A Cellular Automaton Model for Car Traffic with a Form-One-Lane Rule

Yo-Sub Han and Sang-Ki Ko

Department of Computer Science, Yonsei University
Seoul 120-749, Republic of Korea
{emmous,narame7}@cs.yonsei.ac.kr

Abstract. We propose a cellular automaton model that simulates a traffic flow with a junction. We include a 'form-one-lane' rule that decides which car moves ahead when two cars on two different lanes are in front of a junction. We present a fundamental diagram of the proposed model and car distribution examples. We also demonstrate that the proposed model is useful for predicting the real-world traffic flow with a junction.

Keywords: Cellular automata, Car traffic, Traffic junction, Form-one-lane rule.

1 Introduction

Traffic jam is a major problem in most of the major cities in the world. There are several researches that attempt to predict the traffic flow accurately and realistically. One of such approaches is a cellular automaton (CA) model for traffic flow. CA models are intuitive and can simulate a complex behavior with a set of simple CA rules. Wolfram [15] presented a basic one-dimensional CA model for highway traffic flow (R184). Nagel and Schreckenberg [9] proposed another traffic simulation model using CAs, which is a variant of R184 [15]. This model shows a transition from laminar traffic flow to start-stop-waves as the car density increases using Monte-Carlo simulations. Benjamin et al. [1] developed another model (in short, BJH model) that is similar to the Nagel-Schreckenberg (NaSch) model with a 'slow-to-start' rule that reflects the flawed behavior of real drivers. A 'slow-to-start' rule assumes that drivers sometimes lose attentions because of having been stuck in the queue of stopped cars and then start with some delay. Clarridge and Salomaa [2] proposed a 'slow-to-stop' rule and added the new rule into the BJH model. The 'slow-to-stop' rule is based on the following behaviors of drivers: Drivers decelerate before the traffic jam to avoid collision. Note that these models simulate the single-lane highway traffic with one-dimensional CAs. However, in reality, most highways have several junctions where two or more lanes join and they may cause a heavy traffic congestion. See Fig. 1 for example.

Benjamin et al. [1] examined the presence of a junction. They studied the effects of acceleration, disorder and slow-to-start behavior on the queue length at the entrance to the highway. Xiao et al. [17] analyzed a bridge traffic bottleneck

B. Bouchou-Markhoff et al. (Eds.): CIAA 2011, LNCS 6807, pp. 154–165, 2011.

Fig. 1. An example of traffic jam around a junction where several lanes join

based on the R184 model [15,16]. Researchers investigated two-lane or multi-lane traffic simulations using CAs and lane changing rules [6,7,13].

We focus on how to simulate a traffic with a junction and how to compute a maximal traffic flow that does not increase the traffic jam while the traffic density varies. We use two one-dimensional CA arrays with various parameters. This helps us to predict the flux of a junction and the length of traffic congestion in front of a junction and to simulate other possible cases with a junction.

2 CA-Based Traffic Simulation Models

A CA is a collection of cells on a grid that evolves through a number of discrete time steps according to a set of rules based on the states of neighboring cells [16]. The NaSch model [9] is the first nontrivial traffic simulation model based on CAs. There are several papers analyzing this model in detail [8,10,11,12] and modifying the model for better simulations [3,4,5]. The NaSch model is defined on a one-dimensional array with periodic boundary conditions. Each cell may either be occupied by a car or be empty. Each car has an integer velocity with values between zero and v_{max}. For an arbitrary configuration, one update of the system consists of four consecutive steps performed in parallel for all cars. These four steps are acceleration, slowing down, randomization and car motion, and respectively reflect the features of cars on highways.

The BJH model [1] is an extension of the NaSch model [9]. Benjamin et al. [1] noticed that drivers have a possibility of starting slowly when they pull away from being in a static queue of cars. This can arise from a driver's loss of attention as a result of having been stuck in the queue. The BJH model introduces p_{slow} to simulate the driver's behavior stochastically. The p_{slow} is the probability of starting slowly from the static queue of cars. When the velocity of a car is 0 and the distance between the next car is long enough, this car stays at velocity 0 on this time step with probability p_{slow} and accelerates to 1 on the next time step. On the other way, this car may accelerate normally with probability $1 - p_{slow}$. This rule is called a 'slow-to-start' rule.

Lastly, there is one more rule for more realistic traffic simulation, a 'slow-to-stop' rule by Clarridge and Salomaa [2]. They observed that the cars following the previous models behave in an unrealistic fashion when approaching a traffic jam. If a car B ahead has velocity 0, then a car A may drive up to B at velocity v_{max} only to brake down to velocity zero in one time step in the cell right behind B. To make it more realistic, they suggested the addition of a 'slow-to-stop' rule. This rule causes drivers to go slower when approaching jams since drivers would slow down beforehand where a small jam is visible from a distance. Clarridge and Salomaa [2] used this rule to the BJH model and demonstrated that there are fewer long jams with many cars at a complete stop, and instead there appear to be many slowdowns to avoid these situations, which is more realistic than the BJH model.

3 Form-One-Lane Rule Model

We propose new CA transition rules for traffic simulation with a highway junction where two lanes become a single lane.

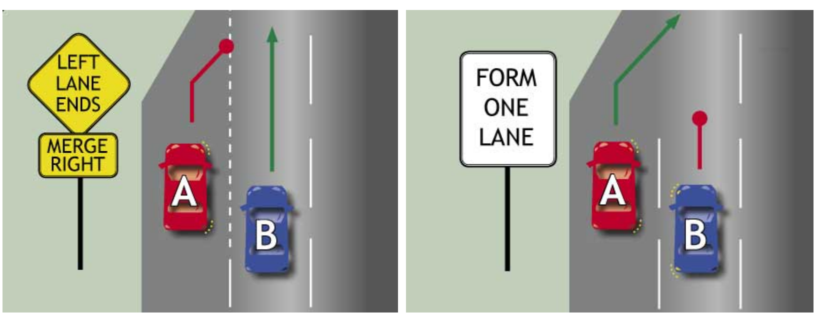


Fig. 2. The left diagram shows a case of merging traffic and the right diagram shows a case of forming one lane. These two cases are the same in the respect to joining two lanes at a junction and becoming a single-lane.

When two lanes join at a junction, there are two types of rules: the first is merging and the second is forming one lane [14]. Merging traffic is where a lane is ending and a driver is required to cross a broken or dotted line to merge with other traffic. In this case, the driver who is about to cross the broken line must give way to traffic in close proximity in another lane regardless of which car is in front. This case is shown in the left side of Fig. 2; car A must give way to car B. The 'form-one-lane' rule requires a driver to give way to a car in another lane if that car is in front of the driver's car when the lanes merge. Hence, "The driver in front has right of way". This case is shown in the right side of Fig. 2; car B must give way to car A. Between these two rules, we consider the second rule, 'form-one-lane', since it gives the same priority to all lanes. This implies that all cars on the road have the same priority regardless of which lane they are in.

In the 'form-one-lane' rule, when there are two cars near a junction, the car that is in front of the car on the other lane goes first. We can adopt this rule to the velocity rule of the BJH model and the 'slow-to-stop' model. In the BJH model, the next velocity of the car is determined based on the current velocity, the maximal velocity and the distance between the next car. The 'slow-to-stop' model uses one more information, the velocity of the next car, for determining the next velocity.

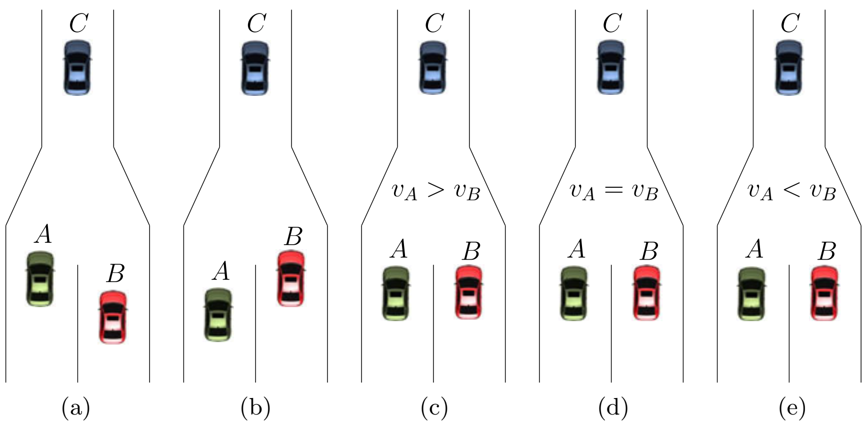


Fig. 3. Five possible cases with a junction where two lanes join. We say that A and B are on the same position in (c), (d) and (e).

There are five possible cases with a junction as illustrated in Fig. 3. In trivial cases, there are two cars approaching to a junction on each lane and they are at different positions. Let L_1 and L_2 be the two lanes that join and make a junction and L_3 be the joined lane after the junction. Let A, B and C be the closest cars to the junction on L_1, L_2 and L_3, respectively. In Fig. 3(a), A goes first because A is in front of B. Let $next(A)$ denote the next car of A. Then $C = next(A)$ and $A = next(B)$ in Fig. 3(a). Similarly B goes first, and $C = next(B)$ and $B = next(A)$ in Fig. 3(b).

Assume that $A = next(B)$. This implies that we can put the distance of B as $pos(A) - pos(B)$, where $pos(A)$ is the index of the cell occupied by A. However, sometimes two cars in front of a junction can be in the same position; namely, $pos(A) - pos(B) = 0$. In the real world, under this condition, the faster car goes first. This is quite reasonable since the faster car is more likely to reach a

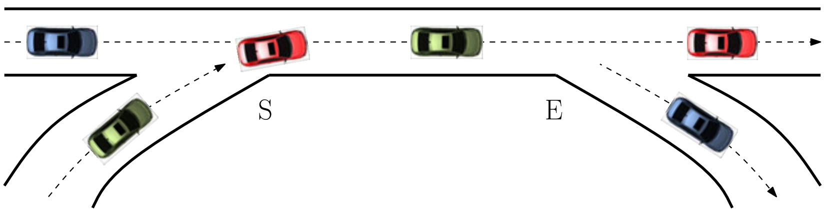


Fig. 4. Our model simulates the traffic where two lanes join at a junction and split into two lanes later. Let S and E be the beginning and the ending of the joint single-lane, respectively.

junction earlier than the other car. These cases are depicted in Fig. 3(c) and (d). In (c), A goes first and $A = next(B)$ since A is faster than B.

Here we consider one more case: Two cars with the same distance from a junction and the same velocity as depicted in Fig. 3(e). We can assume that one of the cars would decrease the velocity because otherwise the two cars would crash into each other. We introduce a simple rule to avoid a crash: We randomly select a car and reduce the velocity of the selected car with probability p_{follow} to mimic the real-world behavior. Since the two lanes have the same priority, we set $p_{follow} \leftarrow 0.5$. Now the two cars have different velocities and thus we can follow one of the two cases in Fig. 3(c) and (d). We say that two cars A, B are on the same position if $pos(A) = pos(B)$.

Fig. 4 illustrates the traffic flow that we consider: Two lanes join as a single-lane and later split again. We use *start* and *end* to denote the beginning and the ending of the joint single-lane. Let $\mathcal{N}(start, i)$ be the car nearest to the junction on lane i. We include a 'form-one-lane' rule as follows:

1. On the joint single-lane: if $pos(start) \leq pos(A) < pos(end)$, then d is the distance from A to the nearest car ahead of A.
2. Closest to a junction: if $pos(\mathcal{N}(start, 1)) = pos(A) > pos(\mathcal{N}(start, 2))$, then d is the distance from A to the nearest car C ahead of A and we set v_{next} as the velocity of C. (Fig. 3(a) case)
3. Behind another car on the other lane: if $pos(\mathcal{N}(start, 1)) = pos(A) < pos(\mathcal{N}(start, 2))$, then $d \leftarrow pos(\mathcal{N}(start, 2)) - pos(A)$ and $v_{next} \leftarrow v(\mathcal{N}(start, 2))$. (Fig. 3(b) case)
4. On the same position: when $pos(\mathcal{N}(start, 1)) = pos(A) = pos(\mathcal{N}(start, 2))$.
   (a) If $v > v(\mathcal{N}(start, 2))$, then d is the distance between the car that is in front and on the single-lane. (Fig. 3(c) case)
   (b) If $v < v(\mathcal{N}(start, 2))$, then we set $d \leftarrow 0$ in order to stop the car and follow the car on the other lane. In this case, we do not need to set v_{next} since this car stops here regardless of the velocity of the next car. (Fig. 3(d) case)
   (c) If $v = v(\mathcal{N}(start, 2))$, then with probability p_{follow} d is the distance between the next car on the single-lane, v_{next} is the velocity of that car and the velocity of the car on the other lane decreases ($v(\mathcal{N}(start, 2)) \leftarrow v(\mathcal{N}(start, 2)) - 1$). (Fig. 3(e) case)

5. Otherwise: the car is not affected by the junction. Let C be the car that is ahead of A. Then d is the distance from A to C and v_{next} is the velocity of C.

These rules determine the values of d and v_{next} of all cars. However, it is impossible to determine the values of all cars in parallel with these rules. When two cars are on the same position and have the same velocity, we reduce the velocity of one car with probability p_{follow}. Then the two cars become to have different velocities and the slower car follows the faster car. If these rules are applied to all cars in parallel, then the velocity of two cars can be reduced at the same time and this causes two cars to stop. Thus, for two cars with the same velocity, we avoid this problem by applying these rules to each car one by one.

1. Slow-to-start: if $v = 0$ and $d > 1$, then the car accelerates on this step or stays there and accelerates on the next step.
2. Deceleration (when the next car is near): if $d \leq v$ and either $v < v_{next}$ or $v \leq 2$, then the next car is either very close or going at a faster speed, and we prevent a collision by setting $v \leftarrow d - 1$ but do not slow down more than is necessary. Otherwise, if $d \leq v$, $v \geq v_{next}$, and $v > 2$ we set $v \leftarrow min(d-1, v-2)$ in order to possibly decelerate slightly more, since the car ahead is slower or the same speed and the velocity of the current car is substantial.
3. Deceleration (when the next car is far): if $v < d \leq 2v$, then if $v \geq v_{next} + 4$, decelerate by 2 ($v \leftarrow v - 2$). Otherwise, if $v_{next} + 2 \leq v \leq v_{next} + 3$ then decelerate by one ($v \leftarrow v - 1$).
4. Acceleration, Randomization, Car motion: these rules are same as in the NaSch model.

These velocity rules calculate the velocities of all cars. Now we design a 'form-one-lane' model by using these two rule sets. The main concern of our model is how to determine the next car when two cars are near a junction, especially when they are on the same position.

Note that when two cars are on the same position, by our rules, the faster car has d as the distance to the car that is in front of the two cars and v_{next} as the velocity of that car while the other car has d as 0. By the deceleration rule, if d is 0, then the velocity of the car becomes -1. However, since the domain of velocity is from 0 to v_{max}, we set velocity as 0. This follows that when two cars are on the same position near a junction, one car stops there and thus two cars cannot advance at the same time.

4 Experiments and Analysis

4.1 Single-Lane Model and the Proposed Model

We simulate the proposed model for traffic with a junction and compare the simulation results with an example of the 'slow-to-stop' model by Clarridge and Salomaa [2]. Fig. 5 is an example of two simulations.

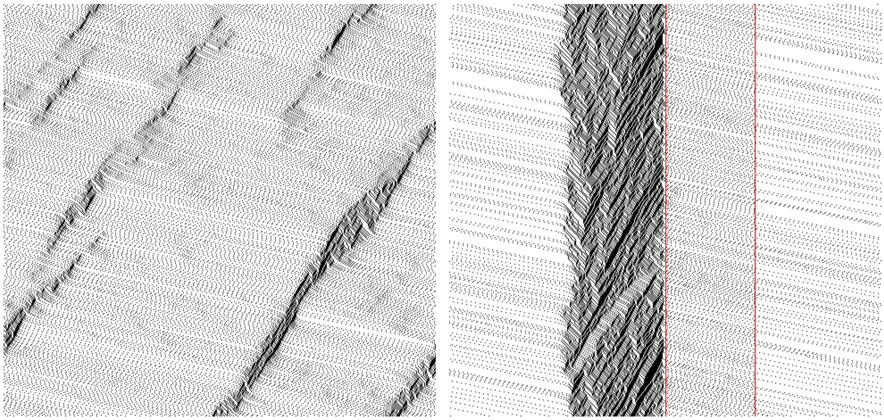
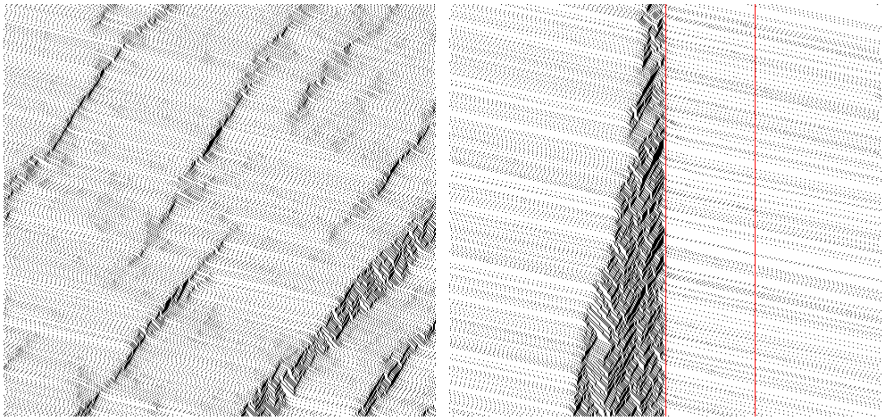

(a) (b)

Fig. 5. The two pictures depict the distribution of cars over 500 consecutive time steps. (a) is an example of the 'slow-to-stop' model by Clarridge and Salomaa [2] and (b) is an example of the proposed simulation model. Two vertical lines denote the beginning and the ending of a single-lane. Cars are moving from left to right.

In the simulation in Fig. 5, we use the following parameter values: $v_{max} = 5, p_{fault} = 0.1, p_{slow} = 0.5$ and $\rho = 0.15$. v_{max} is the maximal velocity of cars. The cars on cells can move to the right at most v_{max} cells for each time step. p_{fault} and p_{slow} are the probabilities for the disorder rule and the slow-to-start rule. We calculate the traffic density ρ by the ratio of the number of cars to the number of cells. In our model, there are two lanes at first and become a single-lane at a certain point. We have simulated with the same density (0.15) for two lanes.

Notice that the start-stop-waves (traffic jams) in Fig. 5(a) often appear in the real-world traffic. These jams move backwards slowly as the time passes and occur randomly. Note that the locations of traffic jams are different and unpredictable. On the other hand, the location of traffic jams in our model are consistent.

As shown in Fig. 5(b), most start-stop-waves occur in front of the traffic junction. This is quite similar to the real-world traffic flow, where most of traffic jams occur in front of traffic junctions when the number of lanes decreases. The lengths of traffic jams are almost the same when we simulate it with the fixed number of cars. This is because our simulation is carried out using a circular road. In our simulation, there is a fixed number of cars on a fixed number of cells. The influx, which is the number of cars coming into the start of the lane in each time step, is the same as the outflux of the lane after a junction. This means that the flux of a junction goes into the start of the lane along the circular road repeatedly. Thus, the amounts of the influx and the outflux are the same and this is the reason why the lengths of traffic jams remain still as the time passes.



(a) (b)

Fig. 6. The two pictures depict the distribution of cars over 500 consecutive time steps. We use randomly generated roads to add the cars to a stretch of road. (a) is an example of single-lane model and (b) is an example of form-one-lane model.

However, roads are typically not circular in reality. Furthermore, if the influx is larger than the maximal flux that the junction can process, then the number of cars before the junction would increase. On circular roads, it is impossible to simulate this. Clarridge and Salomaa [2] addressed this problem with an alternative method using Bernoulli process arrivals. This method adds new cars to a stretch of road instead of using CAs with circular boundary conditions. We address this issue by generating a road with the desired flux with a fixed density of cars. For instance, when we make a road with density 0.15, the flux is maximal (which is about 0.52). If we make a road with density 0.07, then the influx of each lane is 0.34. We randomly generate roads by iterating simulations 1000 times to stabilize with the fixed density of cars. In Fig. 6, we confirm that the traffic jam does not show the periodic trends anymore and the lengths of traffic jams increase linearly.

4.2 The Traffic Flux with a Junction

Because the construction of new roads or traffic facilities costs a lot of money, it is better to predict a possible traffic flow before the construction. Especially, if we attach a new road to an existing road without a traffic jam, it may cause some traffic jams because of the new junction. The proposed model can estimate a traffic flow when building a new road and creating a junction.

The traffic flux is the number of cars passing through the lane in a time step. This is an important factor for adding new roads or building traffic facilities since high flux implies the efficient traffic flow. Thus it is better to maximize the traffic flux. In the simulation conducted by Clarridge and Salomaa [2], the flux is maximal when the density of the traffic is around 0.15.

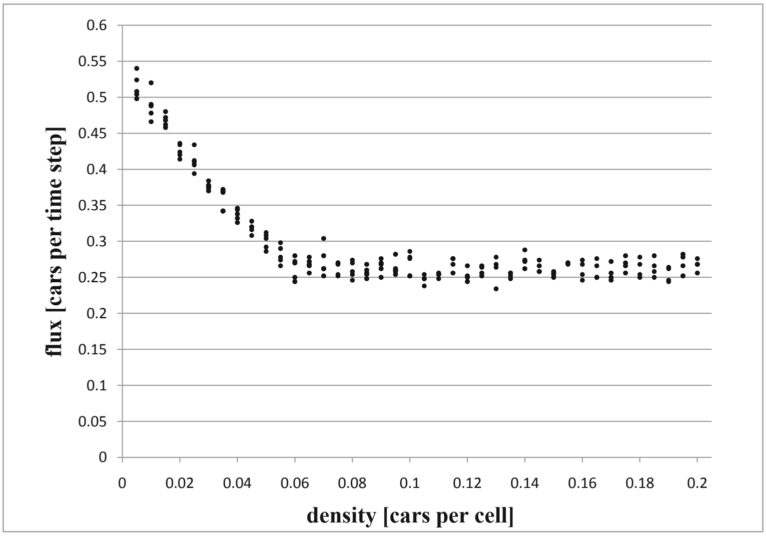


Fig. 7. The fundamental diagram of our model. We simulate with two lanes, where a lane L_1 has a constant density 0.15 and the other lane L_2 has a varying density. In simulation, we increase the density of L_2 as 0.005 in each step.

We examine the case when a lane L_1 is at maximal flux and the density of the other lane L_2 varies. We fix the density of L_1 as 0.15 and vary the density of L_2. The fundamental diagram of this simulation depicted in Fig. 7 shows a linear decrease until the density of L_2 becomes 0.06. After 0.06 the flux of L_1 is stabilized around 0.27. This is because the length of the queue of static cars in front of the traffic junction does not affect the minimal flux of L_1 anymore. Based on this observation, we can estimate the worst-case time complexity to pass the junction.

4.3 The Length of Traffic Jam

Since the traffic jams in our simulation often occur in front of a traffic junction, it is possible to compute the length of traffic jams and make use of them more easily than general start-stop waves. In Fig. 5(b), for instance, we can estimate the rough length of the jams. With the fixed densities of the traffic, it always has similar length of traffic jams in our simulation. If it occurs in the real-world traffic flow in a similar way, then we can use these simulation results for predicting the length of actual traffic jams in front of a junction. However, since there is no general method of measurement for the traffic jams, we design a new rule for the simulation. When we decide whether or not a traffic is jammed, we focus on the partial density of the traffic. If a partial traffic is denser than the other part, then we consider the part as a traffic jam. For the part from the starting point of the traffic to a junction, we measure the average density of the traffic. If the density is lower than the standard, then the starting point moves to the right.

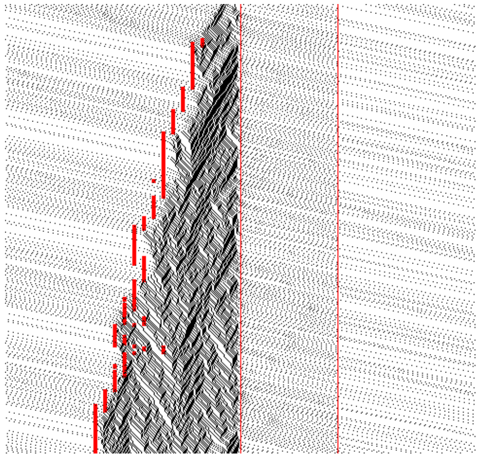


Fig. 8. Thick dots denote the point where the traffic jam begins. The standard density is 0.4.

It stops at the point where we have the average density for the part higher than the standard density.

We check the starting point of traffic jam as thick dots in Fig. 8. We use the standard density 0.4 since the maximal density is approximately 0.4 when a car is in a complete jam in our model. We observe that the length of traffic jam increases linearly.

4.4 Applications of the Proposed Model

When we use the simulation results for the real-world traffic flow, the quantitative comparison is needed. Since the unit in our simulation is abstract, we need some scaling between our model and the real-world traffic. One approach of scaling is to use the maximal velocity. Assume that the maximal velocity of the real-world traffic is 100km/h. The maximal velocity in our simulation is 5 cells per time step and, therefore, we can scale two values based on the following proportional equation:

$$\frac{5 \text{ cells}}{1 \text{ time step}} = \frac{100\text{km}}{1 \text{ hour}} = \frac{100000\text{m}}{3600 \text{ seconds}}$$

If we regard the size of a cell as 7m, then the time scale is 1.26 time steps to one second. Based on this scaling, we can calculate either the time to escape from the jam or the estimated arrival time. If a car in our simulation takes n time steps to escape from the jam and enters the single-lane, we can simply convert n time steps to $\frac{n}{1.26}$ seconds in real time.

In a similar way, we can predict the length of traffic jams in the real-world traffic. The simplest way is to use the size of one cell. Nagel and Schreckenberg [9] claimed that in a complete jam each car occupies about 7.5m of place, which becomes the length of one cell. The second method is the comparative method. We can simulate a traffic when there is a junction with various parameters using our model. When the traffic is in the standard condition and has a traffic jam whose length is 10km in reality, we can simulate the condition with our model and obtain the traffic jam whose length is 50 cells. Then, we can simulate with other parameters such as various amounts of flux and obtain the expected length of traffic jam. If the length of traffic jam is, say, 100 cells in simulation, it becomes 20km.

5 Conclusions

When two lanes join, there is always a junction. We have proposed a CA-based traffic simulation model with a 'form-one-lane' rule for simulating traffic flow with a junction. We have considered five possible cases near a junction based on the BJH model [1] and the 'slow-to-stop' model [2], and demonstrated that the proposed model can predict the traffic flow with a junction accurately.

With some diagrams and examples, we have analyzed the experimental results and examined the length of traffic jams when the traffic density varies and the flux of the traffic with a junction. Remark that the flux of the lane becomes stable when the density of a lane is greater than a certain threshold. Based on this observation, we can predict the worst-case arrival time when the density of the other lane varies. We have suggested two approaches of using the proposed model to the real-world traffic.

In future, we need to compare the empirical traffic data with our simulation result. From the real-world data and the simulation results, we can adjust parameters and adopt new scale ratio to establish more precise prediction.

Acknowledgements

We thank Adam Clarridge for providing his source code of the car traffic model with a slow-to-stop rule. We wish to thank the referees for providing us with constructive comments and suggestions.

This research was supported by the Basic Science Research Program through NRF funded by MEST (2010-0009168).

References

1. Benjamin, S.C., Johnson, N.F., Hui, P.M.: Cellular automata models of traffic flow along a highway containing a junction. Journal of Physics A: Mathematical and General 29(12), 3119–3127 (1996)
2. Clarridge, A., Salomaa, K.: Analysis of a cellular automaton model for car traffic with a slow-to-stop rule. Theoretical Computer Science 411(38-39), 3507–3515 (2010)
3. Emmerich, H., Rank, E.: An improved cellular automaton model for traffic flow simulation. Physica A: Statistical and Theoretical Physics 234(3-4), 676–686 (1997)

4. Knospe, W., Santen, L., Schadschneider, A., Schreckenberg, M.: Towards a realistic microscopic description of highway traffic. Journal of Physics A: Mathematical and General 33, 477–485 (2000)
5. Makowiec, D., Miklaszewski, W.: Nagel-Schreckenberg model of traffic - study of diversity of car rules. In: International Conference on Computational Science, vol. 3, pp. 256–263 (2006)
6. Nagel, K., Schreckenberg, M., Latour, A., Rickert, M.: Two lane traffic simulations using cellular automata. Physica A: Statistical and Theoretical Physics 231(4), 534–550 (1996)
7. Nagel, K., Wolf, D.E., Wagner, P., Simon, P.: Two-lane traffic rules for cellular automata: A systematic approach. Physical Review 58(2), 1425–1437 (1998)
8. Nagel, K.: Particle hopping models and traffic flow theory. Physical Review E 53(5), 4655–4672 (1996)
9. Nagel, K., Schreckenberg, M.: A cellular automaton model for freeway traffic. Journal de Physique I 2(12), 2221–2229 (1992)
10. Sasvári, M., Kertész, J.: Cellular automata models of single-lane traffic. Physical Review E 56(4), 4104–4110 (1997)
11. Schadschneider, A., Schreckenberg, M.: Cellular automaton models and traffic flow. Journal of Physics A: Mathematical and General 26, 679–683 (1993)
12. Wagner, P.: Traffic simulations using cellular automata: Comparison with reality. In: Traffic and Granular Flow. World Scientific, Singapore (1996)
13. Wagner, P., Nagel, K., Wolf, D.E.: Realistic multi-lane traffic rules for cellular automata. Physica A: Statistical and Theoretical Physics 234(3-4), 687–698 (1997)
14. Website, T.S.M.S.: Road rule changes (June, 2009), `http://www.seniorsmovingsafely.org.au/road_rules.html`
15. Wolfram, S.: Theory and Applications of Cellular Automata. World Scientific, Singapore (1986)
16. Wolfram, S.: A New Kind of Science. Wolfram Media, Champaign (2002)
17. Xiao, S., Kong, L., Liu, M.: A cellular automaton model for a bridge traffic bottleneck. Acta Mechanica Sinica 21, 305–309 (2005)