traffic information globally will become stronger, which will make the short-sighted nature become weaker.

In real life, we often find that the situation of traffic flow is skipping greatly: for intermittent flow, traffic condition varies greatly due to differences in location. From the macro level, traffic flow can be considered **discontinuous:** traffic situation doesn't change continuously along with location.

In summary, traffic system has three properties: (1) non-aftereffect; (2) short-sighted; (3) discontinuous. Our research is based on these three properties.

## 1.2 Basic theory of traffic system

### 1.2.1 Two natural parameters of traffic flow

Parameters associated with vehicles are needed to measure traffic condition, such as speed, vehicle length, turning radius, acceleration and so on. In our model, vehicle speed is the most important.

**Definition 1:**

Vehicle speed refers to the vehicle's instantaneous rate, and the unit of it is $m \cdot s^{-1}$. It is generally expressed by $u$.

**Definition 2:**

Average speed refers to the average of the instantaneous rate of the vehicle within a region (i.e., "space mean speed" ) . It is generally expressed by $u_s$.

### 1.2.2 Volume and density

**Definition 3:**

Traffic volume (or "flow rate") is the most important parameter in describing the rate of traffic flow. It refers to the number of vehicles passed through a section of road (or a section of lane) in unit time. The unit of it is $v \cdot s^{-1}$. It is generally expressed by $q$.

**Definition 4:**

Traffic flow density refers to the number of vehicles on unit length road(or lane). The unit of it is $\mathrm{v} \cdot \mathrm{m}^{-1}$. It is generally expressed by $k$.

This formula [1] describes the relationship between volume, average speed and density:

$$k = \frac{q}{u_s}$$

### 1.2.3 Relationship between speed and density

Density describes the intensity of road vehicles or the degree of congestion on the road. The smaller the density is, the better the traffic situation will be. So it is easy to find that the density and speed have a strong negative correlation.

A variety of models can be constructed to describe the relationship between speed and density, such as linear model, logarithmic model, exponential model, etc. Among these models, linear model is not only easiest, but also fits the actual situation better. The classical linear speed-density function of B.D.Greenshields is listed as following:

**Equation 1: The classical linear speed-density model of B.D.Greenshields** [1]

$$u = u_f \cdot (1 - \frac{k}{k_j})$$

Wherein, $u_f$ refers to free-flow speed (That is also the vehicle speed when vehicles travel freely without interfering with each other). Assuming the scope of our equation is the urban roads instead of the highway and the road speed limit is reasonable, we can assume that $u_f$ is approximately equal to the road speed limit $u_{max}$. In order to facilitate the presentation, $u_f$ mentioned below refers to the reasonable speed limit. $k_j$ refers to the density when traffic is completely blocked.

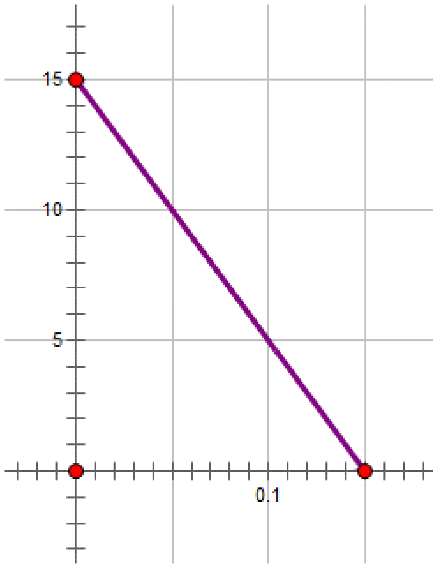


**Figure 1: graph of speed-density function of B.D.Greenshields**

### 1.2.4 Intersections and Mandatory Movement Lane Change

Intersection refers to a junction where streets or roads crosses directly. In fact, area near the intersection is one of the most complex areas between vehicles. This is because the vehicle near the intersection in order to sail in the direction they want to go, it has to make a lot of lane change operations. Wherein, due to limitations of traffic regulations, the vehicle in all directions must complete the target lane change beyond a certain distance before entering the intersection. Such a process is called mandatory lane change. The process in which vehicle enters its own lane because of mandatory lane change is called channelization, shown in Figure 2. In channelized areas, due to the limitations of solid white lines, vehicles cannot change lanes completely.

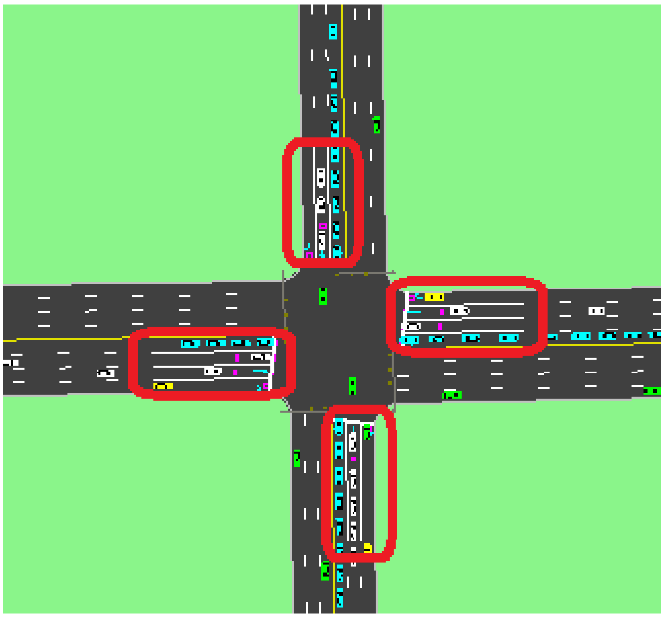

**Figure 2: schematic diagram of road channelization (areas rung by red circle is the channelized area)**

## 1.3 The structure of the model

Based on the structure of cellular automaton, our traffic system model is based on the two components we propose: “**Road Unit**” and “**Generalized Intersection**”. Similar to a cell, "Road Unit" is an integral part of this model, but has a big difference with the traditional cellular automaton model of traffic network in the characteristics, size, and state transition rules and so on.

“Generalized Intersection” is the basic organization of "Road Unit", which is equivalent to the spatial distribution of cellular in cellular automaton model. And its unique "bipartite graph" type of connection makes it different. Based on some fundamental assumptions, these two components constitute our transport system model by some rules and contacts.

### 1.3.1 Four basic assumptions

**Assumption 1:**

If the length and paragraphs selected are appropriate, the traffic condition in an area can be seen as homogeneous.

Considering both the discontinuity and continuity of traffic flow, this assumption is easy to understand.

**Assumption 2:**

Parameters of all vehicles in a traffic network can be replaced by the average of the parameters of these vehicles (e.g., suppose the length of all vehicles is equal to the average length of vehicles).

Obviously, this assumption has an obvious error. For example, simply because of its length, a large truck usually turns and changes lane inefficiently, making jams at the intersection. But under this assumption, this phenomenon cannot be described directly. However, in order not to fall into microscopic simulation, Assumption 2 is needed. Problems such as large trucks which cannot be described by this assumption will be solved in 4.1 with an alternative solution.

**Assumption 3:**

The route of the vehicle will not change since its departure; destination of the vehicle is irrelevant with its starting point.

The first part of this assumption is based on the short-sightedness of drivers, because the drivers, unable to predict traffic conditions in front, are generally not to leave their current route, which is planned in advance. The latter part is a purely conjecture based on experience. Both parts of this assumption can work together to draw a corollary:

**Corollary 1:**

The proportion of vehicles which travel in different directions in an intersection is constant.

**Assumption 4:**

vehicle is a continuous medium. The number of vehicles is not a discrete positive integer, but can be any real number.

Since the model is a microscopic one, Assumption 4 can simplify mathematical analysis without loss of practical meaning. To extract the integer data, we can just use the nearest integer instead.

### 1.3.2 First element of the model: Road Units

Assumption 1 easily reminds us of the classic cellular thought: to a certain extent, roads can be divided into some small parts with fixed position among which the traffic condition is completely homogeneous. However, such an approach does not accurately describe the phenomenon of vehicle queues.

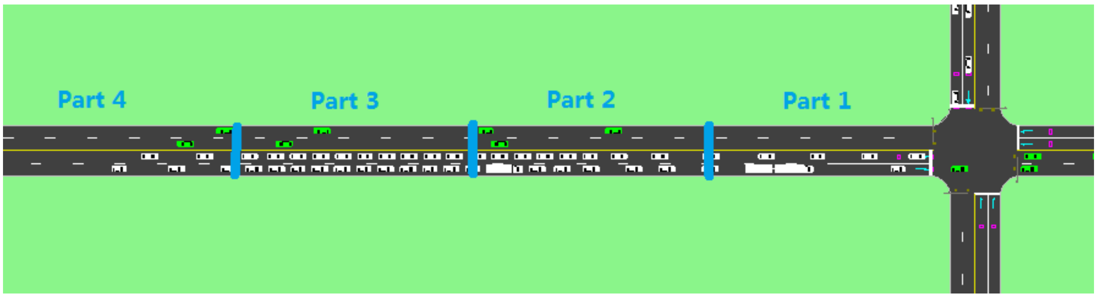


**Figure 3: scenario of a queue, simulated by SimTraffic (length of each piece is about $50m$)**

As Figure 3 showing, among the vehicles eastbound, the distribution of vehicles in Part1 is relatively sparse and intuitively relatively uniform, which satisfies Assumption 1. Part 3 also satisfies the assumption because of its high density. However, traffic conditions of Part 2 and Part 4, the beginning and end of the queue of vehicles, are highly discontinuous. As Part 4, the beginning of it is the end of queue with large density and low speed. On the contrary, the rear is sparse. In particular, the end of a queue is critical in the queue's formation, developing and dissipating. If Part 4 is simply regarded as homogeneous, the simulation may be very inaccurate, since the value of position of the queue's beginning and ending will become quite discrete: only the cutting points of the road can be taken. To improve the accuracy without changing the model, the distance between two cutting points has to be reduced, which will greatly increase the cost of simulation.

The key to solving the problem is to make part of the description of the queue be more precise. Thus we introduce our ideas: Road Unit model.

**Definition 5:**

Road Unit, RU, refers to a section of single-lane road with a fixed length $L$, and have a fixed location on map. It consists of two parts: free part $F$ and queue part $Q$. The two portions have their own length $L_f$ and $L_q$ respectively. Both the length is not fixed (Both are subject to change with simulation), but meet an equation: $L_f + L_q = L$. These two parts are viewed as homogeneous respectively. Among them, the queue part is considered completely blocked, with $k = k_j$, $u = 0$. On the other hand, the vehicle of the free portion can be seen as able to move, $k < k_j$, $u > 0$. Similar to a cell in cellular automata, the state of Road Unit $U$ can be described as: $U = \{L, k_j, L_f, L_q, k, u\}$.

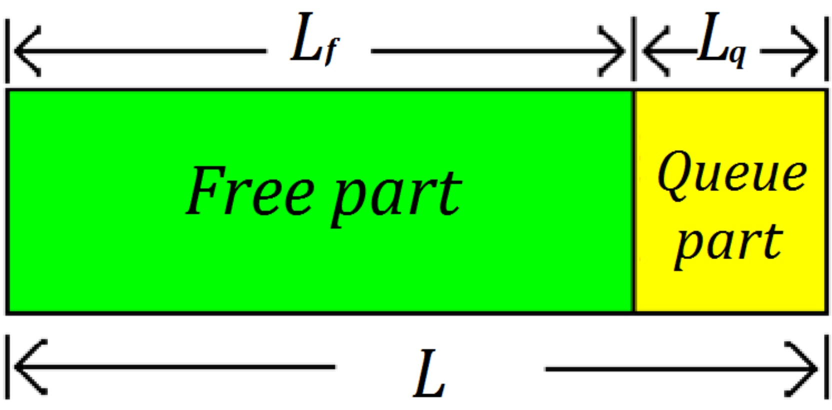


**Figure 4: Schematic diagram of a Road Unit**

Obviously, this model gives a better description of the vehicle queue: the tail of queues can be more accurately described, although it is undeniable that Road Unit model is not essentially improved for the descriptions of the front of queue part (Position value of the forward terminal remains at discrete points). But the characteristics of the front of the queue dissipating make the front border of queues vague, so the error can be accepted.

### 1.3.3 Organization of Road Units in traffic network

**Definition 6:**

Given the number of lanes of a road $n$, we know that: (1) $n$ RUs lies across the road, representing $n$ parallel lanes, which is called a Road Unit Group; (2) several Road Unit Groups are connected with each other longitudinally along the road, which constitutes the traffic flow forward on the road.

In order to describe the process of changing lanes, now introduce Assumption 5 as following.

**Assumption 5:**

The process of changing lanes only occurs where two Road Unit Group connecting with each other.

Although errors exist in the assumption, it is necessary to enable the resolution of the model to subside into a microscopic model (For example, it is almost impossible to describe the interference of changing lanes for traffic flow). We accept this assumption firstly, and we get:

**Definition 6 (continued):**

(3) Between the two adjacent Road Unit Groups, a directive connection is established from

any one of the previous Road Unit to any rear one to describe the process of lane change.

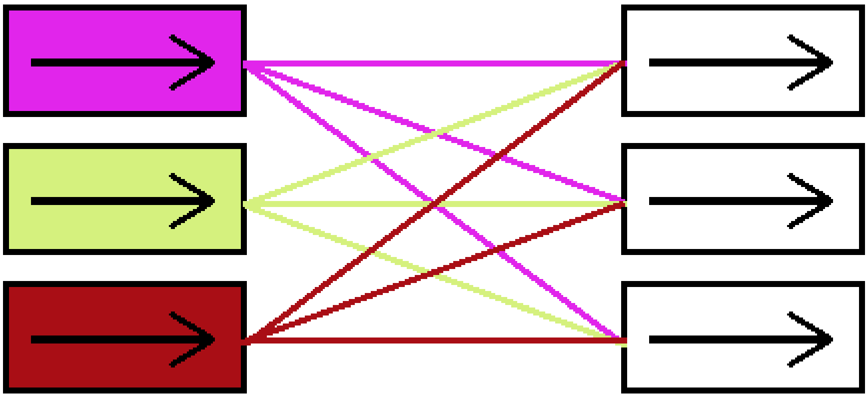

**Figure 5: Organization of two Road Unit Groups**

It is important to note that the organization is equally applicable to actual intersections. The Road Units entering intersections are regarded as the Front Road Unit Group, and the others exiting road intersections are regarded as the Rear Road Unit Group. Organization of such intersections is very similar to the organization of RUs in the middle of a road, only a little more complicated in the conjunctions, shown as Figure 6. Different blocks of color represent different Road Units near the intersection (there are two lanes in each side of the road, so each color-block actually represents two roads units). Red represents the Road Units entering the intersection, while blue represents the Road Units exiting the road intersection. Then the connections between the 16 Road Units can be made in a similar way to Figure 5. This idea leads to the second element of the model.

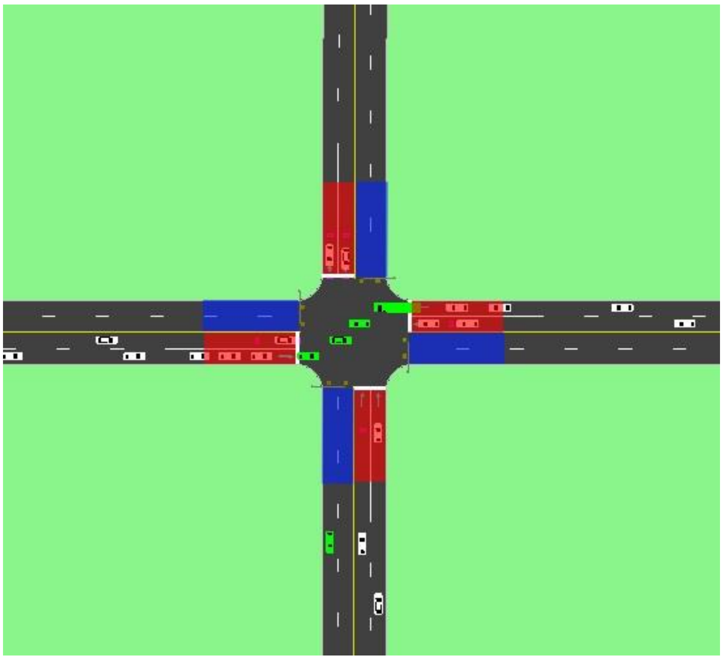

**Figure 6: Organization of Road Units in an actual intersection**

### 1.3.4 Second element of the model: Generalized Intersections

Considering a Road Unit as a node and the connection between Road Units as an edge, the entire traffic network can be viewed as a directed graph. Obviously, such a graph is too large to find any meaningful pattern in it. For this reason, our model focuses on finding special sub-graphs inside this complex graph. According to the similarity between connections of Road Unit Groups and intersections, we find a common model with the method of analogy.

**Definition 7:**

Generalized Intersections, GI, means connection between two adjacent Road Unit Groups. This definition includes the additional definition of actual intersections in the last paragraph of 1.3.3.

It can be seen in Figure 5 that if the front Road Unit Group is considered as the first part, the rear Road Unit Group as the second part, and the connection between the two parts as the direction of traffic flow, such a Generalized Intersection is actually a bipartite graph.

**Definition 7 (continued):**

In a Generalized Intersection, we call the front Road Unit Group “input side” and call the rear Road Unit Group “output side”. So the edges of the bipartite graph exist only between the inputs and outputs sides of the GI. And the necessary and sufficient condition to make two Road Units linked together is that the route from the first Road Unit to the second one is accord with the traffic rules.

Obviously, the following conclusion can be drawn:

**Corollary 2:**

Each Road Unit in the traffic network can only belong to one input side and one output side, and it can belong to only two Generalized Intersections at the same time.

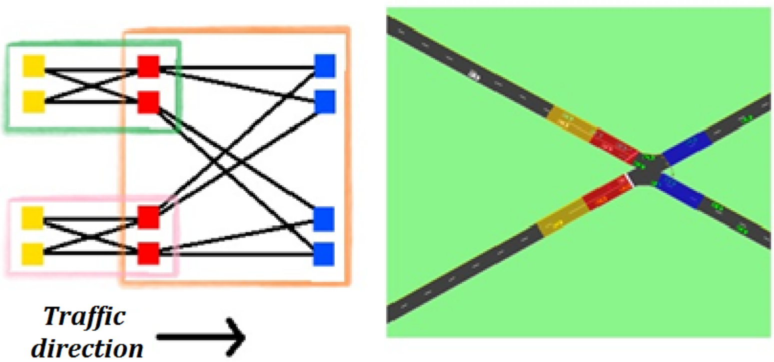


**Figure 7: The Generalized Intersection structure of traffic network (abstract graph and scene graph)**

According to Graph 9, every point is a Road Unit, and every component inside the square frame is a bipartite graph structured Generalized Intersection. The traffic network can be seen as a network made up by multiple bipartite graphs. In this way, the Road Unit model and the Generalized Intersection structure together form the basic structure of our traffic network model.

# 2 State transition of Road Units

Quantitive and precise mathematical calculation play a key role in a model. So is the case with the RU model. This section aims at describing the operation rule of traffic flow from the aspect of RUs with mathematical tools. In this section, we apply the idea of state transition of traffic flow of the cellular automata model and set the rule of state transition between and within one single Road Unit and other different Road Units. We also set up a more comprehensive traffic status evaluation model based on a drivers' psychological character.

In this model, state transition of RUs consists of three parts:

1) Traffic flow inside Road Units: the vehicle preceding process (vehicles go from the free part to the queue part);
2) Traffic flow between two Road Units: Queue dissipation (vehicles go from the queue part to the free part) and Flow Push-in (vehicles go from one free part to another free part);
3) Updating the basic parameters of the Road Units.

All three parts will be analyzed in 2.1, 2.2, 2.3~2.4 respectively.

## 2.1 The vehicle preceding process

This section discusses the vehicle preceding process of free part. In cellular automation model, a car may correspond to one or more cellular. However, our model of Road Units contains two different parts and more vehicles. Therefore, the movement of vehicles inside Road Units needs to be taken into account. The process is defined as a process that the queue part of the Road Unit can't dissipate forward and the vehicles in free part will gradually move into queue part, which makes queue part grow with time and free part became sparse. We build up the relationship between the length and the number of vehicles of the free part and time. Due to the idealization of Road Unit model, it is relatively simple to describe this process with mathematical formulas.

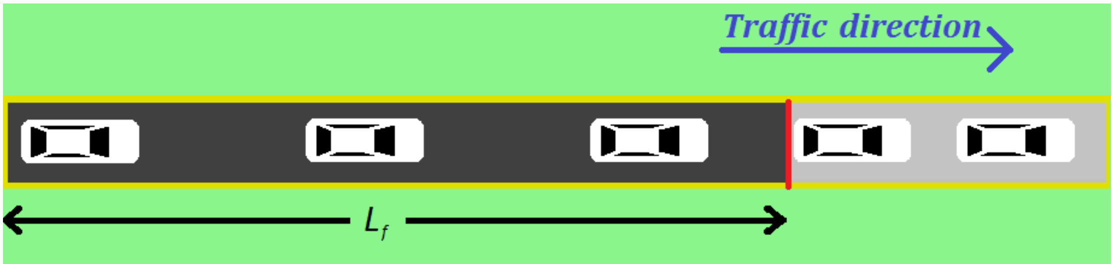


**Figure 8: A scene of vehicle preceding process**

### 2.1.1 Simple differential equation description of vehicle preceding process

The following descriptions are based on Definition 5:

The speed of vehicles drive forward and the current density of free part satisfies Equation 1. Define the current number vehicles in free part of Road Unit as $s$, the length of free part as $L_f$, the current speed as $u$, the density as $k$, speed limit as $u_f$, jam density is $k_j$, then:

$$k = \frac{s}{L_f}$$

$$u = u_f \cdot \left(1 - \frac{k}{k_j}\right)$$

However, density decreases as the vehicles move forward. This, in return, will increase the speed of the vehicles. In fact, it results from the reduction of the number of vehicles of free part. Define the number of vehicles of free part as $s$, the number of vehicles in free part at the next time unit can be calculated as follows:

$$s + \Delta s = s - u \cdot k \cdot \Delta t$$

Obviously, the reduction of vehicles in free part is,

$$\Delta s = -u \cdot k \cdot \Delta t$$

Accordingly, this makes an increase in quantity and length of the vehicle queue part, compared, so the length of the free part decreases. Define the length of free part is $l$. The length changes is:

$$\Delta L_f = \frac{\Delta s}{k_j} = -\frac{u \cdot k \cdot \Delta t}{k_j}$$

Take the expression of $u$, $k$ into the above equation. Then let $\Delta t \to 0$, the equation above can be written in differential form:

**Equation 2: Simple differential equation model of vehicle preceding process**

$$\begin{cases} \dfrac{ds}{dt} = -\dfrac{u_f \cdot s}{L_f}\left(1 - \dfrac{s}{L_f \cdot k_j}\right) \\ \dfrac{dL_f}{dt} = -\dfrac{u_f \cdot s}{k_j \cdot L_f}\left(1 - \dfrac{s}{L_f \cdot k_j}\right) \end{cases}$$

This is the simple differential equation for describing vehicles driving. However, it's too complicated to solve. According to the recursive nature in the process, the solution we use is: firstly, select minor $\Delta t$ and use Pascal language to program for solution (refer to the Appendix for source code). Secondly, select the representative initial conditions (see illustration in the graph below) to recursively calculate the numerical solution. Finally, use Excel to plot the graph:

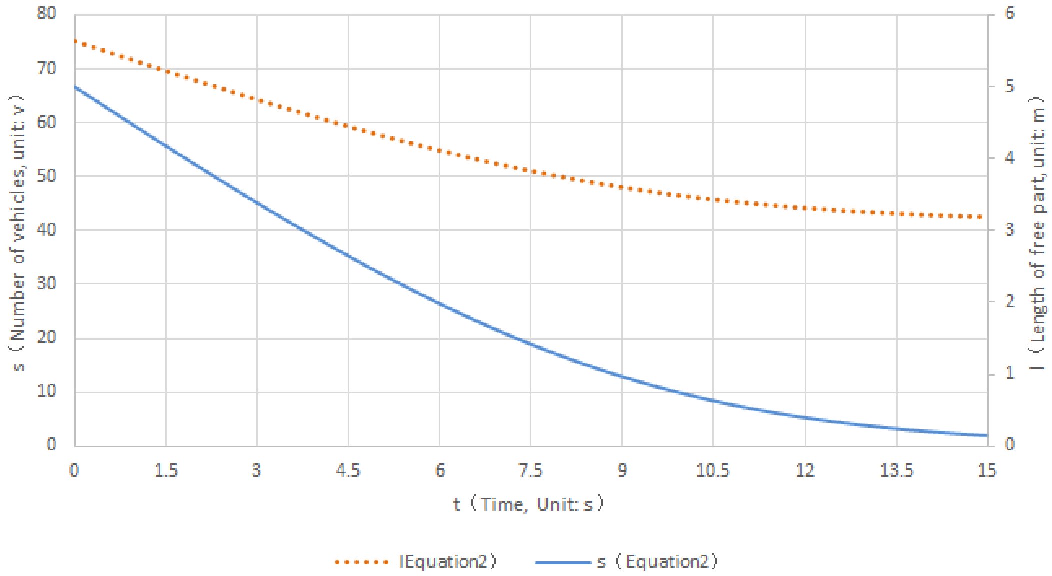


**Figure 9: Graph of Equation 2**
**(Take $\Delta t = 0.01s$, $Time = 15s$, $u_f = 15m \cdot s^{-1}$, $k_j = 0.15v \cdot s^{-1}$,**
**with initial conditions: $L_{f_0} = 75m$, $s_0 = 5v$)**

### 2.1.2 Qualitative analysis of Equation 2

Casting aside the vehicle performance and all the same parameters in Assumption 2 and the homogeneity of the free part in Definition 5, now we discuss the process of vehicles moving forward in reality. In the process of derivation of Equation 2 in 2.1.1, we suppose that there're no vehicles behind the Road Unit as well as complete blockage ahead of the Road Unit. Now, assume that there is an actual segment of road on which there is a car group whose initial density is homogeneous, with no cars behind it and complete blockage ahead.

Since we cancelled the Assumption 2, it's easy to find that the car in the car group moves at different paces. Soon, an obvious distance will form between the fast vehicles and the slow ones because of the different moving speeds of them. If the group is not affected by the block in the front, then the most front point and the rearmost point of the group will display a speed difference, which makes the length of the vehicle group become longer and longer and the density of the group smaller and smaller. From the relational equation in 1.2.2, we get:

$$q = k \cdot u_s$$

In the process of vehicles preceding, we select the section between queue part and free part as the research subject. In the vehicle group, the more rearward the vehicle, the smaller the average speed. Combined with the fact that vehicle group is moving forward, speed $u$ monotonically decreases with respect to time $t$ on this section. According to Definition 3, the number of cars through the section per unit time is called volume. As the density of car group is getting smaller, the density $k$ at the border is monotonically decreasing with respect to time $t$. In summary, volume $q$ through the section in per unit time is monotonically decreasing.

Take the curve of equation 2 into account. The slope of s-t curve satisfies:

$$\frac{ds}{dt} = -q$$

From the curve, we find that the trend of the curve s-t slowly changes from a steep decrease into a gentle decrease, which means the second derivative is a positive number. So Equation 2 concurs with the conclusion, that the flow $q$ of the vehicles passing through the section in a time unit monotonically decreases, which is drawn from qualitative analysis.

### 2.1.3 Improvement of Equation 2: compensation for unequal density

Although the overall trend of Equation 2 is in line with our qualitative analysis, there is an obvious problem in Equation 2 compared with the real condition. Due to the restriction that free part is homogeneous, cars always scatter throughout the whole Road Unit. Therefore, this situation cannot be avoided: There is always some vehicles at the rear end of the Road Unit and they are unable to enter the queue part. All of these are reflected in the curve, that is, $s - t$ curve tends to be infinitely close to 0 but cannot reach 0.

Continue the inference in 2.1.2, let's analyze qualitatively again. Under the condition of homogeneity of the vehicle group, although the length of the group becomes longer, the end of the group can always all enters the queue in limited time, as the speed of any vehicle is always greater than 0. Now we consider such a point of time: the end of the car group is about to enter the queue. At this time, the average density of the free part is very low (because there're few cars in the rear end of free part, which is relatively large). However, those vehicle which are moving freely have almost all gathered in the front of the free part.

If the number of the vehicles in this part is small enough, then the free part can be divided into two parts: one with few vehicles and short length; the other without any vehicles, empty but covers most of the end of the Road Unit. Assume that density, speed and other parameters are all equal. At this time the relationship between $s$ and $t$ is linear. In the vehicle preceding process, the smaller the total density, the more significant the linear relationship, then the Equation 2 is correspondingly more inappropriate.

The figure below is a scene graph made by a microscopic simulation software SimTraffic [6] on the purpose of investigating this phenomenon.

**Figure 10: Virtual scene of vehicle preceding from SimTraffic**

In this figure, length of the road between two intersections is approximately 250m. Take $u_f = 60km \cdot h^{-1} \approx 16.67m \cdot s^{-1}$. The vehicles are made to move freely for some time, so that this section of the road traffic is approximately homogeneous. Then make the two traffic lights of the direction turn red at the same time, making vehicles inside the road unable to go out and the vehicles behind unable to go in. Recording the time needed for this originally well-distributed vehicle flow to completely turn into a queue, the following table is obtained:

| group number | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| vehicle amount: S | 10 | 11 | 10 | 10 | 12 | 11 | 11 | 11 | 11 | 11 |
| QUEUE formation Time: T | 27 | 28 | 29 | 28 | 29 | 27 | 28 | 29 | 27 | 27 |

**Table 1: the number of vehicles and the formation of the queue length data**

Considering the linear relationship mentioned above on some extent, we introduce an item represented such a linear relationship into Equation 2. Since the rate of linear type movement are:

1) Closely related to the initial preceding volume through the section in the front: $q_0 = u_0 \cdot k_0$;

2) The smaller the number of vehicles, the more significant the linear relationship.

Therefore, we add an item to the Equation 2, so that it becomes as follows:

$$\begin{cases} \dfrac{ds}{dt} = -\dfrac{u_f \cdot s}{L_f}\left(1 - \dfrac{s}{L_f \cdot k_j}\right) - x \cdot u_0 \cdot k_0 \cdot \left(1 - \dfrac{s}{s_0}\right) \\ \dfrac{dL_f}{dt} = -\dfrac{u_f \cdot s}{k_j \cdot L_f}\left(1 - \dfrac{s}{L_f \cdot k_j}\right) - \dfrac{x \cdot u_0 \cdot k_0}{k_j} \cdot \left(1 - \dfrac{s}{s_0}\right) \end{cases}$$

Where $u_0 \cdot k_0$ corresponds with 1), and $1 - \frac{s}{s_0}$ corresponds with 2). $x$ is an empirical constant. According to the data from Table 1, the best $x = 0.353$. So we get the differential equations after compensation for linear preceding.

**Equation 3: differential equation model of improved vehicle preceding process**

$$\begin{cases} \dfrac{ds}{dt} = -\dfrac{u_f \cdot s}{L_f}\left(1 - \dfrac{s}{L_f \cdot k_j}\right) - 0.353 \cdot \dfrac{u_f \cdot s_0}{L_{f_0}} \cdot \left(1 - \dfrac{s_0}{k_j \cdot L_{f_0}}\right) \cdot \left(1 - \dfrac{s}{s_0}\right) \\ \dfrac{dL_f}{dt} = -\dfrac{u_f \cdot s}{k_j \cdot L_f}\left(1 - \dfrac{s}{L_f \cdot k_j}\right) - 0.353 \cdot \dfrac{u_f \cdot s_0}{k_j \cdot L_{f_0}} \cdot \left(1 - \dfrac{s_0}{k_j \cdot L_{f_0}}\right) \cdot \left(1 - \dfrac{s}{s_0}\right) \end{cases}$$

Using methods and procedures similar to Equation 2, the following graph can be drawn:

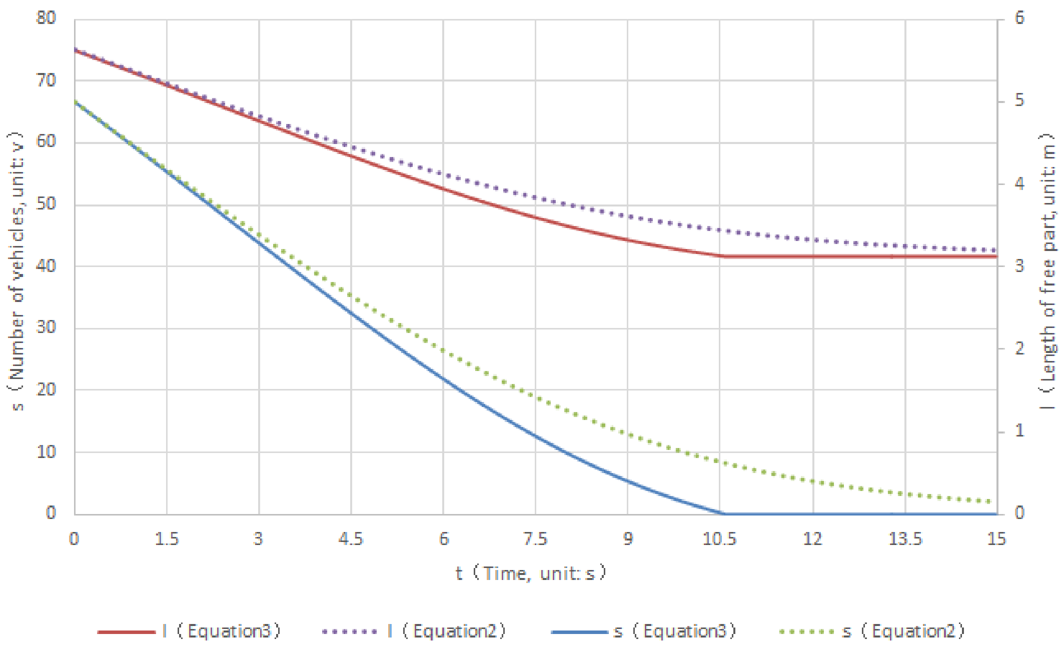


**Figure 11: A graph of Equation 3 and a comparison with Equation 2 (Take $\Delta t = 0.01s$, $time = 15s$, $u_f = 15m \cdot s^{-1}$, $k_j = 0.15v \cdot m^{-1}$, with initial conditions: $L_{f_0} = 75m$, $s_0 = 5v$)**

As can be seen, Equations 3 is more close to actual situation, which reflects the facts that the graph of $s$ is sure to have an intersection to the $T$ axis. Therefore, using Equation 3 to describe the vehicle preceding process is more reasonable.

## 2.2 Queue dissipation and flow push-in

The definition of queue dissipation refers to the process that vehicles go out of a queue and enter the free traveling area. Flow push-in is the state change of free part when vehicles enter it from a queue. In flow push-in, we assume ideally: the queue is long enough and the free traveling area is blocked in the front by a completely restricted section, which does not move. Similar to 2.1, we will introduce a mathematical model of the process.

### 2.2.1 Selection of sections: the moving section and the fixed section

For the process of queue dissipation, we first consider the standard of the so-called dissipation itself. Dissipation means vehicles pass through a section which means the border of a queue. The section between the free part and the queue part described above is taken for granted as being the right choice. But this seemingly natural section is moving: it is constantly moving backward as the vehicles dissipate. So it is called “moving section” of queue dissipation.

For a vehicle in queue, it can break away from the queue only if the vehicle in front of it breaks away from the queue. Therefore, before the very moment that a vehicle leaves the queue, the vehicle's position does not change. So, let $q$ be the volume through the moving section and take traffic direction as positive direction, the section’s displacement $x$ satisfies:

$$x = -\frac{\int_0^t q \cdot dt}{k_j}$$

Let $L$ be the length of the area where vehicles have move away, it satisfies:

$$L = L_0 + \frac{\int_0^t q \cdot dt}{k_j}$$

Where $L_0$ is the initial length of the area.

On the other hand, stop lines in signalized intersections are also a very natural selection of section. To generalize it, we set the head of the vehicle which is in the most front before a queue starts dissipating as a practicable section. The biggest difference between this section and a moving section is that this section is fixed. Hence, we call it "fixed section".

Compared to moving section, there is a big advantage of fixed section, that is, it is more convenient for vehicle counting and analysis. Next, we first study the fixed section dissipation flow by an experiment. And then based on it, we study the related properties of moving section.

### 2.2.2 Fixed section's dissipation flow-limit speed relationship

To study the dissipation through fixed section, we used SimTraffic to get the data about volume $q$.

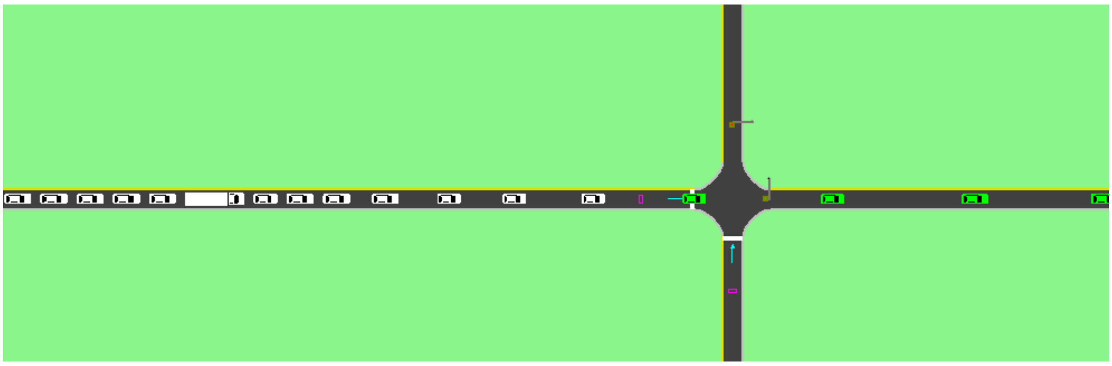

**Figure 12: A virtual intersection for studying queue dissipation in SimTraffic**

As shown in Figure 12, we inject the saturated flow to intersection of east driveway (the value calculated in Synchro is $1863v \cdot h^{-1}$). Then give the intersection a long red signal, which makes the queue long enough. When the light turns green, record the number of vehicles which has left the queue at different time point, and repeatedly do the experiment to get average values. We recorded the number of dissipated vehicles at different speed limits with Excel (see Appendix for the detailed data):

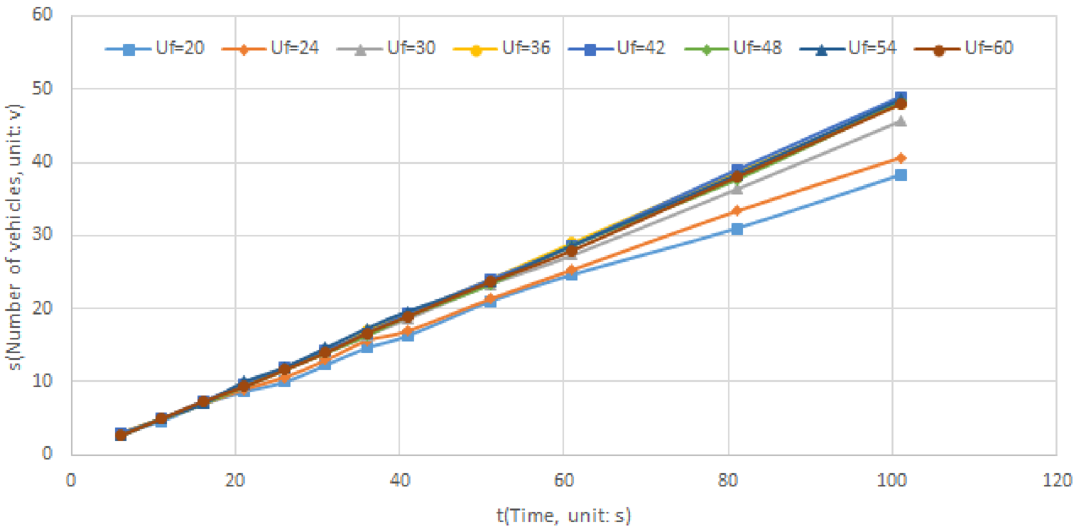


**Figure 13: The graph of the number of dissipated vehicles at different time with different speed limit (Note: the unit of limit speed is $km \cdot h^{-1}$)**

Obviously, the total number of dissipated vehicles present very good linear correlation with time. That is to say at the same speed queue dissipation volume (i.e., the slope) remains almost constant, almost completely unaffected by the change of time. On the other hand, it means that the traffic condition at the queue exit hardly affects queue dissipation flow rate. Linear-regression analysis is conducted on these 8 curves and their slopes are displayed as follows:

| $U_f(km \cdot h^{-1})$ | 20 | 24 | 30 | 36 | 42 | 48 | 54 | 60 |
|---|---|---|---|---|---|---|---|---|
| **Q(slope)** | 0.3769 | 0.4006 | 0.4504 | 0.4795 | 0.4858 | 0.4784 | 0.4810 | 0.4756 |
| **B(intercept)** | 0.8168 | 0.6579 | 0.0752 | -0.3752 | -0.5116 | -0.6333 | -0.3487 | -0.4902 |

**Table 2: The table of linear regression varying about $u_f$**

According to Table 2, a graph about $q$ and $u_f$ was made by Excel:

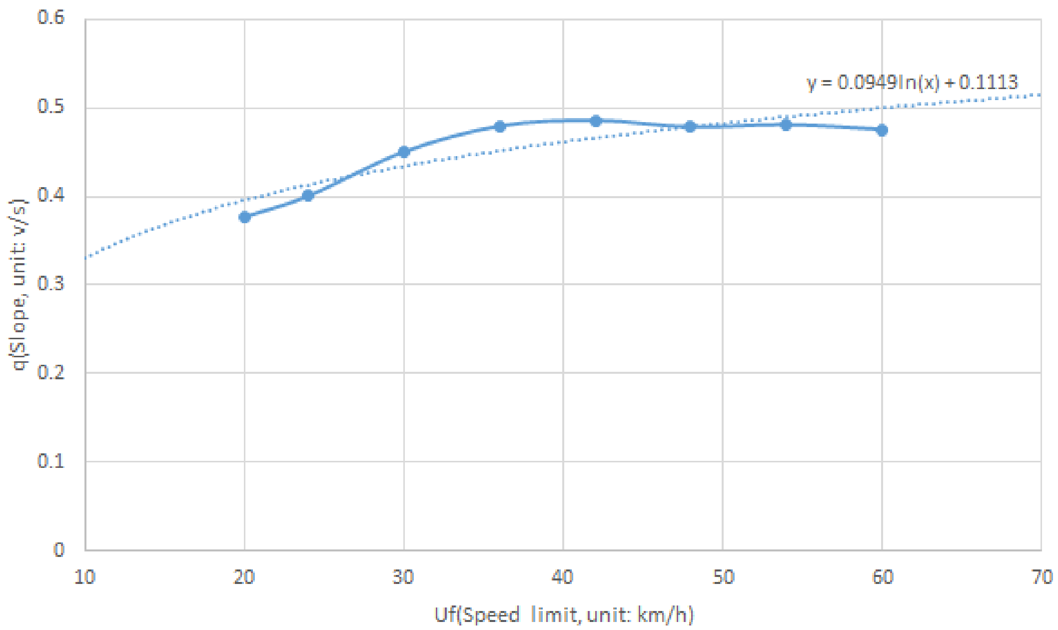


**Figure 14: slope-speed graph based on Table 2**
**(showing logarithmic regression curve, $R^2$ = 0.7866)**

As can be seen, $q$ approximately obeys the logarithmic relationship: $q = 0.0949 \cdot ln(u_f) + 0.1113$, where $u_f \in [20,60]$. Considering that in city traffic network, the range of $u_f$ is basically in $[20,60] \cdot km \cdot h^{-1}$, and it's very unlikely to have a higher or lower speed limit. So, the situations where $u_f$ is out of $[20,60] \cdot km \cdot h^{-1}$ can be simply ignored.

In conclusion, after converting the speed unit to $m \cdot s^{-1}$, we can describe the relationship between $q$ and $u_f$ as:

**Equation 4: The fixed section dissipation volume and speed limit relationship**

$$q = 0.0949 \cdot ln(u_f) + 0.2329,\ u_f \in [\frac{50}{9}, \frac{50}{3}] \cdot m \cdot s^{-1}$$

### 2.2.3 Description of fixed section dissipation

We've got a logarithmic dissipation volume model. Based on the experiments and analysis above, the description of fixed section dissipation will be conducted in this section.

Considering the expression of $L$ in 2.2.1:

$$L = L_0 + \frac{\int_0^t q \cdot dt}{k_j}$$

With derivation of Equation 4, we get:

$$\frac{dL}{dt} = \frac{0.0949 \cdot ln(u_f) + 0.2329}{k_j},\ u_f \in [\frac{50}{9}, \frac{50}{3}] \cdot m \cdot s^{-1}$$

This means that the derivative of $L$ is constant. With initial condition: $L = L_0$, it's easy to get:

**Equation 5:**

$$L = \frac{0.0949 \cdot ln(u_f) + 0.2329}{k_j} \cdot t + L_0,\ u_f \in [\frac{50}{9}, \frac{50}{3}] \cdot m \cdot s^{-1}$$

Similarly, considering the number of dissipated vehicles $s$, apparently:

$$s = \int_0^t q \cdot dt$$

Whose solution is:

**Equation 6: The linear model of fixed section queue dissipation volume**

$$s = [0.0949 \cdot ln(u_f) + 0.2329] \cdot t,\ u_f \in [\frac{50}{9}, \frac{50}{3}] \cdot m \cdot s^{-1}$$

From the equation above, if $u_f$ is constant, $s$ will be approximately in direct proportion to $t$, whose proportion coefficient is $0.0949 \cdot ln(u_f) + 0.2329$.

### 2.2.4 Dissipation flow of moving section

According to this result of fixed section, dissipation of moving section is going to be discussed. The result of it will be used in 2.3.1.

In moving cross section, define the volume as $q$, total number of vehicles as $s$; in fixed cross section, define the flow is $q_1$, total number of vehicles is $s_1$; the number of vehicles between the fixed section and the moving section is $s_2$. Apparently, we get:

$$s = s_1 + s_2$$

That is:

$$q = q_1 + \frac{ds_2}{dt}$$

Assuming the density of dissipated flow is $k_1$, and the displacement of the moving section is $x$, we have:

$$\frac{ds_2}{dt} = k_1 \cdot \frac{dx}{dt}$$

The next question is how to obtain $k_1$. Based on experiences, the traffic flow dissipated shortly from a queue is relatively closer to the saturated flow. Owing to:

$$q = u \cdot k = u_f \cdot k \cdot \left(1 - \frac{k}{k_j}\right)$$

This is a quadratic function, so it obtains its maximum value when $k = \frac{1}{2} \cdot k_j$. So, based on linear speed-density function, the density of saturated flow is $k = 0.5 \cdot k_j$. On condition that dissipation time is short, it can be assumed that

$$k_1 = \frac{1}{2} \cdot k_j$$

Assume the most front-end displacement of dissipated flow is $L$, because of vehicle conservation, we have:

$$\frac{dx}{dt} \cdot k_j = \frac{dx}{dt} \cdot k_1 + \frac{dL}{dt} \cdot k_1$$

That is:

$$\frac{dx}{dt} = \frac{dL}{dt}$$

Because:

$$k = \frac{q}{u_s}$$

We obtain:

$$q = q_1 + k_1 \cdot \frac{dx}{dt} = q_1 + k_1 \cdot \frac{dL}{dt} = 2 \cdot q_1 = 0.1898 \cdot ln(u_f) + 0.4697$$

**Equation 7: The model of moving section dissipation**

$$\begin{cases} q' = 0.1898 \cdot ln(u_f) + 0.4697 \\ s_2 = [0.1898 \cdot ln(u_f) + 0.4697] \cdot t \end{cases}$$

Both the number of vehicles and the dissipation volume are double that of a fixed dissipation. It should be noted that "queue dissipation flow roughly equals to the saturated flow" is established with the condition that the dissipation time is not long, so is Equation 7. However, the limited time is still enough to our model, as the unit time of our simulation is only about $5s$.

### 2.2.5 The description of flow push-in

Since the capacity of Road Units in this model is much greater than that of a cell in common traffic cellular automata, the process of pushing vehicles into the next RU is much more complicated. Now we take another question into consideration: in the model mentioned above, how many vehicles from queue part can be accepted into free part in limited time? Since the flow from queue part is the most crowded and hardest to contain, the largest capacity of an RU can be obtained if we can obtain the number of vehicles which can be accepted from queue part. This can also be solved by differential equation.

Regardless of the abstract theoretic model, let's consider about the vehicles dissipated from the queue. Since this vehicle group is limited in speed and driving time, the leading vehicle can only move a limited distance from its original position. When this limited distance cannot completely cover the whole free driving area, the area beyond this distance is not related to containing this vehicle group. Therefore, the "valid containing segment" and "invalid containing segment" is shown as follows, in which only valid containing part is related to the capacity of free part.

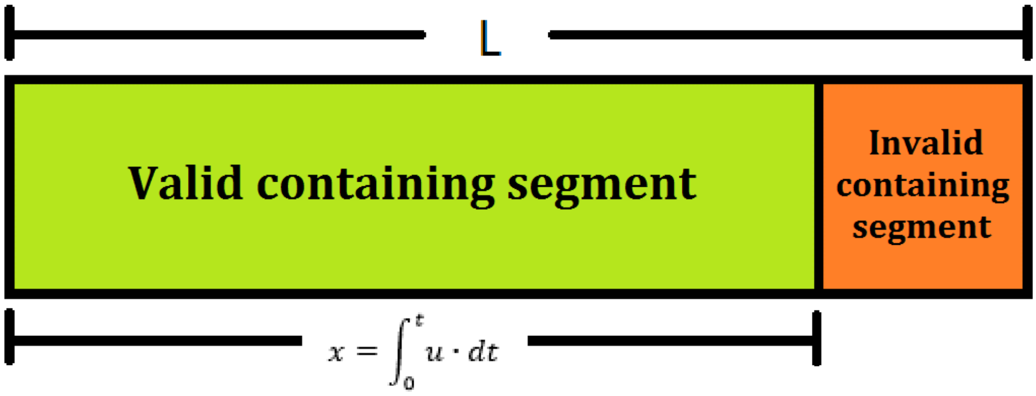


**Figure 15: Schematic diagram of RU capacity**

Now, calculate the length of the valid containing segment. The length of the valid containing segment $x$ is the integration of speed $u$:

$$x = \int_0^t u \cdot dt$$

Supposing that the initial vehicle number of the free parts is $s_0$, the total length is $L$, from Equation 6, we can get:

$$\Delta s = [0.0949 \cdot ln(u_f) + 0.2329] \cdot t$$

From linear speed-density relationship we obtain:

$$\frac{dx}{dt} = \mathrm{u} = u_f\left(1 - \frac{k}{k_j}\right)$$

$$k = \frac{s_0 + \Delta s}{L}$$

Combine the two equations above:

$$\frac{dx}{dt} = u_f\left(1 - \frac{s_0 + \Delta s}{k_j \cdot L}\right)$$

Putting these together with Equation 6 we have the derivative of $x$:

$$\frac{dx}{dt} = u_f\left(1 - \frac{s_0 + [0.0949 \cdot ln(u_f) + 0.2329] \cdot t}{k_j \cdot L}\right)$$

This indicates that the first derivative of $x$ is linear with respect to $t$. Considering that the initial value of $x$ is 0, then get:

$$x = -\frac{0.0949 \cdot ln(u_f) + 0.2329}{2 \cdot k_j \cdot L} \cdot u_f \cdot t^2 + \left(1 - \frac{s_0}{k_j \cdot L}\right) \cdot u_f \cdot t$$

However, consider the physical significance of the first derivative of $x$: the speed of the leading vehicle in the dissipated flow. Obviously this speed must be nonnegative. When the speed is 0, it means that the free moving parts are saturated. For this reason, we have:

$$\frac{dx}{dt} = max\left\{u_f \cdot \left(1 - \frac{s_0 + [0.0949 \cdot ln(u_f) + 0.2329] \cdot t}{k_j \cdot L}\right), 0\right\}$$

On the other hand, it is necessary that:

$$x < L$$

Therefore, x is written in the following form:

**Equation 8: Model of valid containing segment in flow push-in**

$$x = min\left\{L, \begin{cases} -\frac{0.0949 \cdot ln(u_f) + 0.2329}{2 \cdot k_j \cdot L} \cdot u_f \cdot t^2 + \left(1 - \frac{s_0}{k_j \cdot L}\right) \cdot u_f \cdot t, t \in \left[0, \frac{k_j \cdot L - s_0}{0.0949 \cdot ln(u_f) + 0.2329}\right] \\ \frac{u_f \cdot (k_j \cdot L - s_0) \cdot \left(1 - \frac{s_0}{k_j \cdot L}\right)}{0.1898 \cdot ln(u_f) + 0.4657}, t \in \left(\frac{k_j \cdot L - s_0}{0.0949 \cdot ln(u_f) + 0.2329}, +\infty\right) \end{cases}\right\}$$

This complicated formula is a function of $L$, $t$ and $s_0$.

Consider the largest capacity of valid containing segment, whose length is $x$. Since the max density of the road is block density $k_j$, we can suppose that it will not reach its largest capacity until the segment of road is completely blocked by vehicles. Assuming its largest capacity is $O$, we have:

**Equation 9:**

$$O = \left(k_j - \frac{s_0}{L}\right) \cdot x$$

$O$ can be obtained with Equation 8 combined.

This figure is a graph of Equation 9:

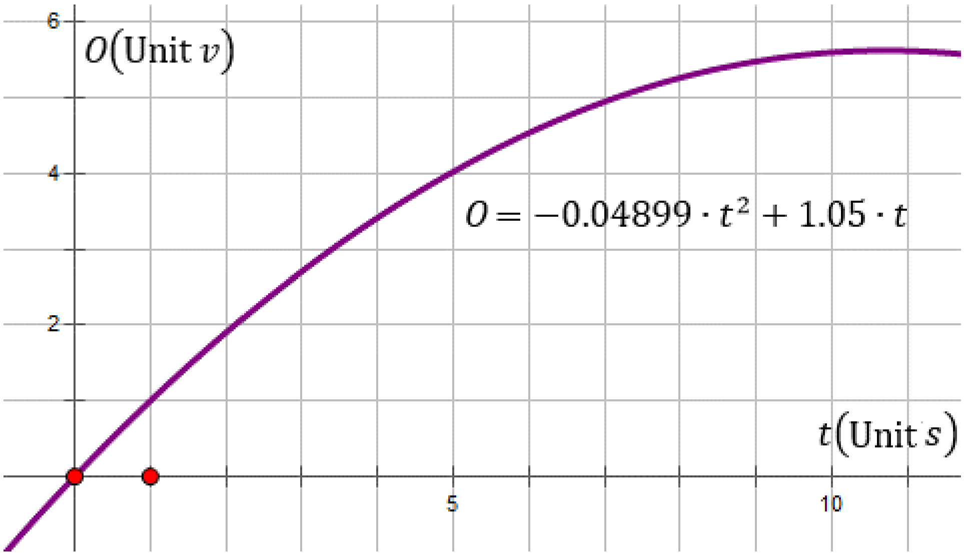


**Figure 16: An graph of Equation 9**

**(Take $u_f = 15m \cdot s^{-1}$, $k_j = 0.15v \cdot m^{-1}$, $L = 75m$, $s_0 = 6v$)**

## 2.3 Some important parameters of Road Units

### 2.3.1 Output and capacity of Road Units

During the operation of the whole traffic system, for a certain Road Unit, some vehicles entering this RU as well as other vehicles leaving it and entering next RU, this is called the output and capacity of Road Unit. In a unit time, each RU has its output ability and capacity based on its basic parameters. Therefore, following parameters are defined:

**Definition 8:**

Road Unit's Largest Output $I$: when only considering the traffic condition of current Road Unit, the maximum number of vehicles which can go out of this RU in a unit time.

**Definition 9:**

Road Unit's Largest Capacity $O$: when only considering the traffic condition of current Road Unit, the maximum number of vehicles which can go into this RU in a unit time.

Consider the output process first. The output process of Road Unit consists of both the dissipation of queue part and output of free part. For a specific Road Unit, however, it is possible that not both processes are carried out. During the calculation, we will put forward a general calculating method, which will take both two parts into consideration.

Assume that the length of Road Unit is $L$, the total number of vehicles is $s$, the length of queue part is $L_q$, the number of vehicles in queue part is $s_q$, the length of free part is $L_f$, the number of vehicles in free part is $s_f$, the speed in free part is $u$ as well as density is $k$. Define that average length of vehicle (including the gap between two adjacent vehicles) is $avgL$ ($avgL = \frac{1}{K_j}$), time is $t$, unit time is $t_{unit}$. Here we define the simulation time is a relatively small value. Assume that function $f_1(t)$ is the function of the number of dissipated vehicle $s$ about time $t$ (obtained from Equation 7), function $f_2(s_0, L_0, t)$ is the function of $s$, the number of vehicles which go into queue part from free part about time $t$, length of free part $L_0$ and vehicle number in free part $s_0$ (obtained from Equation 3), function $f_3(t)$ is the function of $s$, the number of the vehicles which have completely passed the front section of the RU about time $t$ (obtained from Equation 6). With the change of time, vehicles of queue part will leave, and vehicles of free part will enter queue part. We assume that the decrease in length of queue part caused by dissipation is $\Delta L_1$, and the increase in length of queue part caused by preceding is $\Delta L_2$, the total decrease in length of queue part is $\Delta L$. So we have:

$$\Delta L_1 = f_1(t) \cdot avgL \qquad ①$$

$$\Delta L_2 = f_2(s_f, L_f, t) \cdot avgL \qquad ②$$

$$\Delta L = \Delta L_1 - \Delta L_2 \qquad ③$$

Assuming the length of queue part falls to 0 when time is $t_0$, that is:

$$\Delta L = L_q$$

When $t_0 = 0$, the output of the RU is only from free part, so:

$$I = \frac{u \cdot t_{unit}}{L} \cdot s_f$$

When $0 < t_0 \le t_{unit}$, output of the Road Unit consists of both parts. The first part is completely the vehicles from queue part, so the number of these vehicles is $I_1 = f_3(t_0)$. The calculation of number of output vehicles from the free part is more complex. After completely dissipation in queue part, the vehicles in the RU is divided into two parts. The first part consists of vehicles which have begun to move but haven't left the front section of the RU yet. The second part consists of vehicles in the free part. Since the moving of vehicles in the second part is affected by the first part, our calculation is based on parameters of the first part.

Assuming that the length of average free flow of first part is $L_n$, number of vehicle is $s_n$, density is $k_n$, speed is $u_n$, we have:

$$L_n = L_q + \Delta L_2 \qquad ④$$

$$s_n = [s - f_3(t_0)] - [s_f - f_2(s_f, L_f, t_0)] \qquad ⑤$$

$$k_n = \frac{s_n}{L_n} \qquad ⑥$$

$$u_n = u_f \cdot (1 - \frac{k_n}{k_J}) \qquad ⑦$$

$$I_2 = min\{\frac{u_n \cdot (t_{unit} - t_0)}{L} \cdot (s - f_3(t_0)), s - f_3(t_0)\} \qquad ⑧$$

From formula 1-8, we obtain:

$$I_2 = \min\{u_f \cdot \left(1 - \frac{[s - f_3(t_0)] - [s_f - f_2(s_f, L_f, t_0)]}{k_j \cdot \left(L_{q+}f_2(s_f, L_f, t_0)\right) + 1}\right) \cdot \frac{t_{unit} - t_0}{L} \cdot (s - f_3(t_0)), s - f_3(t_0)\}$$

$$\mathrm{I} = I_1 + I_2 = f_3(t_0) + \min\{u_f \cdot \left(1 - \frac{[s - f_3(t_0)] - [s_f - f_2(s_f, L_f, t_0)]}{k_j \cdot \left(L_{q+}f_2(s_f, L_f, t_0)\right) + 1}\right) \cdot \frac{t_{unit} - t_0}{L} \cdot (s - f_3(t_0)), s - f_3(t_0)\}$$

When $t_0 > t_{unit}$, output of the RU only contains the dissipation flow from queue part:

$$I = f_3(t_{unit})$$

In conclusion, we obtain the formula of largest output of a Road Unit:

**Equation 10: Model of largest output of Road Unit**

$$I = \begin{cases} \dfrac{u \cdot t_{unit}}{L \cdot s_f}, (t_0 = 0) \\ f_3(t_0) + min\left\{u_f \cdot \left(1 - \dfrac{[s - f_3(t_0)] - [s_f - f_2(s_f, L_f, t_0)]}{k_j \cdot \left(L_{q+}f_2(s_f, L_f, t_0)\right) + 1}\right) \cdot \dfrac{t_{unit} - t_0}{L} \cdot \left(s - f_3(t_0)\right), s - f_3(t_0)\right\}, (0 < t_0 \le t_{unit}) \\ f_3(t_{unit}), (t_0 > t_{unit}) \end{cases}$$

As for the largest capacity of Road Unit, in 2.2.5, if we put the free parameter into formula 9, the formula of the largest capacity will be:

**Equation 11: Model of largest capacity of Road Unit**

$$s = \left(k_j - \frac{s_{f0}}{L_f}\right) min\left\{L_f, \begin{cases} -\dfrac{0.0949 \cdot ln(u_f) + 0.2329}{2 \cdot k_j \cdot L_f} \cdot u_f \cdot t^2 + \left(1 - \dfrac{s_{f0}}{k_j \cdot L_f}\right) \cdot u_f \cdot t, \left(t \in \left[0, \dfrac{k_j \cdot L_f - s_{f0}}{0.0949 \cdot ln(u_f) + 0.2329}\right]\right) \\ \dfrac{u_f \cdot (k_j \cdot L_f - s_{f0}) \cdot \left(1 - \dfrac{s_{f0}}{k_j \cdot L_f}\right)}{1.898 \cdot ln(u_f) + 0.4657}, \left(t \in \left(\dfrac{k_j \cdot L_f - s_{f0}}{0.0949 \cdot ln(u_f) + 0.2329}, +\infty\right)\right) \end{cases}\right\}$$

A simple method of solving $t_0$ is to solve differential equations. Note that $\Delta L$ is not monotonic. According to the inference of 2.1 and 2.2, we know that the output flow in the front of the queue is close to saturated flow while the flow entering the queue from rear may be greater or less than saturated flow. But when a queue has been dissipated completely, in our model, the nature of unsaturated flow leads to this: within a unit time, a Road Unit will not generate a queue for a second time if the first queue has gone. So when the differential equations suggest that $\Delta L = L_q$, this moment is $t_0$.

### 2.3.2 Average output volume of Road Unit

The largest output describes the output ability in unit time, we need a parameter to describe the output ability in a certain period of time. We define that:

**Definition 10:**

Average output volume $K_r$ is the average output volume of a Road Unit in a time period, that is:

$$K_r = \frac{\int_0^{t_{unit}} q \cdot dt}{t_{unit}}$$

The average volume is not an instantaneous one. Obviously, instantaneous volume can more truly reflect the output process of Road Unit, but the change of instantaneous volume is complex. In order to simplify the calculation, we propose:

**Assumption 6:**

In a unit time, the output volume of a Road Unit is constant. Use average output volume $K_r$ instead of $q$.

According to the assumption, average output flow equal to largest output divided unit time:

$$K_r = \frac{I}{t_{unit}}$$

### 2.3.3 Evaluation of traffic condition based on drivers' psychology

In section 3 of this paper, we will need to describe the traffic condition of a Road Unit in the view of drivers. Now we discuss: which information on the road is most easily obtained for drivers and most effective for evaluating the traffic condition?

For human eyes, it's difficult to notice the small changes in vehicle density. However, even slightest difference in speed can be easily noticed by drivers, because the speed difference between surrounding vehicles and their own will cause relative displacement, which is very obvious. Therefore, speed is an important parameter for drivers to evaluate the surrounding traffic condition.

Besides, a long vehicle queue is also very conspicuous on the road. Vehicle queue means that traffic accidents may have happened ahead, and the traffic condition is bad. Generally speaking, the drivers will actively avoid the lanes whose queues are long.

Now, we apply the above reasoning to Road Units. To keep the model simple, we only consider the traffic condition of current RU and ignore the conditions of other RUs, according to drivers' short-sightedness. In more organized words:

1) The drivers can obtain speed in free part $u_i$;
2) The drivers can obtain the number of vehicles in queue part $s_Q$.

Define the traffic condition coefficient as $t_f$, and: the bigger $t_f$, the better the traffic condition. So:

3) $t_f$ should be in positive correlation with $u_i$;
4) $t_f$ should be in negative correlation with $s_Q$.

Furthermore, the evaluation of traffic condition that drivers made should have the following characters:

5) when the traffic condition is very good or bad, $t_f$ is not sensitive to traffic condtion;
6) when the traffic condition is moderate, $t_f$ is sensitive to traffic condition.
7) As long as one of $u_i$ and $s_Q$ gets bad enough, $t_f$ becomes bad.

Cosine function can realize 5) and 6). To realize 7), the two items of $u_i$ and $s_Q$ are multiplied. Considering the standardization of the value of $t_f$, after attempts, we use this expression to evaluate the traffic condition in the view of drivers:

**Equation 12: Evaluation of traffic condition based on drivers' psychology**

$$t_f = \frac{1}{4} \cdot \left\{ cos\left[\pi \cdot \left(1 - \frac{u_i}{u_f}\right)\right] + 1 \right\} \cdot \left[ cos\left(\pi \cdot \frac{s_Q}{s_{max}}\right) + 1 \right]$$

Where, $s_{max}$ is the largest number of vehicles that the Road Unit can hold, namely $s_{max} = l \cdot k_j$. Obviously, $t_f \in [0,1]$.

The following figure is a graph of $t_f$ about $u_i$ and $s_Q$ drawn by Matlab. Colors in the graph represents different values of $t_f$. The $Z$ axis stands for $t_f$, and $X$ axis and $Y$ axis represent $u_i$ and $s_Q$ respectively. It can be seen that as $u_i$ decreases and $s_Q$ increases, $t_f$ decreases. The decrease is steeper when $t_f$ is about 0.5, and becomes much more gentle when $t_f$ is close to 1 or 0. The graph satisfies all the 7 demands above. This function will be used in 3.2.

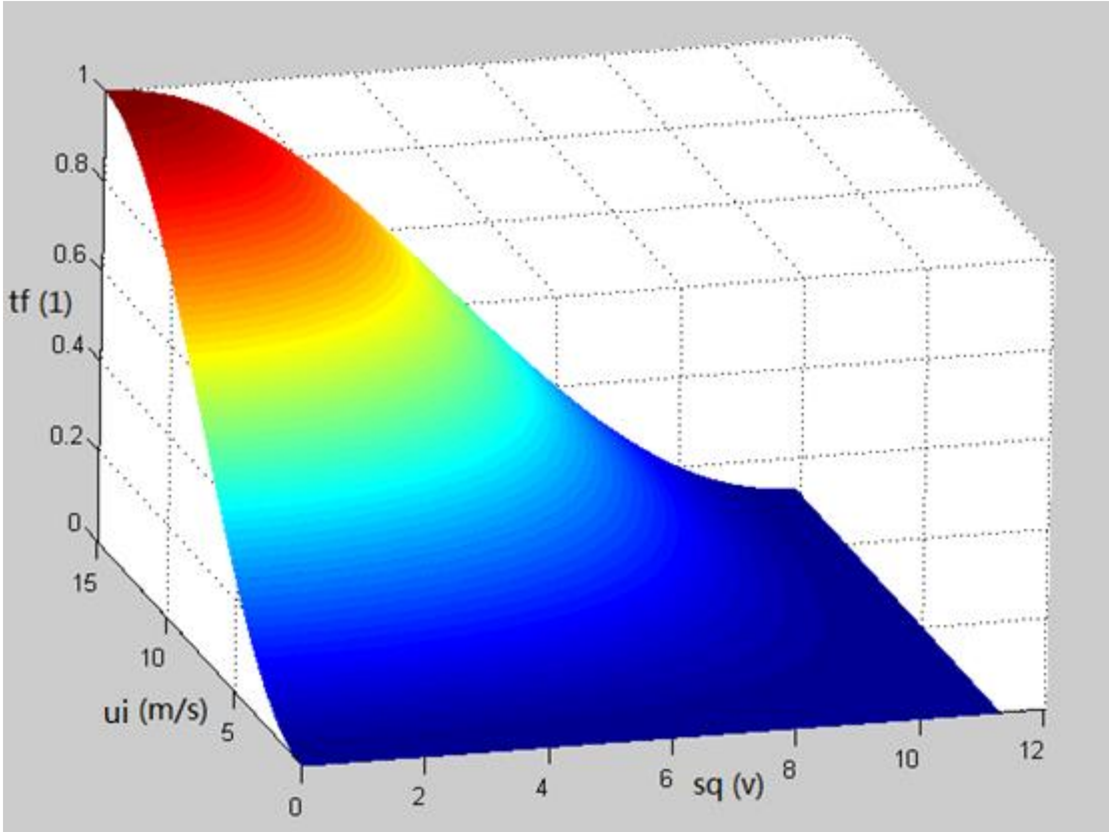


**Figure 17: The graph of $t_f$ about $u_i$ and $s_Q$** [3]
**(Take $u_f = 15m \cdot s^{-1}$, $s_{max} = 11.25v$)**

## 2.4 The recalculation of basic parameters of Road Unit

After flow distribution (to be discussed in 3.3), the condition of Road Unit will change because of the output and capacity process. The change of the condition affected by output is computed according to the formulas in 2.3.1 and the assumption in 2.3.2, while the condition change affected by capacity is computed according to the expression of largest capacity and the rule that the vehicles that enter into the Road Unit are directly added to the free part. Since other parameters can be obtained by $s_q$ and $s_f$, we will just discuss the computation of the parameters of Road Unit based on the basic conclusions above.

Set that the actual number of largest output vehicles is $s_{out}$, the number of vehicles that remain inside the RU is $s_r$, the number of vehicles that enter the RU is $s_{in}$, the original number of vehicles in the RU is $s_0$, the original number of vehicles that are in the queue is $s_{q0}$, the original number of vehicles that are in free part is $s_{f0}$, the original length of free part is $L_{f0}$, the length of the whole RU is $L$, unit time is $t_{unit}$, the remaining time after output is $t_r$, the total number of vehicles of the RU after recalculation is $s$, the number of vehicles in free part is $s_f$, the number of vehicles in the queue is $s_q$. Where, $s_{out}$ and $s_{in}$ have been computed in matrix $Flow$ (to be discussed in 3.3), and $s_r$ is obtained according to $I = s_{out} + s_r$.

### 2.4.1 The recalculation of output that only includes free part

When the number of output vehicles equals to the largest output, $I = S_{out}$, so we have

$$\begin{cases} s = s_0 - s_{out} + s_{in} \\ s_q = 0 \\ s_f = s \end{cases}$$

When the number of output vehicles is less than the largest output, according to the assumption in 2.3.2, the period of time that the Road Unit doesn't output vehicles $tr = \frac{s_r}{I} \cdot t_{unit}$.

In the period that no vehicles are output, there must be a blockage in front of the RU, so queue part begins to form, which can be computed according to Equation 3. The remaining length of free part is $l - u \cdot (t_{unit} - tr)$, the number of remaining vehicles is $s_0 - s_{out}$. The computation formula of , $s_q$ , $s_f$ is as follows:

$$\begin{cases} s = s_0 - s_{out} + s_{in} \\ s_q = f_2(s_0 - s_{out}, L - u \cdot (t_{unit} - tr), tr) \\ s_f = s - s_q \end{cases}$$

### 2.4.2 The recalculation of output that includes both parts

When the number of output vehicles equals to the largest output, the process is the same as that of 2.4.1.

When the number of output vehicles is less than the largest output, the parameters of the dissipated flow change little over time, and the two parts mentioned in 2.3.1, which are called respectively the "front free part" and the "rear free part", have similar traffic conditions. So when the output process stops and there's a blockage in front of the RU, we can approximatively think that the rear free parts have the same parameters as the front part.

First of all, transform the rear free part into the front free part (here we directly use the parameters $k_n$, $t_0$, $I_2$ defined in 2.3.1, and set that the number of vehicles of back-end homogeneous free flow is $s'$,the transformed length is $L'$):

$$s' = s_0 - s_{out}$$

$$L' = \frac{s'}{k_n}$$

The expression of s, $s_q$, $s_f$ is as follows:

$$\begin{cases} s = s_0 - s_{out} + s_{in} \\ s_q = f_2(s', L', tr) \\ s_f = s - s_q \\ tr = \frac{s_r}{I_2} \cdot (t_{unit} - t_0) \end{cases}$$

### 2.4.3 The recalculation of output that only includes queue part

Figure 18 shows the actual condition of Road Unit in two circumstances:

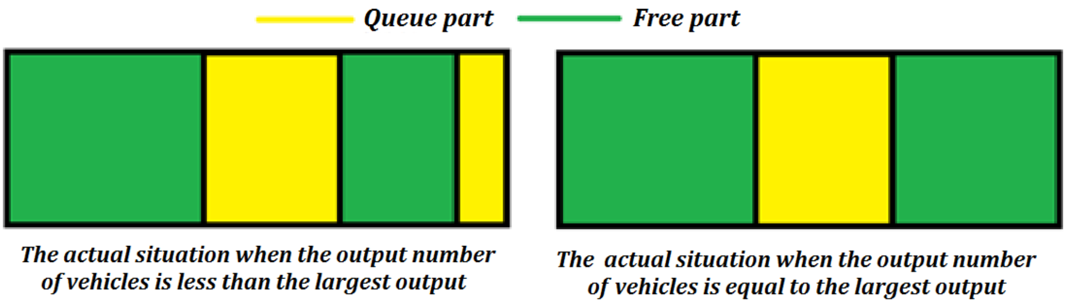


**Figure 18: The actual condition of Road Unit after queue dissipation**

According to the definition of it, Road Unit is simply made up of only one queue part and one free part, but the two circumstances in the above figure don't meet the definition. Therefore, we should process the two circumstances by changing the queue part and the free part in the middle of the RU into queue part, as shown in Figure 19:

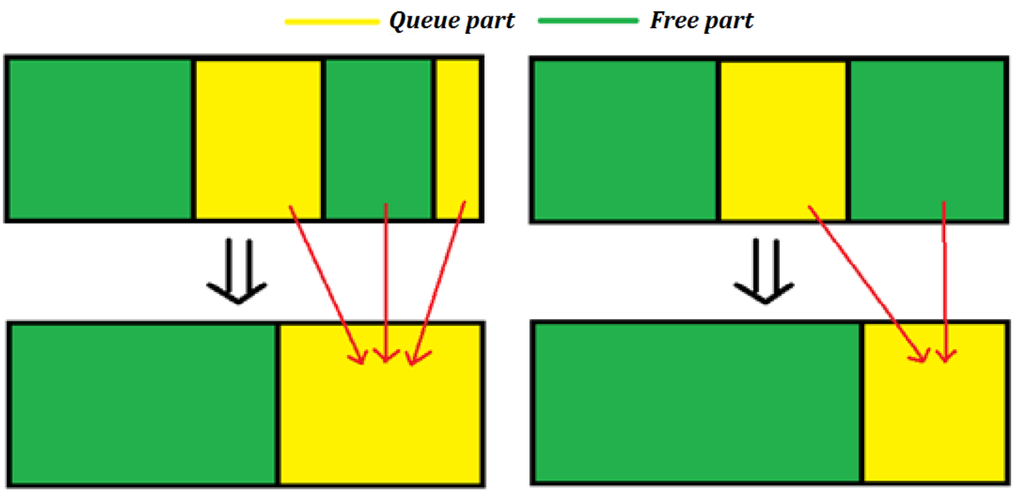


**Figure 19: The conversion of the parts in the middle**

After conversion, the expressions of $s$、$s_q$、$s_f$ are as follows:

$$\begin{cases} s = s_0 - s_{out} + s_{in} \\ s_q = s_{q0} - s_{out} + f_2(s_{f0}, L_{f0}, t_{unit}) \\ s_f = s - s_q \end{cases}$$

# 3 Lane change and flow distribution

In section 2, we describe the state transition within the Road Units. In this section we will deal with the problems of state transition between Road Units, which are lane change of vehicles and flow distribution. Based on a series of assumptions about Generalized Intersection, we can change the random continuous lane change process into certain, discrete process by limiting the lane change process to the bipartite graph of Generalized Intersection.

The process of distributing the input flow of Generalized Intersection to output is called flow distribution. The combination of flow that pass through every side of bipartite graph is called volume matrix. The process of vehicles choosing lanes is closely related to the description of drivers' psychology. Because of the limit of output and capacity, the calculation of flow matrix is actually a distribution that best meets drivers' psychology under the limit. Next, we will start with our original simple psychology model of drivers, simplifying and formularizing the complex process of flow distribution.

## 3.1 Two motivations of lane change

### 3.1.1 **Lane change based on maximizing efficiency**

It is obvious that the driver would like to arrive at the destination sooner when driving on the road. When driver find that the traffic condition of other lanes is obviously better than that of current lane, he would change lane to increase speed so that he can arrive at destination sooner. The motivation of lane change based on maximizing efficiency is related to the traffic condition of the Road Unit Group in front.

We need to note that the driver cannot always maximize the efficiency as he wants. Since changing lane will cause the vehicle to go on a different direction, some drivers tend not to change lane even if they face a Road Unit with a better traffic condition. At the same time, some drivers cannot change lanes to maximize efficiency because of the driving direction or unexpected traffic condition.

### 3.1.2 Lane change based on direction of destination

As stated in 3.1.1, vehicles tend to maximize the efficiency far away from the actual intersection. So at a certain distance from the intersection, vehicles are not strictly on the lane that leads to direction of destination but are more likely to be on the lane with a better traffic condition. As stated in 1.2.4, vehicles will gradually change lane according to their direction of destination, and drivers must finish lane change before the mandatory change land of the actual intersection (i.e., channelized area) because of the restriction of traffic regulations. It's not difficult to see that the motivation of lane change based on direction of destination is related to the distance from the actual intersection.

## 3.2 Lane change mechanism: attraction model and tendency of drivers

In this section, we will describe the mechanism of lane change. The mechanism is based on Assumption 6: lane change cannot happen within a Road Unit. The way of lane change calculation is calculating the tendency of vehicles in the input side Road Units entering the output side Road Units.

To describe this tendency, we introduce attraction model. Attraction model will deal with factors that influence lane change such as two motivations of lane change, conservatism and short-sightedness of drivers, and physical limit of lane change, to calculate the psychological tendency of drivers.

**Definition 11: the definition of attraction**

Attraction refers to the driver tendency of going to a certain lane. Attraction coefficient $G_{ijk}$ is used to describe the attraction of the output side Road Units to the vehicles in the input side Road Units, which represents the general attraction of the output side Road Unit $j$ to the vehicles with direction of destination $k$ in the input side Road Unit $i$.

Without extra description, subscript $i$ , $j$ , $k$ of matrix in the following have the same meanings as $i$ , $j$ , $k$ in Definition 11.

For convenience, we define the concept of direction of destination:

**Definition 12:**

Direction of destination is the corresponding driving or turning direction of the lane for next actual intersection. Because in reality, the lanes that belong to the same direction of destination $k$ are connected to each other. So here is the definition for $Ks$ and $Ke$:

$$\forall\, i \in Dir_k, i \in [Ks, Ke]$$

Where, $Dir_k$ refers to the set of lanes with direction of destination $k$.

### 3.2.1 A psychological model based on conservatism and short-sightedness

Normally drivers will see the traffic condition of each lane not far away when they are driving, but they can't see the traffic condition in the distance because their view is blocked by vehicles ahead. Besides, because the traffic condition in the distance has a relatively small impact on it, drivers will focus on the traffic condition in front and both sides, but don't pay attention to farther areas. This is drivers' short-sightedness. On the other hand, drivers will focus more on the most outstanding lane and care little about other lanes. All in all, given a set of values $A_i$ and the weights $Af_i$ calculated according to their distance, we can get the integrated values of this set of values in drivers' eyes:

**Equation 13: Psychological short-sightedness model of drivers**

$$A = \frac{1}{2 \cdot (n-1)} \cdot \sum_{i \neq imax} A_i \cdot Af_i + \frac{1}{2} \cdot A_{imax} \cdot Af_{imax}$$

Where, $n$ is the sum of data, $imax$ is the subscript of the largest data, that is:

$$\forall\, i \in [1, n], A_{imax} \cdot Af_{imax} \geq A_i \cdot Af_i$$

$Af$ is a number that is only related to the distance between the driver and destination. According to the experience, the value table of $Af$ array is as follows:

| Lane distance I | 0 | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|---|
| AF | 1.00 | 0.90 | 0.81 | 0.73 | 0.66 | 0.59 |

**Table 3: The value of $Af$ array**

In Equation 13, the former item of the expression of $A$ corresponds to the attention to outstanding data, and the latter corresponds to the limited attention of other items.

Conservatism refers to the reluctance of drivers to change the current situation. Generally speaking, drivers are reluctant to change lane except that some lanes have a remarkable advantage over the lane where the driver is driving on. This is because lane change needs drivers' energy and time, and reduces efficiency, and even takes some risks. On the other hand, for direction of destination, if the driver is already on the right direction, he will be extremely reluctant to leave the right direction. This psychology increases as the distance to the next actual intersection decreases. To be specific, if we quantify the excellent level of traffic condition of each lane in driver's eyes to an array $y_j$ and the driver is on lane $i$, then after considering conservatism, $y$ will be replaced by $G$. We can obtain the qualitative expression of psychological conservatism model of drivers:

$$G_j: \begin{cases} = const \cdot y_j, when\ i = j\ or\ y_j \gg y_i \\ < const \cdot y_j, when\ y_j > y_i\ is\ moderately\ significant \\ \ll const \cdot y_j, when\ y_j > y_i\ is\ not\ significant\ or\ y_j \leq y_i \end{cases}$$

Where, $const$ is a fixed constant that doesn't change with other parameters.

### 3.2.2 Attraction of a single lane

Next we will discuss the bottom of attraction model: attraction of a single lane. According to statement in 3.1, attraction should contain attraction of the destination direction and attraction of maximizing efficiency.

Maximizing efficiency is always the aim of drivers. A lane with a good traffic condition is more attractive to drivers. Attraction of drivers' maximizing efficiency of a lane depends on the traffic condition coefficient $t_f$ of this lane and is proportional to it. However, just as stated in 3.1.1, maximum efficiency cannot always be reached. Next, we will discuss the relationship between directional lane change, mandatory lane change and lane change of maximizing efficiency in detail.

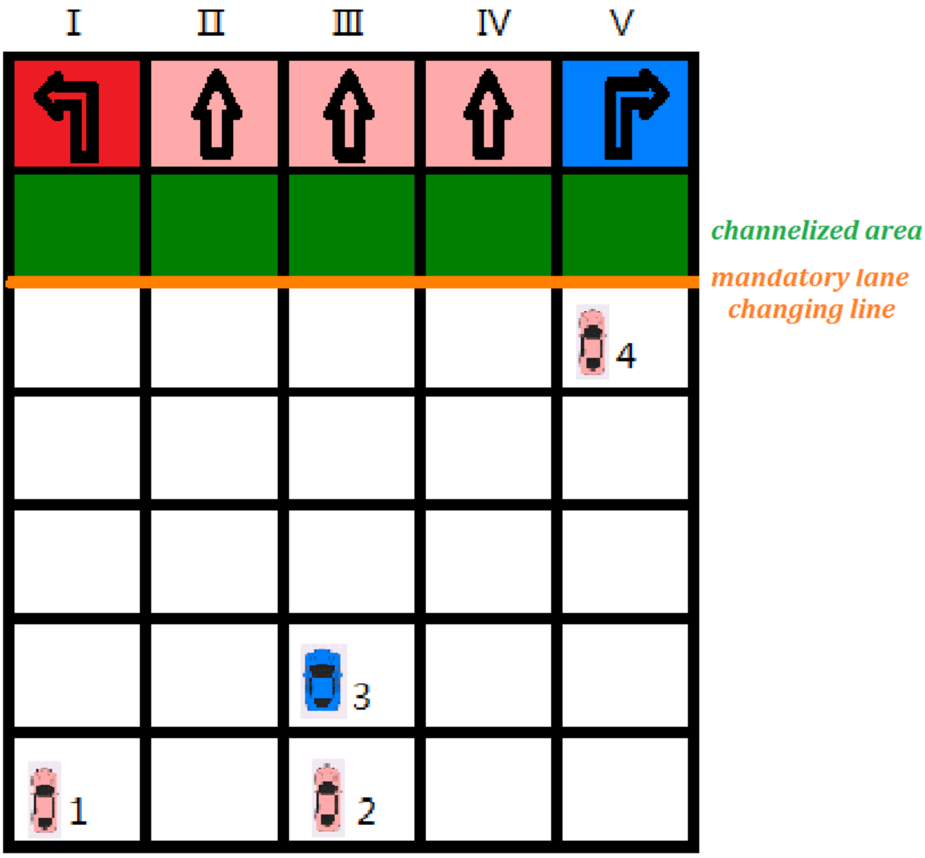


**Figure 20: Schematic diagram of lane change based on direction of destination**

As shown in the diagram, the color of vehicle refers to its direction of destination. Because drivers are driving with an aim (diversion at actual intersection), drivers have a tendency that lane change can reach his destination. This tendency is not obvious when they are far away from actual intersection, but becomes more obvious as he approaches. Like Vehicle 4, at this moment it's in front of mandatory lane change line (i.e., the boundary of channelized area), but it is not on the lane that the driver wants. So at this time Lane II , III , IV has an infinite attraction to Vehicle 4. However, as for Vehicle 1 which is not on the right lane as well, since Vehicle 1 is far away from mandatory lane change line, Lane II , III , IV is not so attractive to Vehicle 1. We conclude, the attraction of direction of destination $W_{ijk}$ is in inverse proportion to the distance between the vehicle and mandatory lane change line. For Vehicle 3, its aim is Lane V, but Lane IV is still attractive to the driver because Lane IV is between Lane III and Lane V, and Lane IV can make Vehicle 3 closer to its target lane. However, for the vehicles which are already on target lane, like Vehicle 2, the outer lane (Lane I , V ) not only have no attraction for it , but it might be a negative number.

Note that even though the boundary of target lane(such as the boundary of Lane I , II in the diagram) is not a real boundary, it has great impact on the psychology of drivers, especially when drivers are very close to channelized area. The drivers who are on the target lane will be extremely reluctant to cross the boundary from right direction to wrong direction, but those who are not on the target lane will hope to cross the boundary as soon as possible.

The attraction of direction of destination is apparently related to the distance between the current vehicle and next actual intersection. If the vehicle is far enough from next actual intersection, the effect of this attraction will tend to zero. On the contrary, if the vehicle is close to next actual intersection, this effect will tend to infinity (take mandatory lane change into account). Besides, given that the lane change of vehicles focus on the place that is near actual intersection. We set $x = \frac{d}{u_f}$ (use the proportion of distance and speed limit to describe the distance effect), and use $f(x)$ to describe the significance of lane change of direction of destination in drivers' eyes who are in that position. The following are some demands:

1) When $x \to 0$, $f(x) \to +\infty$;
2) When $x$ is sufficiently small, the value of $f(x)$ can make attraction of direction of destination sufficiently larger than attraction of maximizing efficiency (take $x = 10$ (when $u_f = 15m \cdot s^{-1}$, $d = 150m$), $f(x) = 0.3$);
3) When $x$ is relatively small, $f(x)$ keeps a relatively high level, representing that this area is where drivers mainly change lane based on direction of destination (take $x = 20$ (when $u_f = 15m \cdot s^{-1}$, $d = 300m$, $f(x) = 0.11$), take $x = 40$ (when $u_f = 15m \cdot s^{-1}$, $d = 600m$), $f(x) = 0.03$);
4) When $x$ is relatively large, $f(x)$ decreases slowly to 0 as $x$ decreases (take $x = 40$ (when $u_f = 15m \cdot s^{-1}, d = 1500m$), $f(x) = 0$);
5) When $x$ is sufficiently large, $f(x) = 0$, representing that drivers who are far enough away from actual intersection will not affected by attraction of direction of destination.

In conclusion, combined with the unity of the data itself, with many adjustments, we get the following expression:

$$f(x) = \begin{cases} \frac{3}{x}, (x \leq 10) \\ -0.019x + 0.49, (10 < x \leq 20) \\ -0.004x + 0.19, (20 < x \leq 40) \\ -0.0005x + 0.05, (40 < x \leq 100) \\ 0, (x > 100) \end{cases}$$

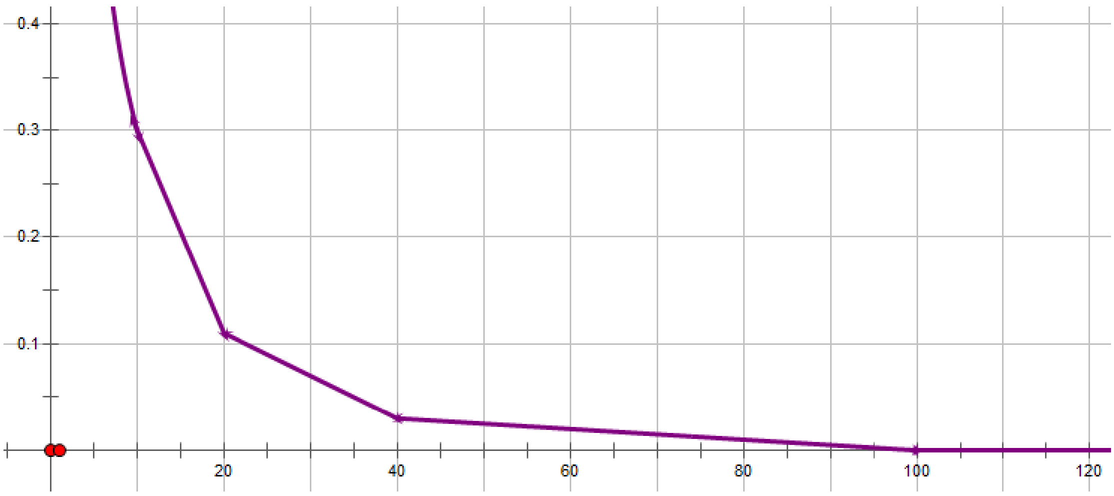


**Figure 21: The graph of $f(x)$**

According to the above description, we obtain attraction of direction of destination $W_{ijk}$

through inductions and attempts:

**Equation 14: Attraction of direction of destination of a single lane**

$$W_{ijk} = \begin{cases} (1 + y_w \cdot |i-j|) \cdot f\left(\frac{d}{u_f}\right), (Ke_k < i \; or \; i < Ks_k, and \; Ks_k \le j \le Ke_k) \\ -(1 + y_w \cdot |i-j|) \cdot f\left(\frac{d}{u_f}\right), (Ks_k \le i \le Ke_k, and \; Ke_k < j \; or \; j < Ks_k) \\ -(0.8 + y_w \cdot |i-j|) \cdot f\left(\frac{d}{u_f}\right), (j < i < Ks_k \; or \; Ke_k < i < j) \\ y_w \cdot |i-j| \cdot f\left(\frac{d}{u_f}\right), (i < j < Ks_k \; or \; Ke_k < j < i) \\ 0, (i = j, or \; Ks_k \le i \le Ke_k \; and \; Ks_k \le j \le Ke_k) \end{cases}$$

In this equation, $|i-j| \cdot y_w$ means: the farther the vehicle is away from target direction, the stronger the appeal of directional lane change is. The 1 which is added to it refers to the psychological influence of crossing the boundary between different directions on drivers. 0.8 refers to the psychological influence of aggravating the mistake even on the wrong direction. Where, $y_w$ is an empirical constant, $y_w = 0.2$.

Connecting traffic condition coefficient in 2.3.3, we consider $t_{f_j}$ as the attraction of traffic condition of lane $j$ (i.e., maximizing efficiency) to drivers. So we get the unified attraction of a single lane:

**Equation 15: The unified attraction of a single lane**

$$A_{ijk} = W_{ijk} + t_{f_j}$$

### 3.2.3 The calculation of superimposed attraction

The lane attraction $A_{ijk}$ stated above describes the attraction of a single lane to drivers. However, it is not the final attraction that leads to drivers' changing lane. The final attraction of lane change behavior is called superimposed attraction. What the attraction in Definition 11 exactly refers to is superimposed attraction.

Consider this case (as the following diagram shows): a driver is driving on Lane $i$ (the red car in the following diagram). The traffic conditions of Lane $i+1$ and Lane $i$ are similar but the traffic condition of Lane $i+2$ is much better than that of Lane $i$. In this case, most drivers in reality are likely to change lane to Lane $i+1$ first, and then change lane to Lane $i+2$ : even though Lane $i+1$ is not outstanding, drivers are also willing to change lane to it. From this case, it's not difficult to conclude that attraction of changing lane to a certain lane is the superposition of attraction of lane change towards that direction.

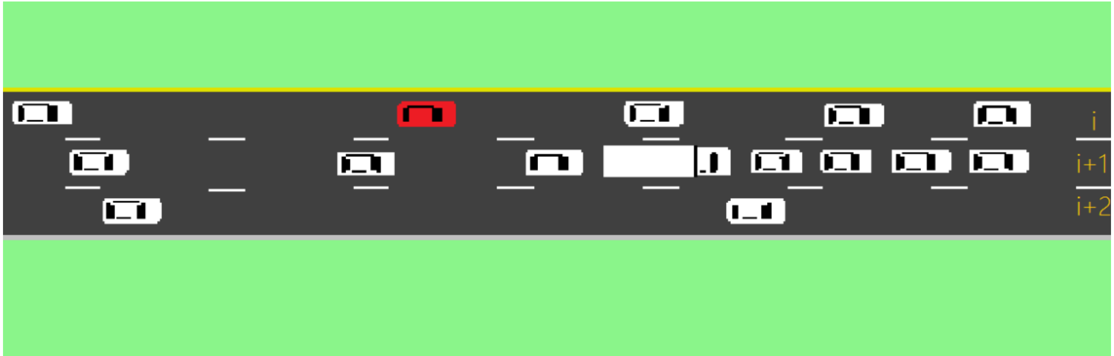


**Figure 22: Schematic diagram of applying superimposed attraction**

Consider another case(as the following diagram shows):a driver is driving on Lane $j$ (the red car in the following diagram). The traffic conditions of Lane $j+1$ and Lane $j$ are similar, and the traffic condition of Lane $j+2$ is much better than that of Lane $i$, but the traffic condition of Lane $j+3$ is very bad. In reality, drivers will not refuse to change lane towards that direction because Lane $j+3$ lacks attraction: drivers want to change lane to Lane $j+2$ but not Lane $j+3$. So, the calculating range of superimposed attraction should be limited. In this case, Lane $j+3$ shouldn't be calculated. From this case, when calculating superimposed attraction, we need to enumerate lanes which drivers really tend to change lane to and use the max one as superimposed attraction.

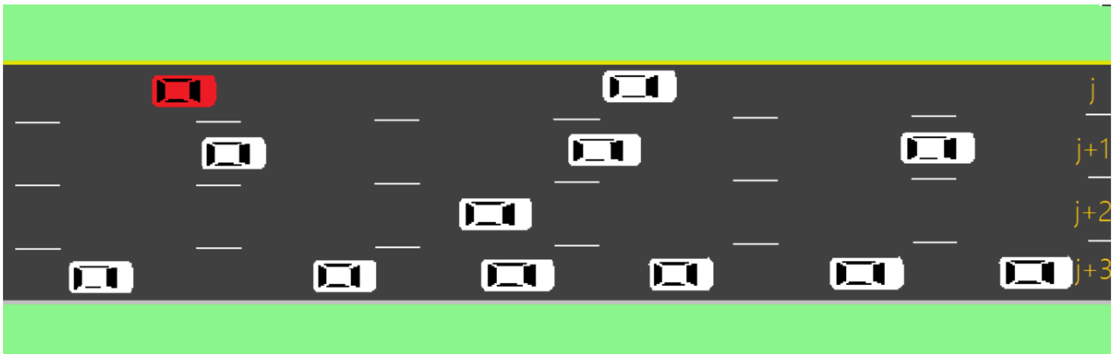


**Figure 23: Schematic diagram of applying superimposed attraction in optimized range**

Applying short-sighted model in Equation 13, we can conclude:

**Equation 16: The unified attraction model after superposition**

$$y_{ijk} = \begin{cases} max\left\{\frac{1}{2\cdot(\#R-1)}\cdot\sum_{r\neq rmax,r\in R} A_r\cdot Af_r + \frac{1}{2}\cdot A_{rmax}\cdot Af_{rmax}\right\}, (i\neq j) \\ A_{ijk}\cdot Af_0, (i=j) \end{cases}$$

Where, $\#R$ is the number of elements of the set of Road Units that was got closed to when changing lane from $i$ to $j$. When calculating superimposed attraction, we enumerate $R$ by enumerating lanes that drivers finally reach.

### 3.2.4 Application of conservatism coefficient in the attraction model

In order to quantify the qualitative expression in 3.2.1, we define conservatism coefficient $B_{ik}$. Its meaning is conservatism of staying on lane $i$ for drivers on lane $i$ with direction of destination $k$. Specifically speaking, is:

$$G_{ijk} = \begin{cases} B_{ik} \cdot y_{ijk}, (i = j) \\ y_{ijk}, (i \neq j) \end{cases}$$

The calculation of $B_{ik}$ is listed below:

**Equation 17: Quantitative expression of driver's psychological conservatism model**

$$\begin{cases} B_{ik} = 1 + e \cdot z \\ x = \dfrac{Max\{y_{ijk}\}}{y_{iik}} \\ z = \begin{cases} 9, (0 \leq x \leq 1.05) \\ 9 - 16 \cdot (x - 1.05), (1.05 \leq x \leq 1.55) \\ 0, (1.55 \leq x) \end{cases} \\ d_i = \begin{cases} Ks_k - i, (i \leq KS_k) \\ i - Ke_k, (i \geq Ke_k) \end{cases} \\ e = \begin{cases} 1, (Ks_k \leq i \leq Ke_k) \\ Max\left\{0,1 - \dfrac{(1 + y_w \cdot d_i) \cdot f\left(\frac{d}{u_f}\right)}{0.24}\right\}, (i \leq Ks_k \text{ or } i \geq Ke_k) \end{cases} \end{cases}$$

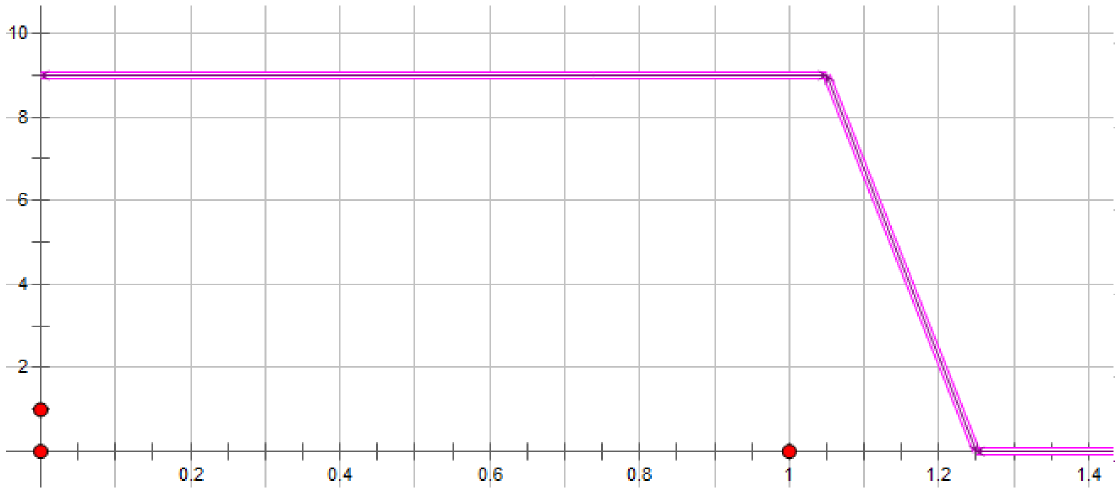


**Figure 24:The graph of z - x relation in Equation 17**

Where, $e$ represents the conservatism of staying on lanes of this direction of destination, $z$ represent s the conservation of staying on the current lane, 1.05、1.25、9、16、0.24 are empirical constants， and $y_w$ is empirical constant defined in 3.2.2.

It is obvious that the equation set is reasonable. From the graph above, we can see that it corresponds to the regularity: drivers are significantly conservative when there is no other lanes significantly better than the current lane; as the attraction of other lanes increase, the conservatism of staying on the current lane decrease. Principle of calculation method of $e$ is similar to the calculation method of $W_{ijk}$ in 3.2.2. Besides, For $B_{ik}$, it is obvious that:,

$$B_{ik} \in [1,10]$$

Where, when the current lane is not correspond to direction of destination or its traffic condition is very poor, $B_{ik}$ takes 1,which represents that drivers are not conservative at all; when the current lane is correspond to direction of destination and its traffic condition is not very poor, $B_{ik}$ takes 10,which represents that drivers' tendency of staying on the current lane is ten times larger than before and the conservatism reaches the highest level. It correspond to the definition in 3.2.1 and the psychology of drivers in reality.

### 3.2.5 Physical limit coefficient of lane change

Lane change is limited by two major physical factors. One is the distance between the target lane and the current lane. The farther the distance between the target lane and the current lane is, the more difficult for drivers to change lane to another lane within a Road Unit it is. So according to number of simulation experiments and everyday experience, we get an array $C_x$, which describes the physical limit based on the distance between the current lane and the target lane, where $x$ is the distance between the current lane and the target lane $|i - j|$.Considering that it is very hard for drivers to change lane for many times within a Road Unit, we list the specific values as below table:

| LANE DISTANCE | 0 | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|---|
| C | 1.000 | 0.900 | 0.100 | 0.040 | 0.015 | 0.006 |

**Table 4: The values of C array**

Another physical limit is the traffic condition of lanes between the current lane and the target lane. If there are two lanes or more between the current lane and the target lane, then vehicles changing lane will pass by other lanes in the middle. If the traffic condition of these lanes are poor, it is very difficult to change lane. Because the difficult degree of lane change relies largely on the Road Unit with the worst traffic condition passing by (i.e., bottleneck). In this condition, the short-sightedness model is not applicable, so we modify the Equation 13, and obtain the physical limit coefficient matrix of traffic condition $H_{ij}$:

**Equation 18: Physical limit coefficient of lane change**

$$H_{ij} = \begin{cases} 1, (|j - i| \le 1) \\ t_{f_{\frac{i+j}{2}}}, (|j - i| = 2) \\ \frac{1}{2} \cdot t_{f_{kmin}} \cdot \left(1 + \frac{1}{|j - i| - 2} \cdot \sum_{k \in K, k \ne kmin} t_{f_k}\right), (|j - i| > 2) \end{cases}$$

Where, $K$ is a Road Unit set that between lane $i$ and $j$.When $|j - i| \le 1$, there is no middle lanes in the process of lane change, so H takes 1. When $|j - i| = 2$, there is only one middle lane in the process of lane change; $t_{f_{\frac{i+j}{2}}}$ represents the middle Road Unit. When

$|j-i|>2$, $t_{f_{kmin}}$ represents the Road Unit with the worst traffic condition (i.e., the most important bottleneck of lane change); $\frac{1}{2}\cdot\left(1+\frac{1}{|j-i|-2}\cdot\sum_{k\in K,k\neq kmin}t_{f_k}\right)$ represents the additional effect caused by other Road Unit.

### 3.2.6 Psychological tendency coefficient of drivers

On the basis of the above conclusions, we can calculate psychological tendency coefficient of drivers.

**Definition 13: Psychological tendency coefficient of drivers**

$$F_{ijk}=G_{ijk}\cdot C_{|i-j|}\cdot H_{ij}$$

Rationality of this definition is obvious: $C$ and $H$ are both numbers within the range of [0, 1], representing the overall negative impact of physical limit on the attraction. In addition, for convenience, psychological tendency coefficients can be standardized:

**Equation 19: Psychological tendency coefficient of drivers after standardization**

$$T_{ijk}=\frac{F_{ijk}}{\sum_{r=1}^{m}F_{irk}}$$

Thus, there must be:

$$\sum_{r=1}^{m}T_{irk}=1$$

$T$ keeps the psychological tendencies proportion of vehicles starting from the same Road Unit, heading towards the same direction, simplifying flow distribution process in 3.3.

## 3.3 Flow distribution rules

Flow distribution rules are both the restrictive conditions for the mentioned in the theoretical framework and the process of flow distribution in Generalized Intersections. So, we definite 2 kinds of volume matrix.

**Definition 14:**

Volume matrix $Q_{F_{ij}}$ indicates the number of vehicles per unit times into the output side Road Unit $j$ from the input side Road Unit $i$.

Volume matrix $Flow_{ijk}$ represents the number of vehicles that the direction of travel is k in the input terminal Road Unit i entering the output terminal Road Unit j.

### 3.3.1 Output flow assumption

For an input side Road Unit group, we calculate the vehicle left the Road Unit output volume according to their parameter of queue part and free part. In the process of the vehicle from a set an input side Road Unit group enter an output side Road Unit group, if look at only one Road Unit group, there is little correlation between these Road Units, then we can get:

**Assumption 7:**

Instantaneous output number of vehicles in the input side Road Unit in accordance with a proportion to output volume.

Matrix $O_{R_{ik}}$ represents the proportion coefficient of instantaneous output volume of the vehicles with the direction of $k$ in the input side Road Unit $i$. Based on this assumption, we can obtain the matrix $O_{R_{ik}}$ based on the proportion matrix $R_{ik}$ of the output volume $K_r$ and input side Road Unit's vehicle direction, the formula is as follows:

**Equation 20:**

$$O_{R_{ik}} = K_{r_i} \cdot R_{ik}$$

### 3.3.2 Psychological tendency assumption

In Section 3.2, according to the attraction model, we can calculated the proportion of psychological tendencies that each vehicle direction of destination in input of each Road Unit go to the output of each Road Unit. Psychological tendency coefficient can actually be seen as the probability of vehicles enter the output of each Road Unit. We can therefore obtain:

**Assumption 8:**

It is proportional for the number of instantaneous vehicles of the same Road Unit and the same direction of the input side Road Unit that enter various output side Road Units based on the mental tendency coefficient.

At this point, we can calculate the basic proportional matrix of our expected flow distribution according to the above assumptions. Assuming that volume proportional matrix $Q_{R_{ij}}$ represents the proportional coefficient of the instantaneous volume from the input side Road Unit $i$ into the output Road Unit $j$ .According to this Assumption 8, we can calculate the matrix $O_{R_{ik}}$ according to tendency proportion $T_{ijk}$ and matrix $Q_{R_{ij}}$.Formula is as follows:

**Equation 21:**

$$Q_{R_{ij}} = \sum_{k=1}^{kmax} O_{R_{ik}} \cdot T_{ijk}$$

### 3.3.3 Restrictions: output and capacity

Obviously, the total output of the vehicle of the input side Road Unit does not exceed its largest output $I_i$, The total number of vehicles entering the output side Road Unit does not exceed the largest capacity $O_j$, therefore, the flow of volume matrix size is limited by the output of input side Road Unit and capacity of output side Road Unit.

**Inference 3:**

$$\begin{cases} OutFlow_i = \sum_{k=1}^{m} Flow_{ik} \le I_i \\ InFlow_j = \sum_{k=1}^{n} Flow_{kj} \le O_j \end{cases}$$

### 3.3.4 Traffic bottlenecks and its randomness

When the input Road Unit is still capable of outputting a vehicle, the output Road Unit is still able to accommodate the road vehicle, we have enough reason to adopt Assumption 6, i.e., with the assumption that the volume is constant. According to the assumption in 3.3.1 and 3.3.2, we calculate the volume proportion matrix $Q_{R_{ij}}$, and distribute flow according to this proportion, when the number of output vehicles of an input side Road Unit is equal to its largest output (i.e., this Road Unit has no vehicles into the next Road Unit group), or the number of the vehicles that go into the RU of the output side Road Unit is equal to its largest capacity (i.e., vehicles in input side Road Unit cannot enter this Road Unit), the vehicle that can leave the input side Road Unit and into the output side Road Unit will change its output volume and its tendency proportion, also the volume proportional matrix $Q_{R_{ij}}$ changes accordingly, we call it traffic bottlenecks.

Every time when traffic bottlenecks are achieved, we need to push the vehicles at this stage into the output side Road Unit, make the number of vehicles join the volume matrix $Q_{F_{ij}}$, recalculate volume matrix $Q_{R_{ij}}$, then follow the new volume proportion of matrix to calculate the next stage, until the next traffic bottlenecks. If the input side can no longer output, the output side RU is run out of capacity, or matrix $Q_R$ is empty, then the process of flow distribution is terminated.

When reach the traffic bottleneck due to an output vehicles of input site RU equal to its maximum output, the Road Unit had no effect on the output volume and the tendency proportion of other input side RU; When reach the traffic bottleneck due to the number of vehicles that go into the output side RU equal to its largest capacity, some input Road Unit will have probability of being blocked, because the vehicle have no way to enter the output Road Unit, blockage probability is determined by the number of remaining vehicles of this type of vehicles.

Considering the randomness of the flow bottleneck, the vehicles must be regarded as discrete. If and only if the number of the discrete remaining vehicle reaches 1, the rest of the vehicles will be blocked. In approximate calculation, we design the obtained number of the continuous remaining vehicles as $x$, the number of the discrete remaining vehicles as $x_D$, then the probability of the discrete number of remaining vehicles reaches 1 is:

$$P(x_D \geq 1) = \begin{cases} \frac{1}{2} \cdot x, (x < 2) \\ 1, (x \geq 2) \end{cases}$$

Probability of remaining blocked vehicles are in the above formula.

### 3.3.5 Example: calculation of a volume matrix

Considering state transition rules of the Road Unit in the second part, driver mental model and flow distribution model, we have been able to proceed from the most basic parameters of Road Units $\{L, k_j, L_f, L_q, k, u\}$, calculates the volume matrix that the model required. The following is an example of a volume matrix calculation. In this calculation process, we need to enter:

The number of input Road Unit $n$, the number of output Road Unit $m$, the number of direction of destination $K_{max}$, unit time $t_{unit}$, vehicle density when completely blocked $kj$, the connectivity between the input side and output side $map$ (the Generalized Intersection representative bipartite graph adjacency matrix), the number each input side and output side queue part of vehicles $s_q$, the number of vehicles of free part $s_f$, each end of direction of destination corresponding to the left road $Ks$ and the right road $Ke$, The proportion of each road vehicle's direction of destination $R$, distance from the current road to the mandatory lane change line $d$, and limited speed $u_f$.

The following is an example of calculated by a program:

**Input data:**

Basic data:

$$n = 3, m = 3, K_{max} = 3, d = 300m, u_f = 15m \cdot s^{-1}, k_j = 0.15v \cdot m^{-1}, t_{unit} = 5s$$

The data shows that: Our example is a three-lane road with 3 direction of destination; the distance to the mandatory lanes change line is $300m$; limited speed is 15 $m \cdot s^{-1}$; density when completely blocked is $0.15v \cdot m^{-1}$; Unit time is $5s$. So we get the length of RU is: $u_f \cdot t_{unit} = 75m$.

Data of input part:

| i | 1 | 2 | 3 |
|---|---|---|---|
| $\boldsymbol{s_q}$ | 0.0 | 0.6 | 5.5 |
| $\boldsymbol{s_f}$ | 1.4 | 2.7 | 3.1 |

Data of output part:

| j | 1 | 2 | 3 |
|---|---|---|---|
| $s_q$ | 0.0 | 1.5 | 6.5 |
| $s_f$ | 1.3 | 2.9 | 1.7 |

Data of direction of destination:

| | 1 | 2 | 3 |
|---|---|---|---|
| Ks | 1 | 2 | 3 |
| Ke | 1 | 2 | 3 |

Data of proportion of each road vehicle's direction of destination $R$:

| i\k | 1 | 2 | 3 |
|---|---|---|---|
| 1 | 0.57 | 0.19 | 0.24 |
| 2 | 0.08 | 0.66 | 0.26 |
| 3 | 0.00 | 0.03 | 0.97 |

Data of the connectivity between the input side and output side $map$:

| i\j | 1 | 2 | 3 |
|---|---|---|---|
| 1 | True | True | True |
| 2 | True | True | True |
| 3 | True | True | True |

**Pretreatment:**

According to the model of 2.3, we calculate the largest output of each input side $I$, the largest capacity of each output side $O$, the average output volume of each input side $k_r$ and the traffic condition coefficient of each output side $t_f$.

Input side:

| I | 1 | 2 | 3 |
|---|---|---|---|
| I | 1.2258 | 1.6021 | 2.4495 |
| $k_r$ | 0.2452 | 0.3024 | 0.4899 |

Output side:

| J | 1 | 2 | 3 |
|---|---|---|---|
| O | 7.7162 | 4.5594 | 2.7744 |
| $t_f$ | 0.9674 | 0.7627 | 0.2713 |

We can calculate the psychological coefficient $T$ of the driver after standardization through the attraction model of 3.2:

| k | 1 | | | 2 | | | 3 | | |
|---|---|---|---|---|---|---|---|---|---|
| i\j | 1 | 2 | 3 | 1 | 2 | 3 | 1 | 2 | 3 |
| 1 | 0.9470 | 0.0527 | 0.0004 | 0.8144 | 0.1844 | 0.0012 | 0.6864 | 0.2984 | 0.0152 |
| 2 | 0.4274 | 0.5242 | 0.0484 | 0.0858 | 0.9029 | 0.0114 | 0.1891 | 0.7109 | 0.1001 |
| 3 | 0.0684 | 0.7266 | 0.2050 | 0.0616 | 0.6538 | 0.2846 | 0.0611 | 0.6484 | 0.2906 |

**Flow distribution calculation:**

We can obtain volume matrix $O_R$ by calculating Equation 20.

| i\k | 1 | 2 | 3 |
|---|---|---|---|
| 1 | 0.1398 | 0.0466 | 0.0588 |
| 2 | 0.0242 | 0.1996 | 0.0786 |
| 3 | 0.0000 | 0.0147 | 0.4752 |

After we calculate $O_R$ and $T$, according to the approach of traffic bottleneck in 3.3.4 by repeatedly calculation until all road units cannot output flow or the given unit time is running out. By this, it is easy to obtain the final volume matrix $Flow_{ijk}$ through proportional relations:

| k | 1 | | | 2 | | | 3 | | |
|---|---|---|---|---|---|---|---|---|---|
| i\j | 1 | 2 | 3 | 1 | 2 | 3 | 1 | 2 | 3 |
| 1 | 0.6616 | 0.0368 | 0.0003 | 0.1897 | 0.0429 | 0.0003 | 0.2019 | 0.0878 | 0.0045 |
| 2 | 0.0548 | 0.0672 | 0.0062 | 0.0907 | 0.9547 | 0.0120 | 0.0787 | 0.2961 | 0.0417 |
| 3 | 0.0000 | 0.0000 | 0.0000 | 0.0045 | 0.0480 | 0.0209 | 0.1451 | 1.5406 | 0.6904 |

We merge matrix $Flow_{ijk}$ and obtain the volume matrix $Q_{F_{ij}}$:

| **i\j** | 1 | 2 | 3 |
|---|---|---|---|
| 1 | 1.0533 | 0.1675 | 0.0050 |
| 2 | 0.2242 | 1.3180 | 0.0599 |
| 3 | 0.1496 | 1.5886 | 0.7113 |

In the above example $d = 300m$, distance to intersection closely, tendency to choose the path of the driver is determined mainly by destination. While the traffic condition of lane 3 is very poor, so drivers who are on lane 3 have strong tendency to leave it. From the standardized driver's psychological coefficient $T$ can be seen: on lane 1 and lane 2, vehicles which are on the right lane (i.e., the current lane is the same as the direction of destination) remain unchanged; the vehicles which are not on the right lane have begun to change the lane. However, Drivers who are on lane 3 are going to leave lane 3 as far as possible wherever they want to go. Because three output Road Units are all have good capacity, output volume can reach the lane they want to get. From the volume matrix data $Flow_{ijk}$ we can see the same input and the same final turn of the final output vehicle is agree with the proportion of $T$, it is more in line with the actual situation.

# 4 Other details of this model

## 4.1 Random blockage model in Road Units

### 4.1.1 The definition of random blockage

In order to solve the truck blockage problem mentioned in Assumption 2, introduce the random blockage on Road Units. Random blockage is a random process that compulsively lower the capacity, output volume and speed limits of an RU or some more RUs. This process lowers these "shown" parameters directly, without affecting the "hidden" parameters such as queue part length, free part density and so on.

For a Road Unit with output capacity $I$, carrying capacity $O$ and speed limit $u_f$, we have:

**Equation 22: The basic equation of random blockage**

$$\begin{cases} I_1 = I \cdot (1-x) \\ O_1 = O \cdot (1-y) \\ u_{f_1} = u_f \cdot (1-z) \end{cases}$$

Where $x$, $y$, $z$ are random variables, whose distribution is according to traffic conditions, driver conditions and other factors, with a value range [0,1].

Taking truck blockage as an example, we add random blockage into the model wherever there're too many trucks. The effect of random blockage in this case is shown as traffic volume reduction, which simulates the effect of a truck's inefficient turning.

In fact, besides truck problem, the random blockage could simulate many situations in traffic system, such as phantom traffic jam, traffic accident, start-up delay, etc.

### 4.1.2 Analyses of phantom traffic jam

In traffic network, especially on city roads, there is an interesting phenomenon that traffic jam does not necessarily occur in the condition that traffic volume is close to saturated flow rate. Often, when the volume is far less than the saturated flow rate, the traffic condition still seems to be very poor. This is the famous "phantom traffic jam".

Many facts tell us that the speed of vehicles on roads obeys the normal distribution approximately (to be more precise, logarithmic normal distribution). Therefore, on roads, there's a small probability that the speed of a vehicle becomes slow enough to affect the vehicle following it. If the distance between the slow vehicle and the vehicle behind is not too long, for the back vehicle, this condition is equivalent to a decrease in speed limit. This effect is quickly passed behind and the vehicles behind begin to stack, which causes a chain reaction so that the phantom traffic jam happens.

According to the analysis above, the phantom traffic jam can be regarded as a temporary decrease in speed limit in a Road Unit. In addition, this effect has a strong negative correlation with the density in the RU (the lower the density is, the longer the average distance is). Therefore, we summarize the random blockage model of phantom traffic jam as follows:

$$\begin{cases} u_{f_1} = u_f \cdot min\{1 - \frac{x}{u_f} \cdot f(k), 1\} \\ ln(x) \sim N(\mu, \sigma^2) \end{cases}$$

Where, $f(k)$ is an increasing function about the density of the Road Unit, while $ln(x) \sim N(\mu, \sigma^2)$ refers to the distribution of vehicle speed. As the probability of phantom traffic jam is related to the distribution of vehicle speed on roads, and the distribution of vehicle speed is related to the mental state of drivers and many other conditions, phantom traffic jam should be analyzed by particular situation. For the formula above, we aren't able to give a more detailed analysis.

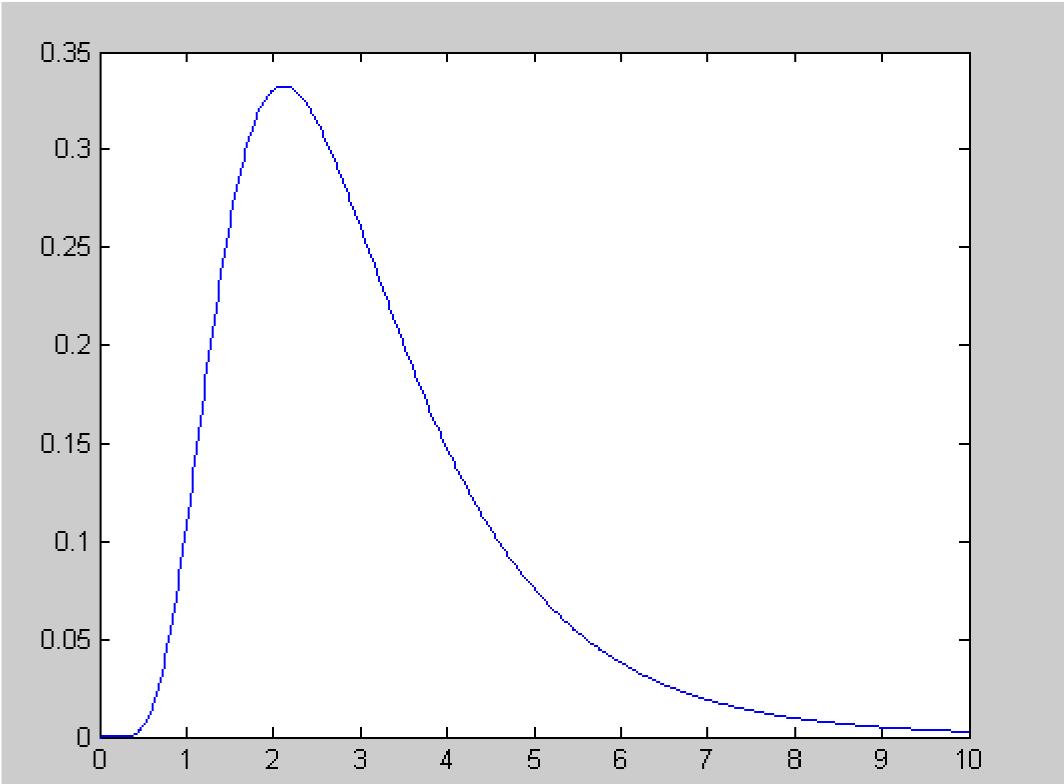


**Figure 25: A typical probability density curve of logarithmic normal distribution**

### 4.1.3 Analyses of traffic accidents

Traffic accident is a very common phenomenon which often causes serious traffic jam. Traffic accident would generally make a lane or several lanes impassable in the accident site. In the random blockage model, we compulsively lower the carrying capacity of the Road Unit which has traffic accidents to 0, as to simulate the fact that drivers never choose the lane which is blocked by traffic accident. As traffic accident site is uneven distributed, which has a far greater probability to happen at an intersection than at other sites, different probability of random blockage should be set according to the different site of Road Units. Also, Different severity of traffic accident has different influence on the traffic flow. In the model, we thought that the accident's impact on traffic (namely, the length of time that capability keeps 0) is positively correlated with the severity of traffic accident. For a certain traffic accident whose degree of severity is $\alpha$, we describe the random blockage model of traffic accidents as follows:

$$\begin{cases} O_1 = 0 \\ t = g(\alpha) \end{cases}$$

Where, $t$ is the length of time that the impact continues, $g(\alpha)$ is a function which is positively correlated with $\alpha$, and the function may be different in different models.

### 4.1.4 Analyses of start-up delay

In the process of queue dissipation, every driver has a certain reaction time after seeing vehicle in front of it begin to start. This reaction time has no influence on vehicle ahead, but it will has an impact on the queue behind. It will cause the vehicles behind to start with delay, so that the time of queue dissipation become longer.

Reaction time varies among drivers, but it obey normal distribution approximately (to be more precise, logarithmic normal distribution). Those drivers whose reaction time is very long increases the waiting time for the vehicles behind. This kind of waiting time can add up, so it is obvious that the longer the queue is, the greater the delay will be. This will lead to increase in the whole queue dissipation time eventually. It should be noted that drivers reacting faster also exist. However, driver who reacts faster can't lead drivers behind to follow it faster. So those who react slower have more influence on the traffic (the so-called "bottleneck effect").

In the analysis, we know that queue delay doesn't always happen, but has a certain probability. The probability increases with the increase of queue length. Therefore, the random blockage model of start-up delay can be regarded as the decrease in output volume, and both the decrease degree and probability are positively related to the reacting time of drivers in the queue. Random block model describing this process is as follows:

$$\begin{cases} I_1 = I \cdot (1 - min(\frac{x}{t_{unit}}, 1)) \\ ln(x) \sim N(\mu, \sigma^2) \end{cases}$$

Where, $ln(x) \sim N(\mu, \sigma^2)$ describes the distribution of reacting time of a driver, $\mu$ and $\sigma$ are closely related to driver's condition. Similar to 4.1.2, this condition largely depends on the traffic on the roads and many other factors, so we can't give further analysis.

## 4.2 Traffic flow on actual intersections

It has been already discussed in 3.2 about the effects of mandatory lane change line and channelization on the behavior of drivers. Therefore, here we are going to discuss the traffic flow on actual intersections qualitatively.

### 4.2.1 Reset of direction on actual intersection

According to Corollary 1, the proportion of vehicles going in every direction of destination on an actual intersection is constant and same everywhere. It is important to note that this corollary applies to the whole intersection rather than a single Road Unit. According to this corollary, after going through the actual intersection, the proportion of vehicles to all directions will be reset as the default proportion of the next actual intersections ahead.

### 4.2.2 Flow distribution on actual intersections

Due to the mandatary of channelized area, traffic flow is strictly limited by traffic rules on actual intersection. On the basis of this limitation, we can also use the flow distribution rules on Generalized Intersections. However, we need to pay attention to the following:

1) According to 4.2.1, on actual intersections, direction of destination of traffic flow has been reset as the direction in the next actual intersections.

2) On next intersections, the arrangement of each Road Unit no longer has a simple order. Therefore, coefficients used in the calculation of attraction, the conservatism coefficient and the physical limit coefficient will be more complex according to the actual arrangement situation.

3) Due to the significant difference of the actual structure between an actual intersection and an ordinary Generalized Intersection, the physical limit coefficient in 3.2.5 need to be modified according to the specific situation.

### 4.2.3 The original source and final destination of traffic flow

It’s obvious that vehicles in this model do not appear or disappear suddenly. They need special actual intersections to work as the source point and destination of the traffic network.

**Definition 15:**

Actual intersections which input flow into the traffic network without input part is called “source intersection”; Actual intersections which work as the final exit in the traffic network without output part is called “destination intersection”.

Traffic flowing in from the source intersection has a default proportion of direction of destination, but obviously there’s no such proportion at destination intersection, since the proportion of direction of traffic flow there is greatly affected by traffic conditions in the whole network.

# 5 Evaluation and practical applications

## 5.1 Strengths of this model and comparison with other models

### 5.1.1 Comparison with micro models

Microscopic models, such as many kinds of traffic simulation software, simulate each drivers' psychology as well as the interaction of the vehicles. SimTraffic, for instance, doesn't perform well when simulating. When simulating a crowded road which is $3km$ long, SimTraffic can only simulate about 10 seconds of the virtual traffic flow in one actual second. However, based on Road Units, the calculation in our model is greatly simplified, and occupies less resources. An ordinary PC can simulate at least 20,000 Road Units. Considering a common setting that an RU is $75m$ in length, a PC can simulate $1{,}500km$ of traffic network.

Important weakness of this model is its natural continuity, and for this reason it loses the discreteness of traditional micro model which is closer to reality. So it cannot position an actual vehicle. This is a regrettably disadvantage to some small scale and detailed analysis, and this is a possible direction of improvement.

### 5.1.2 Comparison with macro models

Macro models generally consider traffic flow as waves, ignoring lane change, different properties of vehicles and mental state of drivers. These models are often rough, and most details are hidden. In out model, lane change has been carefully modeled, and psychology of drivers is considered based on the interaction of Road Units, making our simulation more practical without too much loss of efficiency.

## 5.2 The practical applications of this model

Here we introduce the practical application of our model. With limited level and time, we haven't constructed a complete traffic network simulation program yet. However, both the program calculating Road Units' state transition and the program calculating Generalized Intersections' flow distribution has been completed. In 5.2.1, these program will be used to simulate a segment of actual road. Besides, other possible applications will be introduced in 5.2.3.

### 5.2.1 A practical simulation: the simulation of a tow-lane road

To evaluate our model, we decide to find a segment of roads and build a model of it. For convenience, we selected Xiadu Road, which is close to our school, as our subject of modeling.

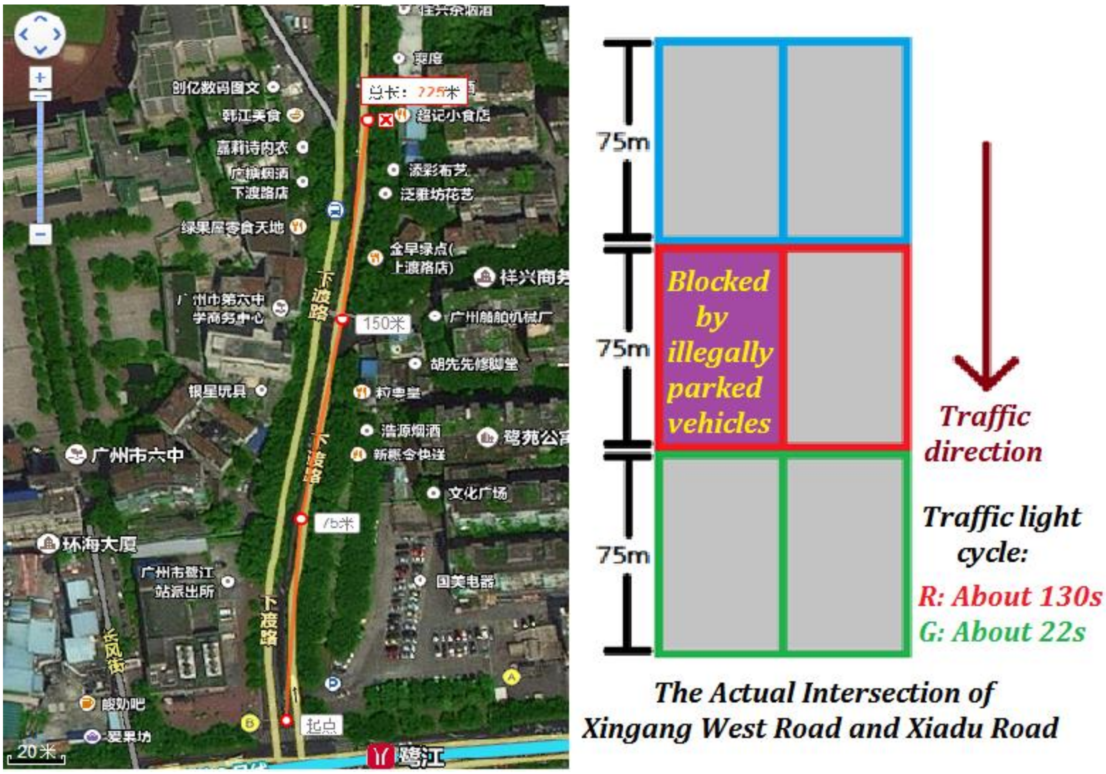


**Figure 26: Map and abstract graph of this segment of road**

As shown above, the starting point is the intersection of Xiadu Road and Shangdu Road, while the end point lies about 225 meters away down the road at the Intersection of Xingang West Road and Xiadu Road. The intersection of Xiadu Road and Shangdu Road has no traffic lights, while the Intersection of Xingang West Road and Xiadu Road has a traffic light, whose cycle about $152s$, where the red phase is about $130s$ and the green phase about $22s$.

This segment is a two-lane road with a complex traffic condition. It should be especially noted that, the middle part of the right lane of the road (in Figure 26, it's on the left) is blocked by illegally parked cars and there're also many taxis picking-up and dropping-off is this area.

Our simulation focuses on the $22s$ green phase. Because of the road's speed limit is not known, we assume it as the commonly used speed limit: $u_f = 15m \cdot s^{-1}$, and define unit time $t_{unit} = 5s$ in the simulation. In this way, we need to simulate 5 units of time. As for the length of Road Units, define that $L_U = u_f \cdot t_{unit} = 75m$, so that the $225m$ segment is divided into 3 of Road Units.

The first thing to do is to collect traffic data in the green phase. Then, according to the data of the input flow from behind, we use our model to predict the traffic condition, and compare the calculated result with the actual recorded data to assess the effectiveness of this model (see Appendix for the detailed data).

### 5.2.2 Result and analysis of the simulation

The graphs below are comparisons between the calculated data and actual data of the two most front Road Units (shown in green boxes in Figure 26). The "L" and "R" means the left RU (on the right in Figure 26) and the right RU (on the left in Figure 26). For the data of the other 4 Road Units, see Appendix.

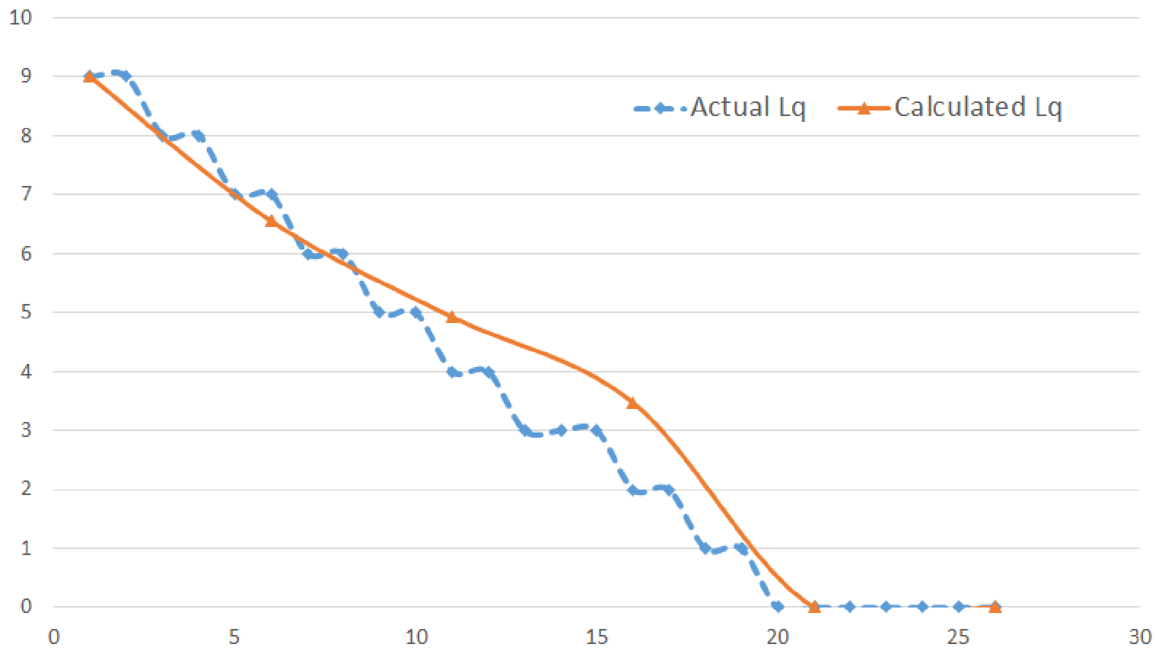


**Figure 27: The comparison of the number of vehicles in queue part in the left RU**

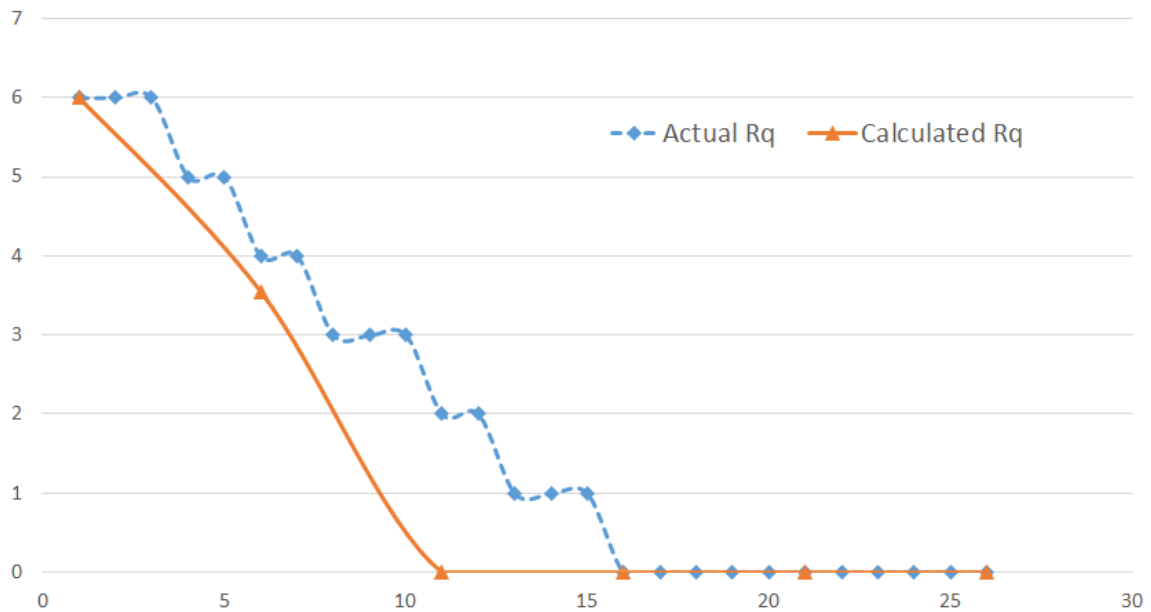


**Figure 28: The comparison of the number of vehicles in queue part in the right RU**

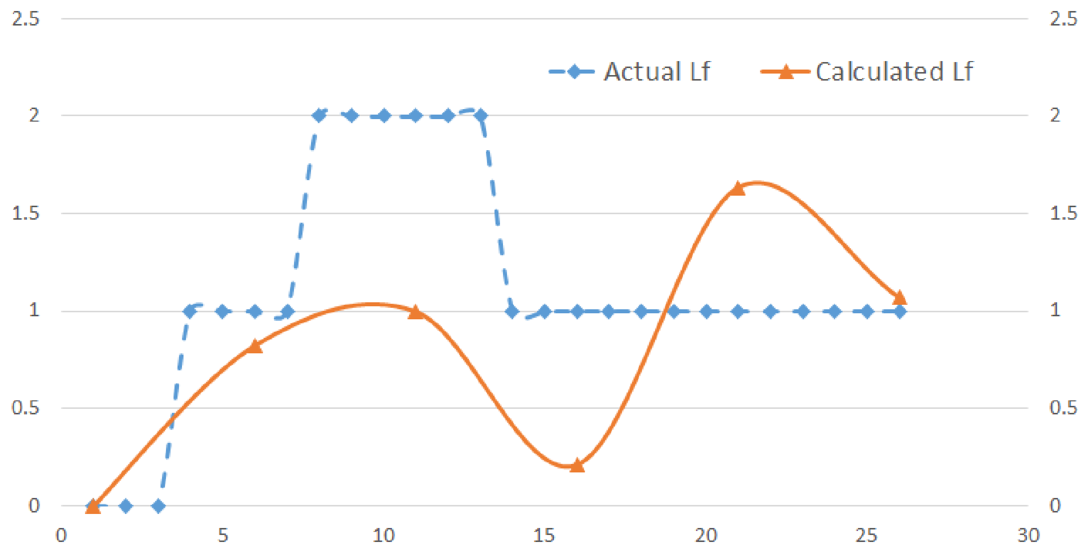


**Figure 29: The comparison of the number of vehicles in free part in the left RU**

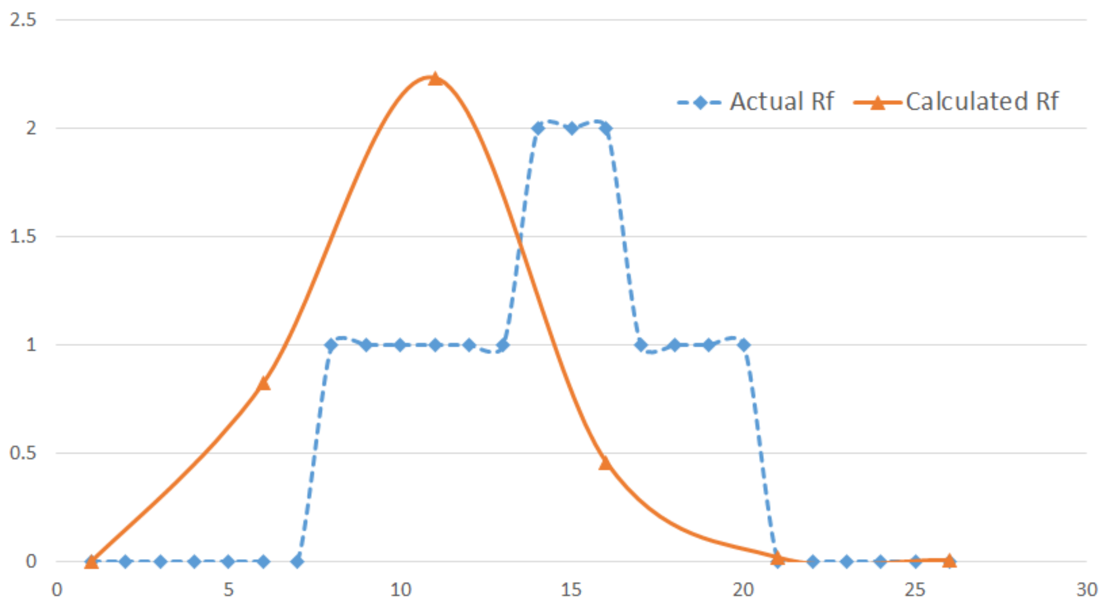


**Figure 30: The comparison of the number of vehicles in free part in the right RU**

From Figure 27 and 28, it obvious that the calculated number of vehicles in the queue part is close to the actual data. As the front end of the two Road Units is an actual intersection, the queue dissipation volume is relatively stable.

In Figure 29 and 30, however, there's an obvious error. In Figure 30, the curve of calculated data leans to the left, and the two curves in Figure 29 have even more differences. The reason is easy to understand: Xiadu Road is relatively narrow and crowded, with many passengers, bikes and taxis, slowing down the traffic flow. Since random blockage is not included in this simulation, the speed in free parts of the Road Units are faster than the actual speed, affecting the simulation in free parts and even, slightly, the queue part of the right RU, shown in Figure 28. Because the parameters in random blockage model are too hard to obtain, we aren't able to build a better model with random blockage.

The result of our simulation is acceptable, as 3 of the 4 calculated curves fit well in the actual situation. However, under some circumstances, this model has an obvious error (like Figure 7). The reason is that traffic network is a system with strong randomness, and that our simulation based on a relatively small sample and a short period of time. In fact, most of mature traffic micro simulation systems on the market, such as VISSIM, CORSIM and SimTraffic, also have obvious errors when simulating the actual situation, with the simulation results vary greatly (the difference in queue length between CORSIM and SimTraffic can go up to 54% [7]) among different software. Considering this, the result of our simulation can be seen as satisfying.

### 5.2.3 Other possible applications of this model

High efficiency is one major advantage of our model. It can simulate and predict the traffic more quickly. So, if combined with OD matrix prediction, this model can be used in GPS navigation systems to provide suggestions on possible routes, helping drivers avoid traffic jams.

On the other hand, if the traffic simulation system based on this model is finished, it can be used to evaluate and optimize city traffic network.

## 5.3 Possible improvement of this model

In order to enhance the accuracy of this model's prediction, we put forward the following improvements:

1. Under the condition of not changing the entire model, many parameters are worthy being adjusted:

1) The different simulation site can lead to different vehicle types passing the road. For example, in logistics zone, trucks appears more often; however, in CBD areas, there're almost only cars and business SUV. This makes the average vehicles length different, so adjustment is needed according to actual situation.

2) Different vehicle acceleration leads to difference in efficiency in queue dissipation. This model gives a formula to calculate the dissipation efficiency, but it only assess the driving condition under one situation. In practice, we can calculate the average acceleration from vehicle data provided by the transport department, and the accuracy of the prediction based on big data analysis will be improved.

2. The following options are the solutions to improve the model:

1) This model uses random blockage to describe accidents. It can be improved by the analysis of big data, working out places where accidents usually happen according to the distribution rate.

2) This model has not discussed the situation of actual intersections deeply. It can be refined by introducing a new way of Road Unit Organization in relatively complicated or bigger intersections, which makes the prediction more precise.

## Summary and Insights

The importance of transportation efficiency is drawing people's attention when the pace of urbanization is accelerating and the population in urban areas is growing. Living in big cities for many years, we three know a lot about the traffic of urban areas. We have lots to say about the jams caused by unreasonable transportation system so we want to build a model about traffic system based on our own ideas to predict the traffic conditions and give advice to drivers on driving route.

Our original idea of studying the traffic system was first inspired by the "Blast Wave Model" build by a team in MIT. After reading related books and papers, we came up with the idea of dividing roads into smaller parts. From this idea the whole model was built with several mathematical tools, including differential equation.

Taking the mental state of drivers into the model is one of the creative parts of our model. Developed from our daily experience, the psychological model makes our model closer to reality. Also, the attraction model is the highlight of our model. It has been carefully adjusted and accurately represents the drivers' mental state.

The idea of random blockage originates from the NaSch cellular automaton model, bet we extended the original idea and made it universal and interesting.

We have known that traffic system is huge and complicated. While building the model, we met lots of problems which are beyond our expectation, like solving differential equations and building up a more precise attraction model. When dealing these problems, we need to use thinking modes and angles which are different from those we used in high school math. When our method is gradually close to the real situation, the joy we get is completely different from that occurring after working out our math homework.

In this research, we try to think of the problems in the drivers' angles. We predict what action the drivers would take facing some traffic situations based on the mental state. Maybe it is the question we have not thought of deeply before this research. The changing of thinking angel deepens our understanding about the driving rules of vehicles and the traffic system.

Our ability is limited and the accuracy of our model is also limited. Through this research, we provide a different way of building models as well as a new thinking mode. Meanwhile, during the research, our ability has been improved and we have acquired knowledge which cannot acquired from the class and daily study.

# References

[1] 邵春福, 魏丽英, 贾斌. 交通流理论[M]. 北京. 电子工业出版社, 2012
[2] 隽志才. 交通系统建模与仿真[M]. 北京. 科学出版社, 2011
[3] 卓金武, etc. MATLAB 在数学建模中的应用[M]. 北京. 北京航空航天大学出版社, 2011
[4] Giordano,F.R, etc.数学建模 (Q, Ye, Trans). 机械工业出版社, 2009
[5] 陈华友, 周礼刚, 刘金培. 数学模型与数学建模[M]. 北京. 科学出版社, 2014
[6] Baidu Library, SYNCHRO 中文简体教程:
http://wenku.baidu.com/link?url=r02DFvvCx3Dj6se1QEtHbJWi1sVzL5G8-EX4xkF2VtPrX9tY91G68kGAP030wUrYa4zrQ3JqOhc-_WOK9V-a6P4KU4mlupHjduMJRrXXfXW
[7] Sina blog, 我应该使用 SimTraffic，还是 CORSIM？:
http://blog.sina.com.cn/s/blog_5df6f1160100hd9e.html
[8] 刘俊利. 汽车驾驶心理学[M]. 济南出版社, 2006
[9] 林松. 驾驶员与乘客心理学[M]. 中国物资出版社, 2011

# Appendix

### 1. Source Code of the program solving Equation 3

Compiled successfully with Free Pascal 2.4.2.

With the linear items deleted, this program can also be used to solve Equation 3.

```
{$n+}
program solving_eq3;
  var
    s,l,k,u,ds,dt,dl,kj,uf,s0,l0:double;
    n,i:longint;
  begin
      assign(output,'solving_eq3.txt');
      rewrite(output);
      readln(l,s,dt,n,uf,kj);
      k:=s/l;
      s0:=s;
      l0:=l;
      writeln('S L K');
      writeln('0 0.0000: ',s:0:4,' ',l:0:4,' ',k:0:4);
      for i:=1 to n do
        begin
            if s<=0.000001 then
              break;
            u:=uf*(1-k/kj);
            ds:=-u*k*dt-0.353*uf*s0*(1-s0/(kj*l0))*(1-s/s0)/l0*dt;
            dl:=ds/kj;
            s:=s+ds;
            l:=l+dl;
            k:=s/l;
            writeln(i,' ',i*dt:0:4,': ',s:0:4,' ',l:0:4,' ',k:0:4);
        end;
      close(output);
  end.
```

**2. Source Code of the program calculating Road Unit state transition**

Compiled successfully with Free Pascal 2.4.2.

```
{$n+}
program RUP_calc;
var n,m,tt:longint;
    uf,kj,len,unittime,avglen,g:double;
    inpq,inpf,outpq,outpf,inp,outp,tf:array [1..10] of double;
    l2,s2:array [0..10000] of double;

function min(a,b:double):double;
begin
    if a<b then exit(a)
      else exit(b);
end;

procedure init;
var i:longint;
begin
    readln(uf,kj,len,unittime);
    readln(n,m);
    for i:=1 to n do readln(inpq[i],inpf[i]);
    for i:=1 to m do readln(outpq[i],outpf[i]);
    tt:=10000;
    avglen:=1/kj;
    g:=ln(uf);
end;

procedure inpcalc;
var i,j:longint;
    k,u,qlen,flen,sq,sf,dt,s,inp1,inp2,lnew,snew,unew,knew,l1,l,l0,s0,ds,dl:double;
    flag:boolean;
begin
    dt:=unittime/tt;
    for i:=1 to n do begin
        qlen:=inpq[i]*avglen;
        flen:=len-qlen;
        sq:=inpq[i];
        sf:=inpf[i];
        if sq=0 then begin
          k:=sf/flen;
          u:=uf*(1-k/kj);
          inp[i]:=u*unittime/len*sf;
          writeln('1');
```

```
    continue;
end;
l0:=flen;
s0:=sf;
l2[0]:=0;
s2[0]:=0;
for j:=1 to tt do begin
        if sf<=0.000001 then begin
            s2[j]:=s2[j-1];
            l2[j]:=l2[j-1];
            continue;
        end;
        k:=sf/flen;
        u:=uf*(1-k/kj);
        ds:=u*k*dt+0.353*uf*s0*(1-s0/(kj*l0))*(1-sf/s0)/l0*dt;
        dl:=ds/kj;
        s2[j]:=s2[j-1]+ds;
        l2[j]:=l2[j-1]+dl;
        sf:=sf-ds;
        flen:=flen-dl;
end;
flag:=false;
for j:=1 to tt do begin
        l1:=(0.1898*g+0.4658)*j*dt*avglen;
        l:=l1-l2[j];
        if l>qlen-0.1 then begin
            flag:=true;
            break;
        end;
end;
if flag then begin
    qlen:=inpq[i]*avglen;
    flen:=len-qlen;
    sq:=inpq[i];
    sf:=inpf[i];
    s:=sq+sf;
    inp1:=(0.0949*g+0.2329)*j*dt;
    lnew:=qlen+l2[j];
    snew:=(s-inp1)-(sf-s2[j]);
    knew:=snew/lnew;
    unew:=uf*(1-knew/kj);
    inp2:=min(unew*(unittime-j*dt)/len*(s-inp1),s-inp1);
    inp[i]:=inp1+inp2;
    writeln('2 ',lnew:0:4,' ',snew:0:4,' ',knew:0:4,' ',j*dt:0:4,' ',inp1:0:4,' ',inp2:0:4);
```

```
            end
               else begin
                        inp[i]:=(0.0949*g+0.2329)*unittime;
                        writeln('3');
                    end;
        end;
end;

procedure outpcalc;
var i:longint;
        qlen,flen,sq,sf,t0:double;
begin
        for i:=1 to m do begin
            qlen:=outpq[i]*avglen;
            flen:=len-qlen;
            sq:=outpq[i];
            sf:=outpf[i];
            t0:=(kj*flen-sf)/(0.0949*g+0.2329);
            if unittime<=t0 then outp[i]:=-(0.0475*g+0.1165)/(kj*flen)*uf*sqr(unittime)+(1-
sf/(kj*flen))*uf*unittime
               else outp[i]:=uf*(kj*flen-sf)*(1-sf/(kj*flen))/(0.1898*g+0.4657);
            outp[i]:=min(flen,outp[i]);
            outp[i]:=outp[i]*(kj-sf/flen);
        end;
end;

procedure tfcalc;
var i:longint;
        qlen,flen,sq,sf,u,k:double;
begin
        for i:=1 to m do begin
            qlen:=outpq[i]*avglen;
            flen:=len-qlen;
            sq:=outpq[i];
            sf:=outpf[i];
            k:=sf/flen;
            u:=uf*(1-k/kj);
            tf[i]:=0.25*(cos(pi*(1-u/uf))+1)*(cos(pi*sq/(len*kj))+1);
        end;
end;

procedure print;
var i:longint;
begin
```

```
	for i:=1 to m do begin
		write(tf[i]:0:4);
		if i<>m then write(' ');
	end;
	writeln;
	for i:=1 to n do begin
		write(inp[i]:0:4);
		if i<>n then write(' ');
	end;
	writeln;
	for i:=1 to m do begin
		write(outp[i]:0:4);
		if i<>m then write(' ');
	end;
	writeln;
	for i:=1 to n do begin
		write(inp[i]/unittime:0:4);
		if i<>n then write(' ');
	end;
	writeln;
end;

begin
	assign(input,'RU C input.txt');
	reset(input);
	assign(output,'RU C output.txt');
	rewrite(output);
	init;
	inpcalc;
	outpcalc;
	tfcalc;
	writeln;
	print;
	close(input);
	close(output);
end.
```

### 3. Source Code of the program calculating Road Unit state transition

Compiled successfully with Free Pascal 2.4.2.

```
{$n+}
program RUP_recalc;
var n,kmax:longint;
    uf,len,tunit,kj,sout,sin:double;
    ns,nsq,nsf,sq,sf,t0,kn,ln,sn,inp:array [1..10] of double;
    part:array [1..10,1..2] of double;
    fin,fout,r,nr:array [1..10,1..10] of double;
    kind:array [1..10] of longint;

function f(s,l,t:double):double;
var ds,dt,dl,s0,l0,u:double;
    i:longint;
begin
    if t=0 then exit(0);
    dt:=t/10000;
    s0:=s;
    l0:=l;
    for i:=1 to 10000 do begin
        if s<=0.000001 then break;
        u:=uf*(1-s/(kj*l));
        ds:=-u*s/l*dt-0.353*uf*s0*(1-s0/(kj*l0))*(1-s/s0)/l0*dt;
        dl:=ds/kj;
        s:=s+ds;
        l:=l+dl;
    end;
    exit(s0-s);
end;

procedure calc1(x:longint);
var u,tr:double;
begin
    tr:=(1-sout/inp[x])*tunit;
    if tr<=0.000001 then tr:=0;
    u:=uf*(1-sf[x]/(kj*len));
    ns[x]:=sf[x]-sout+sin;
    nsq[x]:=f(sf[x]-sout,len-u*(tunit-tr),tr);
    nsf[x]:=ns[x]-nsq[x];
end;

procedure calc2(x:longint);
var s1,l1,tr:double;
```

```
begin
    tr:=(1-(sout-part[x,1])/part[x,2])*(tunit-t0[x]);
    if tr<=0.000001 then tr:=0;
    ns[x]:=sf[x]+sq[x]-sout+sin;
    nsq[x]:=f(sf[x]+sq[x]-sout,(sf[x]+sq[x]-sout)/kn[x],tr);
    nsf[x]:=ns[x]-nsq[x];
end;

procedure calc3(x:longint);
begin
    ns[x]:=sq[x]+sf[x]-sout+sin;
    nsq[x]:=sq[x]-sout+f(sf[x],len-sq[x]/kj,tunit);
    nsf[x]:=ns[x]-nsq[x];
end;

procedure init;
var i,k:longint;
begin
    readln(uf,kj,len,tunit);
    readln(n,kmax);
    for i:=1 to n do readln(sq[i],sf[i],inp[i]);
    for i:=1 to n do begin
        for k:=1 to kmax do read(r[i,k]);
        readln;
    end;
    for i:=1 to n do begin
        for k:=1 to kmax do read(fout[i,k]);
            readln;
        end;
    for i:=1 to n do begin
        for k:=1 to kmax do read(fin[i,k]);
        readln;
    end;
    for i:=1 to n do begin
        read(kind[i]);
        if kind[i]=2 then readln(ln[i],sn[i],kn[i],t0[i],part[i,1],part[i,2])
           else readln;
    end;
end;

procedure main;
var i,k:longint;
    tt:double;
begin
```

```
    for i:=1 to n do begin
        sin:=0;
        for k:=1 to kmax do sin:=sin+fin[i,k];
        sout:=0;
        for k:=1 to kmax do sout:=sout+fout[i,k];
        case kind[i] of
            1:calc1(i);
            2:if part[i,1]>=sout then calc3(i)
                else calc2(i);
            3:calc3(i);
        end;
        tt:=0;
        for k:=1 to kmax do begin
            nr[i,k]:=(sq[i]+sf[i])*r[i,k]+fin[i,k]-fout[i,k];
            tt:=tt+nr[i,k];
        end;
        if tt=0 then continue;
        for k:=1 to kmax do
            nr[i,k]:=nr[i,k]/tt;
    end;
    writeln(n,' ',kmax);
    for i:=1 to n do writeln(nsq[i]:0:4,' ',nsf[i]:0:4);
    for i:=1 to n do begin
        for k:=1 to kmax do begin
            write(nr[i,k]:0:4);
            if k<>kmax then write(' ');
        end;
        writeln;
    end;
end;

begin
    assign(input,'RU R input.txt');
    reset(input);
    assign(output,'RU R output.txt');
    rewrite(output);
    init;
    main;
    close(input);
    close(output);
end.
```

**4. Source Code of the program calculating flow distribution**

Compiled successfully with Free Pascal 2.4.2.

```
{$n+}
program flow_distribution;
  const
    af:array[0..10]of double=(1,0.9,0.81,0.73,0.66,0.59,0,0,0,0,0);
    c:array[0..10]of double=(1,0.9,0.1,0.04,0.015,0.006,0,0,0,0,0);
    y:double=0.5;
  var
    n,m,kmax:longint;
    d,uf,fx:double;
    dir:array[1..10]of longint;
    r:array[1..10,1..10]of extended;
    tf,inp,outp,kr:array[1..10]of double;
    ks,ke:array[1..10]of longint;
    map:array[1..10,1..10]of boolean;
    w,a,g,f,t,flow:array[1..10,1..10,1..10]of double;
    h,oq:array[1..10,1..10]of double;
    ft:array[0..10,0..10]of double;
    qr:array[0..10,0..10,0..10]of double;
  function fmax(x,y:double):double;
    begin
        if x>y then
          exit(x);
        exit(y);
    end;
  function fx_calc:double;
    var
      x:double;
    begin
        if d>0.000001 then
          begin
              x:=d/uf;
              if x<=10 then
                exit(3/x);
              if x<=20 then
                exit(-0.019*x+0.49);
              if x<=40 then
                exit(-0.004*x+0.19);
              if x<=100 then
                exit(-0.0005*x+0.05);
              exit(0);
          end
```

```
        else
          exit(1000000);
    end;
procedure init;
    var
      i,j,t,t1:longint;
    begin
        readln(n,m,kmax);
        fillchar(map,sizeof(map),0);
        fillchar(flow,sizeof(flow),0);
        randomize;
        for i:=1 to n do
          begin
              read(t);
              for j:=1 to t do
                begin
                    read(t1);
                    map[i,t1]:=true;
                end;
          end;
        for i:=1 to m do
          read(tf[i]);
        for i:=1 to n do
          read(inp[i]);
        for i:=1 to m do
          read(outp[i]);
        for i:=1 to n do
          read(kr[i]);
        for i:=1 to kmax do
          begin
              read(ks[i],ke[i]);
              for j:=ks[i] to ke[i] do
                dir[j]:=i;
          end;
        for i:=1 to n do
          for j:=1 to kmax do
            read(r[i,j]);
        readln(d,uf);
        fx:=fx_calc;
    end;
procedure calc_w_a;
    var
      i,j,k,flag:longint;
    begin
```

```
for k:=1 to kmax do
  for i:=1 to n do
    for j:=1 to m do
      begin
          if (dir[i]=k) and (dir[j]=k) then
            flag:=0
          else
            if (dir[i]<>k) and (dir[j]<>k) then
              if i<j then
                if j<ks[k] then
                  flag:=1
                else
                  flag:=2
              else
                if i>j then
                  if j>ke[k] then
                    flag:=1
                  else
                    flag:=2
                else
                  flag:=1
            else
              if (dir[i]<>k) and (dir[j]=k) then
                flag:=3
              else
                flag:=4;
          if d>=0.000001 then
            case flag of
              0:w[k,i,j]:=0;
              1:w[k,i,j]:=y*abs(i-j)*fx;
              2:w[k,i,j]:=-(0.8+y*abs(i-j))*fx;
              3:w[k,i,j]:=(1+y*abs(i-j))*fx;
              4:w[k,i,j]:=-(1+y*abs(i-j))*fx;
            end
          else
            case flag of
              0:w[k,i,j]:=1000000;
              1:w[k,i,j]:=-1000000;
              2:w[k,i,j]:=-1000000;
              3:w[k,i,j]:=1000000;
              4:w[k,i,j]:=-1000000;
            end;
          a[k,i,j]:=w[k,i,j]+tf[j];
      end;
```

```
  end;
procedure calc_g;
  var
    i,j,k,t,ts:longint;
    afs,max,max1,attr,maxg,sumt,tq,td,delta:double;
    tmp:array[1..10]of double;
  begin
      for k:=1 to kmax do
        for i:=1 to n do
          begin
              maxg:=-1;
              for j:=1 to m do
                begin
                    if i=j then
                      g[k,i,j]:=a[k,i,j]*af[0]
                    else
                      begin
                          if i<j then
                            begin
                                ts:=m-j+1;
                                for t:=j to m do
                                  tmp[t-j+1]:=a[k,i,t]*af[t-j];
                            end
                          else
                            begin
                                ts:=j;
                                for t:=j downto 1 do
                                  tmp[t]:=a[k,i,t]*af[j-t];
                            end;
                          if ts<>1 then
                            begin
                                max:=-1;
                                max1:=-1;
                                sumt:=0;
                                afs:=0;
                                for t:=1 to ts do
                                  begin
                                      max:=fmax(max,tmp[t]/af[t-1]);
                                      sumt:=sumt+tmp[t];
                                      afs:=afs+af[t-1];
                                      if t<>1 then
                                        attr:=0.5*max+0.5*(sumt-max)/afs
                                      else
                                        attr:=max;
```

```
                                    max1:=fmax(max1,attr);
                                end;
                              g[k,i,j]:=max1;
                          end
                        else
                          g[k,i,j]:=tmp[1]/af[0];
                    end;
                  if (i<>j) and (g[k,i,j]>maxg) then
                    maxg:=g[k,i,j];
                  g[k,i,j]:=fmax(g[k,i,j],0);
                  if d<0.000001 then
                    if (j<ks[k]) or (j>ke[k]) then
                      g[k,i,j]:=0;
              end;
            if d>=0.000001 then
              begin
                  if (i>=ks[k]) and (i<=ks[k]) then
                    delta:=0
                  else
                    if i<ks[k] then
                      delta:=ks[k]-i
                    else
                      delta:=i-ke[k];
                  if delta=0 then
                    td:=1
                  else
                    td:=fmax(0,1-(1+y*delta)*fx/0.24);
                  if g[k,i,i]<>0 then
                    tq:=maxg/g[k,i,i]
                  else
                    tq:=100000;
                  if tq<=1.55 then
                    if tq>1.05 then
                      g[k,i,i]:=g[k,i,i]*(1+(9-16*(tq-1.05))*td)
                    else
                      g[k,i,i]:=g[k,i,i]*(1+9*td);
              end;
        end;
  end;
procedure calc_h;
  var
    i,j,t,ts:longint;
    min,sumt:double;
    tmp:array[1..10]of double;
```

```
  begin
      for i:=1 to n do
          for j:=1 to m do
              begin
                  if abs(i-j)<=1 then
                      h[i,j]:=1
                  else
                      if abs(i-j)=2 then
                          h[i,j]:=tf[(i+j) shr 1]
                      else
                          begin
                              if i<j then
                                  begin
                                      ts:=j-i-1;
                                      for t:=i+1 to j-1 do
                                          tmp[t-i]:=tf[t];
                                  end
                              else
                                  begin
                                      ts:=i-j-1;
                                      for t:=j+1 to i-1 do
                                          tmp[t-j]:=tf[t];
                                  end;
                              min:=2;
                              sumt:=0;
                              for t:=1 to ts do
                                  begin
                                      if tmp[t]<min then
                                          min:=tmp[t];
                                      sumt:=sumt+tmp[t];
                                  end;
                              sumt:=(sumt-min)/(ts-1);
                              h[i,j]:=0.5*min*(1+sumt);
                          end;
              end;
  end;
procedure calc_f_t;
  var
    i,j,k:longint;
    sumt:double;
  begin
      for k:=1 to kmax do
          for i:=1 to n do
              begin
```

```
                sumt:=0;
                for j:=1 to m do
                  begin
                      if map[i,j] then
                        f[k,i,j]:=g[k,i,j]*h[i,j]*c[abs(i-j)]
                      else
                        f[k,i,j]:=0;
                      sumt:=sumt+f[k,i,j];
                  end;
                for j:=1 to m do
                  t[k,i,j]:=f[k,i,j]/sumt;
            end;
  end;
procedure calc_oq;
  var
    i,k:longint;
  begin
      for k:=1 to kmax do
        for i:=1 to n do
          oq[k,i]:=kr[i]*r[i,k];
  end;
procedure calc_qr;
  var
    i,j,k:longint;
  begin
      fillchar(qr,sizeof(qr),0);
      for k:=1 to kmax do
        for i:=1 to n do
          for j:=1 to m do
            begin
                qr[k,i,j]:=t[k,i,j];
                qr[k,i,0]:=qr[k,i,0]+qr[k,i,j];
            end;
      for k:=1 to kmax do
        for i:=1 to n do
          begin
              for j:=1 to m do
                if qr[k,i,0]<>0 then
                  qr[k,i,j]:=qr[k,i,j]/qr[k,i,0];
              if qr[k,i,0]<>0 then
                qr[k,i,0]:=1;
          end;
      fillchar(qr[0],sizeof(qr[0]),0);
      for k:=1 to kmax do
```

```
        begin
            fillchar(qr[k,0],sizeof(qr[k,0]),0);
            for i:=1 to n do
              begin
                  qr[k,i,0]:=0;
                  for j:=0 to m do
                    begin
                        qr[k,i,j]:=qr[k,i,j]*oq[k,i];
                        qr[0,i,j]:=qr[0,i,j]+qr[k,i,j];
                        qr[k,0,j]:=qr[k,0,j]+qr[k,i,j];
                        qr[k,i,0]:=qr[k,i,0]+qr[k,i,j];
                    end;
              end;
        end;
  end;
function push_and_mark:boolean;
  var
    i,j,k,mini,tmp:longint;
    min:double;
    minf:boolean;
    k1,k2:array[1..10]of double;
  begin
      for i:=0 to n do
        qr[0,i,0]:=0;
      for j:=1 to m do
        qr[0,0,j]:=0;
      for i:=1 to n do
        for j:=1 to m do
          begin
              qr[0,i,0]:=qr[0,i,0]+qr[0,i,j];
              qr[0,0,j]:=qr[0,0,j]+qr[0,i,j];
          end;
      for i:=1 to n do
        qr[0,0,0]:=qr[0,0,0]+qr[0,i,0];
      if qr[0,0,0]<=0.000001 then
        exit(true);
      for k:=0 to kmax do
        for i:=0 to n do
          for j:=0 to n do
            if (i<>0) or (j<>0) then
              qr[k,i,j]:=qr[k,i,j]/qr[0,0,0];
      qr[0,0,0]:=1;
      min:=1000;
      for i:=1 to n do
```

```
    begin
            if qr[0,i,0]<=0.000001 then
                k1[i]:=100000
            else
                k1[i]:=inp[i]/qr[0,i,0];
            if k1[i]<min then
                begin
                        min:=k1[i];
                        mini:=i;
                        minf:=true;
                end;
    end;
for i:=1 to m do
    begin
            if qr[0,0,i]<=0.000001 then
                k2[i]:=100000
            else
                k2[i]:=outp[i]/qr[0,0,i];
            if k2[i]<min then
                begin
                        min:=k2[i];
                        mini:=i;
                        minf:=false;
                end;
    end;
fillchar(ft,sizeof(ft),0);
for i:=1 to n do
    for j:=1 to m do
        begin
                ft[i,j]:=qr[0,i,j]*min;
                ft[i,0]:=ft[i,0]+ft[i,j];
                ft[0,j]:=ft[0,j]+ft[i,j];
                for k:=1 to kmax do
                    flow[k,i,j]:=flow[k,i,j]+qr[k,i,j]*min;
        end;
for i:=1 to n do
    inp[i]:=inp[i]-ft[i,0];
for i:=1 to m do
    outp[i]:=outp[i]-ft[0,i];
if not minf then
    for i:=1 to n do
        begin
                tmp:=random(10000)+1;
                if tmp/10000<=qr[0,i,mini]*(k1[i]-min)/2 then
```

```
                begin
                    for k:=0 to kmax do
                      for j:=1 to m do
                        qr[k,i,j]:=0;
                    inp[i]:=inp[i]-inp[i]*qr[0,i,mini]/qr[0,i,0]+ft[i,mini];
                end;
              for k:=0 to kmax do
                qr[k,i,mini]:=0;
          end
      else
        for i:=1 to m do
          for k:=0 to kmax do
            qr[k,mini,i]:=0;
      exit(false);
  end;
procedure work;
  var
    i,j,k:longint;
  begin
      calc_w_a;
      calc_g;
      calc_h;
      calc_f_t;
      calc_oq;
      calc_qr;
      while not push_and_mark do
        ;
      writeln('I K:');
      for i:=1 to n do
        for k:=1 to kmax do
          begin
              write(i,' ',k,': ');
              for j:=1 to m do
                write(flow[k,i,j]:0:4,' ');
              writeln;
          end;
  end;
begin
    assign(input,'data.txt');
    reset(input);
    init;
    work;
    close(input);
end.
```

**5. Some important settings in Synchro and SimTraffic**

Synchro:
Lane width: $3.6m$;
Saturated flow rate: $1900v \cdot h^{-1}$;
Vehicle length: $8.0m$;
Allow right turns on red: Yes;
Speen limit: It is different in different cases;
Signal time setting: It is different in different cases;

SimTraffic:
Random seed: 0;
Drivers' parameters: All default;
Vehicles' Parameters: All default;

6. **The original data in 2.2.2 used to study queue dissipation**

1) $u_f = 20m \cdot s^{-1}$

| TIME \ GROUP | 1 | 2 | 3 | AVERAGE |
|---|---|---|---|---|
| **6** | 3 | 3 | 3 | 3.00 |
| **11** | 5 | 5 | 4 | 4.67 |
| **16** | 7 | 7 | 7 | 7.00 |
| **21** | 8 | 9 | 9 | 8.67 |
| **26** | 10 | 10 | 10 | 10.00 |
| **31** | 13 | 12 | 12 | 12.33 |
| **36** | 15 | 15 | 14 | 14.67 |
| **41** | 17 | 16 | 16 | 16.33 |
| **51** | 21 | 21 | 21 | 21.00 |
| **61** | 24 | 25 | 25 | 24.67 |
| **81** | 30 | 32 | 31 | 31.00 |
| **101** | 38 | 38 | 39 | 38.33 |

2) $u_f = 24m \cdot s^{-1}$

| TIME \ GROUP | 1 | 2 | 3 | AVERAGE |
|---|---|---|---|---|
| **6** | 3 | 2 | 3 | 2.66 |
| **11** | 5 | 5 | 5 | 5.00 |
| **16** | 7 | 7 | 7 | 7.00 |
| **21** | 9 | 9 | 9 | 9.00 |
| **26** | 11 | 11 | 11 | 11.00 |
| **31** | 13 | 13 | 14 | 13.33 |
| **36** | 16 | 16 | 16 | 16.00 |
| **41** | 18 | 18 | 17 | 17.67 |
| **51** | 21 | 22 | 22 | 21.67 |
| **61** | 25 | 25 | 26 | 25.33 |
| **81** | 33 | 34 | 35 | 34.00 |
| **101** | 42 | 41 | 42 | 41.67 |

3) $u_f = 30m \cdot s^{-1}$

| TIME \ GROUP | 1 | 2 | 3 | AVERAGE |
|---|---|---|---|---|
| **6** | 2 | 3 | 3 | 2.66 |
| **11** | 5 | 5 | 5 | 5.00 |
| **16** | 7 | 7 | 8 | 7.33 |
| **21** | 9 | 9 | 10 | 9.33 |
| **26** | 12 | 12 | 12 | 12.00 |
| **31** | 14 | 14 | 14 | 14.00 |
| **36** | 16 | 16 | 17 | 16.33 |
| **41** | 18 | 19 | 19 | 18.67 |
| **51** | 23 | 23 | 24 | 23.33 |
| **61** | 27 | 27 | 28 | 27.33 |
| **81** | 36 | 36 | 37 | 36.33 |
| **101** | 45 | 45 | 47 | 45.67 |

4) $u_f = 36m \cdot s^{-1}$

| TIME \ GROUP | 1 | 2 | 3 | AVERAGE |
|---|---|---|---|---|
| 6 | 3 | 3 | 3 | 3.00 |
| 11 | 5 | 5 | 5 | 5.00 |
| 16 | 7 | 7 | 7 | 7.00 |
| 21 | 10 | 9 | 10 | 9.67 |
| 26 | 12 | 12 | 12 | 12.00 |
| 31 | 14 | 14 | 14 | 14.00 |
| 36 | 17 | 17 | 17 | 17.00 |
| 41 | 20 | 19 | 19 | 19.33 |
| 51 | 24 | 24 | 24 | 24.00 |
| 61 | 29 | 29 | 29 | 29.00 |
| 81 | 39 | 39 | 38 | 38.67 |
| 101 | 48 | 48 | 48 | 48.00 |

5) $u_f = 42m \cdot s^{-1}$

| TIME \ GROUP | 1 | 2 | 3 | AVERAGE |
|---|---|---|---|---|
| 6 | 3 | 3 | 3 | 3.00 |
| 11 | 5 | 5 | 5 | 5.00 |
| 16 | 7 | 7 | 8 | 7.33 |
| 21 | 9 | 10 | 10 | 9.67 |
| 26 | 12 | 12 | 12 | 12.00 |
| 31 | 14 | 15 | 14 | 14.33 |
| 36 | 17 | 17 | 16 | 16.67 |
| 41 | 20 | 19 | 19 | 19.33 |
| 51 | 24 | 24 | 24 | 24.00 |
| 61 | 29 | 29 | 28 | 28.67 |
| 81 | 39 | 39 | 39 | 39.00 |
| 101 | 49 | 49 | 49 | 49.00 |

6) $u_f = 48m \cdot s^{-1}$

| TIME \ GROUP | 1 | 2 | 3 | AVERAGE |
|---|---|---|---|---|
| 6 | 3 | 3 | 2 | 2.67 |
| 11 | 5 | 5 | 5 | 5.00 |
| 16 | 7 | 7 | 7 | 7.00 |
| 21 | 10 | 9 | 9 | 9.33 |
| 26 | 12 | 11 | 12 | 11.67 |
| 31 | 14 | 14 | 14 | 14.00 |
| 36 | 17 | 16 | 16 | 16.33 |
| 41 | 19 | 19 | 19 | 19.00 |
| 51 | 23 | 24 | 23 | 23.33 |
| 61 | 29 | 29 | 28 | 28.67 |
| 81 | 37 | 38 | 38 | 37.67 |
| 101 | 48 | 49 | 48 | 48.33 |

7) $u_f = 54m \cdot s^{-1}$

| TIME \ GROUP | 1 | 2 | 3 | AVERAGE |
|---|---|---|---|---|
| **6** | 3 | 3 | 2 | 2.67 |
| **11** | 5 | 5 | 5 | 5.00 |
| **16** | 7 | 7 | 7 | 7.00 |
| **21** | 10 | 10 | 10 | 10.00 |
| **26** | 12 | 12 | 12 | 12.00 |
| **31** | 15 | 15 | 14 | 14.67 |
| **36** | 18 | 17 | 17 | 17.33 |
| **41** | 20 | 20 | 19 | 19.67 |
| **51** | 24 | 24 | 23 | 23.67 |
| **61** | 29 | 29 | 28 | 28.67 |
| **81** | 39 | 38 | 38 | 38.33 |
| **101** | 49 | 48 | 49 | 48.67 |

8) $u_f = 60m \cdot s^{-1}$

| TIME \ GROUP | 1 | 2 | 3 | AVERAGE |
|---|---|---|---|---|
| **6** | 3 | 2 | 3 | 2.67 |
| **11** | 5 | 5 | 5 | 5.00 |
| **16** | 8 | 7 | 7 | 7.33 |
| **21** | 10 | 9 | 9 | 9.33 |
| **26** | 12 | 12 | 11 | 11.67 |
| **31** | 14 | 14 | 14 | 14.00 |
| **36** | 17 | 17 | 16 | 16.67 |
| **41** | 19 | 19 | 19 | 19.00 |
| **51** | 24 | 24 | 23 | 23.67 |
| **61** | 28 | 28 | 28 | 28.00 |
| **81** | 39 | 38 | 37 | 38.00 |
| **101** | 49 | 48 | 47 | 48.00 |

**7. The actual traffic condition data in 5.2.1**

1) Road Unit Group 1 (0~75m):

| TIME | LQ | LF | RQ | RF | L IN | L OUT | R IN | R OUT | TURNLR | TURNRL |
|---|---|---|---|---|---|---|---|---|---|---|
| **0:00** | 9 | 0 | 6 | 0 | | | | | | |
| **0:01** | 9 | 0 | 6 | 0 | | | | | | |
| **0:02** | 8 | 0 | 6 | 0 | | 1 | | | | |
| **0:03** | 8 | 1 | 5 | 0 | 1 | | | 1 | | |
| **0:04** | 7 | 1 | 5 | 0 | | 1 | | | | |
| **0:05** | 7 | 1 | 4 | 0 | | | | 1 | | |
| **0:06** | 6 | 1 | 4 | 0 | | 1 | | | | |
| **0:07** | 6 | 2 | 3 | 1 | 1 | | 1 | 1 | | |
| **0:08** | 5 | 2 | 3 | 1 | | 1 | | | | |
| **0:09** | 5 | 2 | 3 | 1 | | | | | | |
| **0:10** | 4 | 2 | 2 | 1 | | 1 | | 1 | | |
| **0:11** | 4 | 2 | 2 | 1 | | | | | | |
| **0:12** | 3 | 2 | 1 | 1 | | 1 | | 1 | | |
| **0:13** | 3 | 1 | 1 | 2 | | | | | 1 | |
| **0:14** | 3 | 1 | 1 | 2 | | | | | | |
| **0:15** | 2 | 1 | 0 | 2 | | 1 | | | | |
| **0:16** | 2 | 1 | 0 | 1 | | | | 1 | | |
| **0:17** | 1 | 1 | 0 | 1 | | 1 | | | | |
| **0:18** | 1 | 1 | 0 | 1 | | | | | | |
| **0:19** | 0 | 1 | 0 | 1 | | 1 | | | | |
| **0:20** | 0 | 1 | 0 | 0 | | | | 1 | | |
| **0:21** | 1 | 0 | 0 | 0 | | | | | | |
| **0:22** | 1 | 0 | 0 | 0 | | | | | | |
| **0:23** | 1 | 0 | 0 | 0 | | | | | | |

2) Road Unit Group 2 (75~150m):

| TIME | LQ | LF | RQ | RF | L IN | L OUT | R IN | R OUT | TURNLR | TURNRL |
|---|---|---|---|---|---|---|---|---|---|---|
| **0:00** | 0 | 2 | 0 | 0 | | | | | | |
| **0:01** | 0 | 2 | 0 | 0 | | | | | | |
| **0:02** | 0 | 3 | 0 | 0 | 1 | | | | | |
| **0:03** | 0 | 2 | 0 | 0 | | 1 | | | | |
| **0:04** | 0 | 2 | 0 | 0 | | | | | | |
| **0:05** | 0 | 2 | 0 | 0 | | | | | | |
| **0:06** | 0 | 2 | 0 | 0 | | | | | | |
| **0:07** | 0 | 0 | 0 | 0 | | 2 | | | | |
| **0:08** | 0 | 0 | 0 | 0 | | | | | | |
| **0:09** | 0 | 0 | 0 | 0 | | | | | | |
| **0:10** | 0 | 0 | 0 | 0 | | | | | | |
| **0:11** | 0 | 0 | 0 | 0 | | | | | | |
| **0:12** | 0 | 0 | 0 | 0 | | | | | | |
| **0:13** | 0 | 0 | 0 | 0 | | | | | | |
| **0:14** | 0 | 0 | 0 | 0 | | | | | | |
| **0:15** | 0 | 0 | 0 | 0 | | | | | | |
| **0:16** | 0 | 0 | 0 | 0 | | | | | | |
| **0:17** | 0 | 0 | 0 | 0 | | | | | | |
| **0:18** | 0 | 0 | 0 | 0 | | | | | | |
| **0:19** | 0 | 0 | 0 | 0 | | | | | | |
| **0:20** | 0 | 0 | 0 | 0 | | | | | | |
| **0:21** | 0 | 0 | 0 | 0 | | | | | | |
| **0:22** | 0 | 0 | 0 | 0 | | | | | | |
| **0:23** | 0 | 1 | 0 | 0 | 1 | | | | | |

3) Road Unit Group 3 (150~225m):

| TIME | LQ | LF | RQ | RF | L IN | L OUT | R IN | R OUT | TURNLR | TURNRL |
|---|---|---|---|---|---|---|---|---|---|---|
| **0:00** | 0 | 1 | 0 | 0 | | | | | | |
| **0:01** | 0 | 1 | 0 | 0 | | | | | | |
| **0:02** | 0 | 0 | 0 | 0 | | 1 | | | | |
| **0:03** | 0 | 0 | 0 | 0 | | | | | | |
| **0:04** | 0 | 0 | 0 | 0 | | | | | | |
| **0:05** | 0 | 0 | 0 | 0 | | | | | | |
| **0:06** | 0 | 0 | 0 | 0 | | | | | | |
| **0:07** | 0 | 0 | 0 | 0 | | | | | | |
| **0:08** | 0 | 0 | 0 | 0 | | | | | | |
| **0:09** | 0 | 0 | 0 | 0 | | | | | | |
| **0:10** | 0 | 0 | 0 | 0 | | | | | | |
| **0:11** | 0 | 0 | 0 | 0 | | | | | | |
| **0:12** | 0 | 1 | 0 | 0 | 1 | | | | | |
| **0:13** | 0 | 1 | 0 | 0 | | | | | | |
| **0:14** | 0 | 1 | 0 | 0 | | | | | | |
| **0:15** | 0 | 1 | 0 | 0 | | | | | | |
| **0:16** | 0 | 1 | 0 | 0 | | | | | | |
| **0:17** | 0 | 1 | 0 | 1 | | | 1 | | | |
| **0:18** | 0 | 1 | 0 | 1 | | | | | | |
| **0:19** | 0 | 1 | 0 | 1 | | | | | | |
| **0:20** | 0 | 1 | 0 | 1 | | | | | | |
| **0:21** | 0 | 1 | 0 | 1 | | | | | | |
| **0:22** | 0 | 1 | 0 | 1 | | | | | | |
| **0:23** | 0 | 0 | 0 | 1 | | 1 | | | | |

**8. Comparisons of calculated data and actual data in 5.2.2**

Road Unit Group 1 (0~75m):

| TIME | LQ | LF | RQ | RF |
|---|---|---|---|---|
| STATE 0 | 9 | 0 | 6 | 0 |
| STATE 1 | 7 | 1 | 4 | 0 |
| **UF=15** | 6.5505 | 0.8222 | 3.5505 | 0.8222 |
| STATE 2 | 4 | 2 | 2 | 1 |
| **UF=15** | 4.9232 | 0.9960 | 0.0000 | 2.2339 |
| STATE 3 | 2 | 1 | 0 | 2 |
| **UF=15** | 3.4697 | 0.2113 | 0.0000 | 0.4584 |
| STATE 4 | 0 | 1 | 0 | 0 |
| **UF=15** | 0.0000 | 1.6337 | 0.0000 | 0.0191 |
| STATE 5 | 0 | 1 | 0 | 0 |
| **UF=15** | 0.0000 | 1.0666 | 0.0000 | 0.0079 |
| STATE 0 | 9 | 0 | 6 | 0 |

Road Unit Group 2 (75~150m):

| TIME | LQ | LF | RQ | RF |
|---|---|---|---|---|
| STATE 0 | 0 | 2 | 0 | 0 |
| STATE 1 | 0 | 2 | 0 | 0 |
| **UF=15** | 0.0000 | 1.2667 | 0.0000 | 0.0000 |
| STATE 2 | 0 | 0 | 0 | 0 |
| **UF=15** | 0.0000 | 0.2308 | 0.0000 | 0.0000 |
| STATE 3 | 0 | 0 | 0 | 0 |
| **UF=15** | 0.0000 | 0.0054 | 0.0000 | 0.0000 |
| STATE 4 | 0 | 0 | 0 | 0 |
| **UF=15** | 0.0000 | 0.9111 | 0.0000 | 0.0000 |
| STATE 5 | 0 | 1 | 0 | 0 |
| **UF=15** | 0.0000 | 1.0731 | 0.0000 | 0.0000 |
| STATE 0 | 0 | 2 | 0 | 0 |

Road Unit Group 3 (150~225m):

| TIME | LQ | LF | RQ | RF |
|---|---|---|---|---|
| STATE 0 | 0 | 1 | 0 | 0 |
| STATE 1 | 0 | 0 | 0 | 0 |
| **UF=15** | 0.0000 | 0.0889 | 0.0000 | 0.0000 |
| STATE 2 | 0 | 0 | 0 | 0 |
| **UF=15** | 0.0000 | 0.0007 | 0.0000 | 0.0000 |
| STATE 3 | 0 | 1 | 0 | 0 |
| **UF=15** | 0.0000 | 1.0000 | 0.0000 | 0.0000 |
| STATE 4 | 0 | 1 | 0 | 1 |
| **UF=15** | 0.0000 | 0.0000 | 0.8890 | 1.0000 |
| STATE 5 | 0 | 0 | 0 | 1 |
| **UF=15** | 0.0000 | 0.0007 | 0.0000 | 0.0889 |
| STATE 0 | 0 | 1 | 0 | 0 |