COMPUTER ANIMATION AND VIRTUAL WORLDS
*Comp. Anim. Virtual Worlds* 2014; **25**:83–97
Published online 19 July 2013 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/cav.1540

RESEARCH ARTICLE

# AA-FVDM: An accident-avoidance full velocity difference model for animating realistic street-level traffic in rural scenes

Xuequan Lu[1], Wenzhi Chen[1], Mingliang Xu[2], Zonghui Wang[1]*, Zhigang Deng[3] and Yangdong Ye[2]

[1] Zhejiang University, Hangzhou, 310027, China
[2] Zhengzhou University, Zhengzhou, China
[3] University of Houston, Houston, TX, USA

## ABSTRACT

Most of existing traffic simulation efforts focus on urban regions with a coarse two-dimensional representation; relatively few studies have been conducted to simulate realistic three-dimensional traffic flows on a large, complex road web in rural scenes. In this paper, we present a novel agent-based approach called accident-avoidance full velocity difference model (abbreviated as AA-FVDM) to simulate realistic street-level rural traffics, on top of the existing FVDM. The main distinction between FVDM and AA-FVDM is that FVDM cannot handle a critical real-world traffic problem while AA-FVDM settles this problem and retains the essence of FVDM. We also design a novel scheme to animate the lane-changing maneuvering process (in particular, the execution course). Through numerous simulations, we demonstrate that besides addressing a previously unaddressed real-world traffic problem, our AA-FVDM method efficiently (in real time) simulates large-scale traffic flows (tens of thousands of vehicles) with realistic, smooth effects. Furthermore, we validate our method using real-world traffic data, and the validation results show that our method measurably outperforms state-of-the-art traffic simulation methods. Copyright © 2013 John Wiley & Sons, Ltd.

**KEYWORDS**

traffic animation; microscopic model; vehicle flow; rural traffic

Supporting information may be found in the online version of this article.

***Correspondence**

Zonghui Wang, College of Computer Science and Technology, Zhejiang University, Hangzhou, China.
E-mail: zjuzhwang@zju.edu.cn

## 1. INTRODUCTION

Traffic simulation techniques can play a key part in urban development and planning, road network design, road systems, and roadside hardware. At the same time, it is also of great importance in improving and completing traffic laws, guidelines, and policies to explore the motion of traffic flows. In addition to these, it is of growing need to apply realistic street-level traffic to applications, for example, virtual traffic driving for driving instructions, virtual tourism, special effects for movies, racing games, and so on. However, even though there are plenty of traffic simulation models, most existing traffic simulators (e.g., [1]) utilize a two-dimensional abstract representation and center on urban-related traffic; therefore, a system for animating realistic traffic in rural places is practically needed for its critical role in society functioning.

Although continuum-based methods [2–8] are fast and efficient, they have limitations such as lack of diversity in individualistic behavior and difficulty in handling scenes with complicated road networks due to their inherent characteristics and inflexibility. Therefore, agent-based methods (or microscopic methods) [9–17] have been the most popular traffic simulation approaches in recent years. Through a number of delicately designed rules, agent-based methods are capable of simulating the following: (i) detailed behavior including anisotropic drivers; and (ii) individualistic behavior with complex dynamics.

Among various agent-based approaches, car-following models have achieved noticeable successes in recent years, such as the optimal velocity model (OVM) [12], the generalized force model (GFM) [13], and the full velocity difference model (FVDM) [9]. In particular, FVDM [9], which is built on top of OVM and GFM, has been proven to be an

Copyright © 2013 John Wiley & Sons, Ltd.

effective traffic simulation model and can simulate realistic traffic in most cases. However, it fails to handle the following situation (called close-car-braking circumstance): if the distance between the leading car and the following car is small, their speeds are quite close, and the leading car makes a sharp brake for an accident ahead or a red traffic light at a crossroad, in this case, accidents often can be avoided in the real world. However, in the FVDM simulation, the two cars will collide after a few seconds (refer to Figures 7 and 11 in Section 5). It should be noted that the close-car-braking circumstance plays a pivotal role in the operation of a realistic traffic system, as this phenomenon often occurs in real-world traffic.

Motivated by the above problem, we propose a novel approach called accident-avoidance FVDM (AA-FVDM) that retains the essence of FVDM while elegantly handling the aforementioned close-car-braking circumstance. From a technical perspective, our approach essentially introduces two force terms ("psychological force" and "body force") if the distance between the leading car and the following car is within a pre-specified threshold. Our approach generates suitable deceleration to the following vehicle, which can timely avoid unnecessary accidents in the close-car-braking circumstance. On the other hand, the lane-changing maneuver was mostly modeled as either an instantaneous event [1,18–21] or an action with uniform duration [22]. Moreover, existing lane-changing models typically focus on making decisions and overlook the execution of the maneuvering process [23]. In this work, we introduce a novel scheme to animate the detailed course of lane-changing behavior, particularly, the execution course covering dynamics and constraints of a lane-changing vehicle.

The *contributions* of this work are as follows:

(1) A novel approach to address an important yet largely underexplored traffic simulation problem, the "close-car-braking circumstance," while preserving the advantages of FVDM
(2) A new scheme to animate lane-changing maneuvering process in detail, especially the execution course

With our method, we build a traffic simulation system that can simulate large-scale lifelike rural traffic flows Figure 1 at interactive rates in complex road networks containing flyovers, suspension bridges, curving tunnels, and many other road structures. To evaluate our approach, firstly, we choose several scenarios including traffic jams, close-car-braking circumstance, lane-changing behavior, traffic at on/off ramps, and traffic in tunnels. Secondly, we do some performance tests and parameter tests. To validate our approach, we further compare real-world traffic data with the simulation results of a variety of traffic models (including our approach). The comparison results demonstrate that our method can measurably outperform many state-of-the-art models.

The remainder of this paper is organized as follows. In Section 2, we briefly review the related efforts that are most relevant to this work. In Section 3, we present fundamental definitions and formulations used in our methodology and give a retrospect to previous car-following models. In Section 4, we describe our proposed method in detail. In Section 5, we present simulation results by our approach and comparison results. Finally, remarks and future work are presented in Section 6.

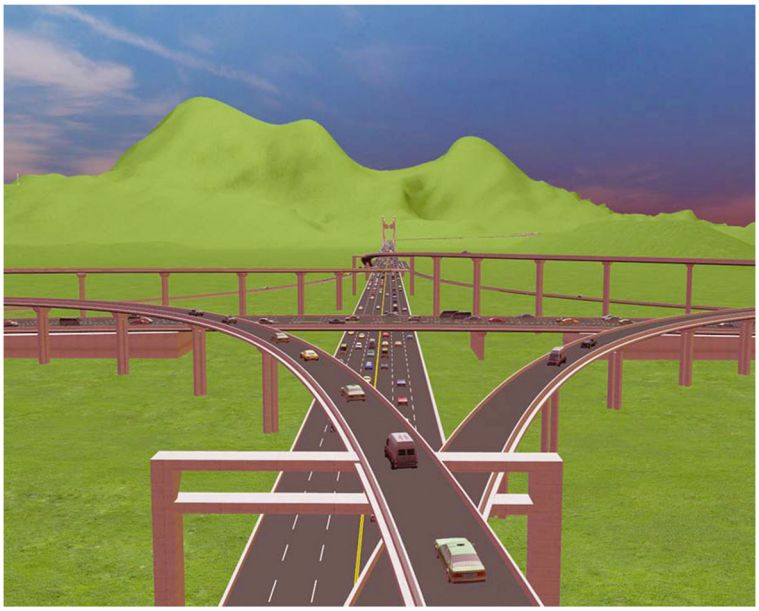


**Figure 1.** A snapshot of traffic flow in rural scenes simulated by our approach.


DOI: 10.1002/cav

## 2. RELATED WORK

### 2.1. Traffic Simulation Models

Visualization and animation of vehicles and traffic flows have attracted increasing interests [7,24–27] in the computer graphics community during the past decade. Technically speaking, there are three roughly classified categories for traffic simulation: microscopic, macroscopic, and mesoscopic. Among the three categories, the least common method is mesoscopic, which uses Boltzmann-type mesoscale equations or other junction qualities in macroscopic and microscopic methods. On the basis of the seminal work by Prigogine *et al.* [28], researchers proposed various variations or extensions to further improve the work [29,30].

To date, agent-based techniques are the most popular methods for traffic simulation, where each vehicle is regarded as an agent and a set of advanced rules is employed to generate natural vehicular behaviors. Gerlough [10] was among the first to discuss car-following rules. Subsequently, Newell [11] further merged more characteristics into agent-based models. Recently, Jiang *et al.* [9] developed an FVDM to address several shortcomings in OVM [12] and GFM [13]. However, their FVDM model (detailed in follow-up sections) could result in unnecessary collisions in the close-car-braking circumstance (described in Section 1). In addition, cellular automata were also applied to the field of traffic dynamics [16]. For more detailed descriptions of various agent-based traffic simulation models, please refer to recent surveys [14,15].

The macroscopic model is also called continuum based. Lighthill *et al.* [2] and Richards [3] did the earliest work independently in this direction; thus, this model is often called the LWR model. Because the LWR equation is a scalar, nonlinear partial differential equation taking a density variable, vehicular velocity depends solely on traffic density. In order to acquire a more complete model that is not only based on traffic density, on top of the LWR model, Payne *et al.* [4] created a second-order system of equations called the PW model, where the second order indicates that there are two distinct variables. However, the PW model might predict negative velocities under some circumstances. Later, researchers made various extensions [5,6] to the PW model to remove this limitation.

### 2.2. Lane-changing Models

Most of existing lane-changing models typically focus on agent-based traffic models. Nagel *et al.* [31] summarized different approaches for lane changing and proposed a general scheme, with which realistic lane-changing rules could be developed. Huang [32] used a cellular automaton to model lane-changing behavior on multilane highways. Hidas [22] developed a novel lane-change model to simulate vehicle interactions using intelligent agent concepts. Kesting *et al.* [21] presented a general lane-changing model for car-following traffic models by minimizing overall braking induced by lane changes. Most of these existing lane-changing models center around drivers' decisions and generally overlook the execution of the lane-changing procedure [23]. In contrast, in this work, we adopt a simple decision-making model and emphatically represent the details of the lane-changing course.

## 3. PRELIMINARIES

### 3.1. Basic Definitions

**Definition 1.** *The road network consists of a series of roads,* $RN = \bigcup \{R_i | i = 1, 2, \ldots\}$.

**Definition 2.** *There are a few lanes on a single road, and we describe lanes with several properties, for instance, the maximum speed* $v_{\max}$*, the adjacent lanes* $ls$*, and the roads the lane belongs to* $R$*. The properties of lane* $j$ *can be described as* $P_j = \{v_{\max}, ls, R\}$.

**Definition 3.** *Every lane* $j$ *has a number of cars,* $Cars_j = \bigcup \{c_i | i = 1, 2, 3, \ldots\}$*. Because our model is agent based, we need some further information to depict its state, including position (center of a car), velocity, length, detecting radius (Definition 4), and vehicle types. For simplicity, we denote it as* $c_i = \{x_i, v_i, l_i, r_i, t_i\}$.

**Definition 4.** *There is a virtual ellipse encircling each car, and we regard its semimajor axis as the vehicle's detecting radius* $r_i$ *and use an upper limit of 3 for* $\gamma$ *in this work.*

$$r_i = \gamma l_i, \gamma \in (1, 3] \tag{1}$$

**Definition 5.** *Before vehicles' behavior is actually animated, it is necessary to set the initial and boundary conditions including position, velocity, type, length, and so forth. The boundary conditions include the length of lanes and how to handle cars that come and go across the boundary of the simulation area.*

### 3.2. Previous Car-following Models

#### 3.2.1. The Optimal Velocity Model.

In 1995, Bando *et al.* [12] proposed a microscopic traffic model termed OVM, the main idea of which is that each vehicle adapts to a distance-dependent optimal velocity. The following OVM equation describes the movement of vehicles.

$$\frac{dv_i}{dt} = \kappa[tV(s_i) - v_i] \tag{2}$$

Here,

$$s_i = |x_{i-1} - x_i| - (l_{i-1} + l_i)\ /\ 2 \tag{3}$$

The optimal velocity function is determined by the headway distance as follows.

$$V(s_i) = V_1 + V_2 \tanh(C_1 s_i - C_2) \tag{4}$$

where $\kappa$ is a sensitivity constant, $s_i$ denotes the netto distance (net spacing) between the car $i$ and the front car $i-1$, and $l_{i-1}$ is the length of vehicle $i-1$.

Helbing *et al.* [13] used empirical data to calibrate the parameters of the optimal velocity function. The resulting optimal parameter values are $\kappa = 0.85\ \mathrm{s}^{-1}$, $V_1 = 6.75$ m/s, $V_2 = 7.91$ m/s, $C_1 = 0.13\ \mathrm{m}^{-1}$, and $C_2 = 1.57$. The OVM can describe many properties of real traffic flows and is easily interpretable, despite its simplicity and few parameters. However, the comparison with the field data indicates that an unrealistically high acceleration and a similar problem in deceleration turn up in OVM [13].

#### 3.2.2. The Generalized Force Model.

To eliminate the shortcomings of OVM, Helbing *et al.* [13] came up with a GFM inspired by the social force concept from pedestrian dynamics [33]. Its original equation is

$$\frac{dv_i}{dt} = \frac{v_i^0}{\tau_i} + f_{i,i-1} \tag{5}$$

where $v_i^0$ is the desired speed the vehicle $i$ prefers, $\tau_i$ is the acceleration time, and $f_{i,i-1}$ is the repulsive force that enables car $i$ to keep a safe distance from the front car $i-1$.

According to the social force concept, the temporal change of velocity (i.e., acceleration) is given by a sum of forces ($v_i^0/\tau_i$ can also be interpreted as a social force). So please note that $f_{i,i-1}$ in Equation (5) is a social force, which is different from the traditional force concept, and the unit of $f_{i,i-1}$ is the acceleration unit.

In GFM, the force $f_{i,i-1}$ is as follows:

$$f_{i,i-1} = \frac{V(s_i) - v_i^0}{\tau_i} - \frac{\Delta v_i \Theta(\Delta v_i)}{\tau_i'} e^{-[s_i - s(v_i)]/R_i'} \tag{6}$$

Here, $V(s_i)$ is the optimal velocity function described in OVM, $\Delta v_i = v_i - v_{i-1}$ is the velocity difference between the following car $i$ and the leading car $i-1$. $\Theta$ is the Heaviside function, and it should contribute only if the following car is faster than the leading vehicle. Ignoring the negative velocity difference leads to several disadvantages, such as poor delay time.

The GFM could be written in the following form, which is more convenient for discussion.

$$\frac{dv_i}{dt} = \kappa\left[v_i^0 - v_i\right] + \kappa\left[V(s_i) - v_i^0\right] - \lambda\Theta(\Delta v_i)\Delta v_i \tag{7}$$

#### 3.2.3. The Full Velocity Difference Model.

In 2001, an FVDM was presented by Jiang *et al.* [9] who considered the negative velocity difference based on GFM. So, the derivation process of the FVDM is like this:

$$\begin{aligned}\frac{dv_i}{dt} &= \kappa\left[v_i^0 - v_i\right] + \kappa\left[V(s_i) - v_i^0\right] \\ &\quad - \lambda\Theta(\Delta v_i)\Delta v_i - \lambda\Theta(-\Delta v_i)\Delta v_i \\ &= \kappa[V(s_i) - v_i] - \lambda\Delta v_i\end{aligned} \tag{8}$$

Here, $\lambda$ is chosen as a step function as follows.

$$\lambda = \begin{cases} a & s_i \le s_c \\ b & s_i > s_c \end{cases} \tag{9}$$

where $\kappa = 0.41\ \mathrm{s}^{-1}$, $a = 0.5\ \mathrm{s}^{-1}$, $b = 0$, and $s_c = 100$ m. FVDM behaves better than OVM and GFM in spite of its simplicity, as it takes more factors into account in car-following rules. However, there are still a few weaknesses in describing some pressing traffic conditions of FVDM.

## 4. OUR METHOD

### 4.1. Overview

As illustrated in Figure 2, in our approach, given a time step, the following steps are executed to simulate vehicular flows.

**Step 1:** Before the simulation, we must specify the initial and boundary conditions, including initialization of position, velocity, and so on.

**Step 2:** We use our introduced AA-FVDM method to compute acceleration for each vehicle except the leading one and then compute the new velocity for the next time step.

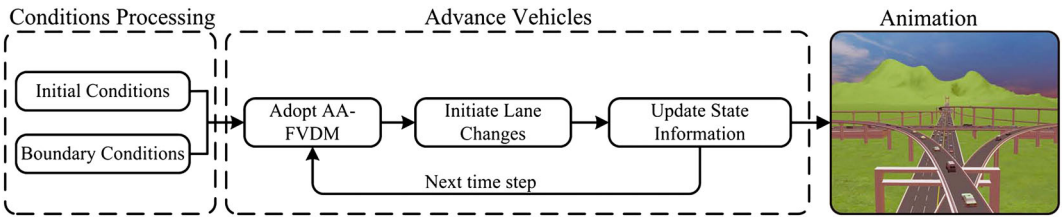



**Figure 2.** The overview for our system.


DOI: 10.1002/cav

**Step 3:** If there are vehicles that meet the lane-changing rules, we start those lane changes.

**Step 4:** We advance cars to the next time step using the computed information and update the network state; we jump to step 2, and another cycle begins.

## 4.2. Accident-avoidance full velocity difference model

Our AA-FVDM is inspired by the GFM [13], the FVDM [9], and crowd dynamics [33,34]; hence, we start from a force perspective. To cope with the close-car-braking circumstance, we assume there is an elliptic range embracing each vehicle. Each car and its corresponding ellipse have the same center point on the horizontal plane. The semimajor axis $r_i$ has a relation with the length of the corresponding car $l_i$, which is $r_i = \gamma l_i, \gamma \in (1, 3]$. If the distance $d_{i,i-1} = |x_{i-1} - x_i|$ between the two cars $i$ and $i-1$ nearest to each other is smaller than the combined radii $r_{i,i-1} = r_i + r_{i-1}$ of their semimajor axes, there will exist two kinds of interaction forces called psychological force $f_2(i, i-1)$ and body force $f_3(i, i-1)$. $f_2(i, i-1)$ describes that a driver will have a psychological tendency to keep away from another car if the distance between them is smaller than the safe interval. $f_3(i, i-1)$ counteracts the body compression of two neighboring ellipses in a lane. As a consequence, we add these two force terms into Equation (8) as follows.

$$\frac{dv_i}{dt} = \kappa[V(s_i) - v_i] - \lambda \Delta v_i + f_2(i, i-1) + f_3(i, i-1) \tag{10}$$

Both $f_2$ and $f_3$ should increase with decreasing distance $d_{i,i-1}$ but disappear if the distance $d_{i,i-1} \geq r_{i,i-1}$. This strategy is consistent with real-world traffic: the following vehicle will decelerate strongly to avoid an accident if the distance between it and the leader is small; the smaller the distance is, the larger the deceleration becomes. A linear function is usually used to describe compression and stretching forces such as spring forces and $f_3$. Additionally, an exponential expression for $f_2$ enables a timely response for vehicles by adapting the force exponentially. The overall acceleration is given by a sum of social forces (not only a single force). If one following vehicle wants to avoid oscillation, a probable way is to change the velocity of the headmost vehicle in the same lane along with time.

Let $z_{i,i-1} = r_{i,i-1} - d_{i,i-1}$, so

$$f_2(i, i-1) = -C\Theta(z_{i,i-1})e^{(z_{i,i-1})/D} \tag{11}$$

$$f_3(i, i-1) = -k\Theta(z_{i,i-1})H(z_{i,i-1}) \tag{12}$$

and $\Theta(x)$ is the Heaviside function

$$\Theta(x) = \begin{cases} 1 & x > 0 \\ 0 & \text{otherwise} \end{cases} \tag{13}$$

together with

$$H(x) = x \tag{14}$$

$C$, $D$, and $k$ are positive constants, and in a traffic simulation, we set $C \in (0, 1]$, $k \in (0, 1]$, $D = \max(r_{i,i-1}) - \min(d_{i,i-1})$.

From our observation and the description of Helbing *et al.* [13], the deceleration capabilities of vehicles are greater than their acceleration capabilities; thus, our approach is consistent with real-world scenario.

From Equation (10), we can tell that our AA-FVDM degenerates to FVDM if there are no psychological force and body force and reduces to GFM regardless of the negative effect of velocity difference and if no added force terms are in effect. It also degrades to OVM if $\lambda = 0$, $f_2(i, i-1) = 0$, and $f_3(i, i-1) = 0$.

## 4.3. Handling Lane Changes

### 4.3.1. Decision Making on Lane Changes.

Inspired by empirical observations, we propose an efficient method in light of visual information, which is, to a certain extent, similar to existing rules or models on decisions of whether performing a lane change.

Generally speaking, drivers try to make lane changes for various reasons, for example, overtaking slow traffic, going to off-ramps, avoiding other cars, slowing down, making turns, road narrowing, and so on. In this work, we simply select overtaking and taking off-ramps as the main incentives. Besides, when changing to other lanes, the safety criterion (i.e., no collisions) should be guaranteed. We first introduce some variables before formulating the incentive criterion and safety criterion.

As illustrated in Figure 3, $v_r^h$ is the velocity of the leading car on the right-hand lane, $v_l^h$ is the velocity of the ahead vehicle on the left-hand lane, $v$ and $pos$ are the speed and position of the car that considers making a lane change, $v^h$ is the speed of the car ahead of the current car, and $\eta$ and $\mu$ are critical coefficients (typically larger than 1).

Preference factor $\delta$ represents the preference of taking off-ramps when meeting an exit, and $v_{\max}$ is the maximum speed on a certain lane. We set $T = 1$ s.

*Incentive criterion:* $v_r^h \geq \eta v^h$, $v_l^h \geq \eta v^h$, $v_r^h \geq \mu v$, and $v_l^h \geq \mu v$ (overtaking incentive); $\delta \geq 0.8$ (exit ramp stimulation).

*Safety criterion:* The boundaries $[pos - Tv_{\max}, pos + vT]$ ensure a safe range on the target lane (i.e., the right lane or the left lane) with the current position $pos$ of the vehicle taken into account.

Note that if there is a long enough distance without any vehicles on the target lanes, it may be reasonable for the

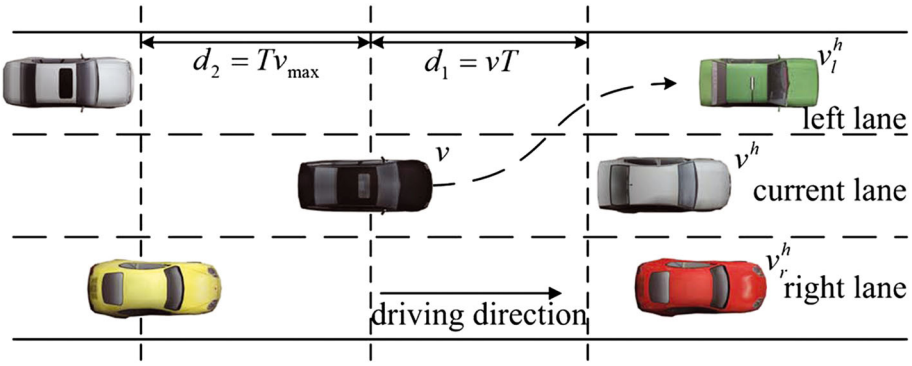



**Figure 3.** An illustration for making decisions of lane changes. The black car meets the safety criterion on the left lane, and it will change to the left lane if it further satisfies the incentive criterion and is willing to perform a lane change.

current car to perform a lane change. Under this situation, we usually initiate a lane change.

We assign each vehicle a probability $P$ when these criterion are met. In other words, a car has a probability of $1 - P$ to stay in the current lane. In our implementation, $P \geq q$ ($q \in [0.7, 1.0]$) is stochastically prescribed for some cars that decide to perform lane changes. Only both criteria are satisfied, and there is a high probability $P$ that a driver can make a lane change. In the simulation, if lane changes for a single vehicle are frequent, we can decrease the chance of lane changes for this lane-changed vehicle by lowering the probability $P$.

#### 4.3.2. Execution of the Lane-changing Course.

We aim to generate lifelike traffic animation, and a lane change needs several seconds according to real-life experience. An effective way is to approximate the lane-changing trajectory by a series of short lines. In this section, we present the process of lane changes at great length. The preceding scheme is an illustration describing the dynamics of a car that performs a lane change (Figure 4).

First of all, it is necessary to make some derivations using a discretization method within a time step $\Delta t$.

$$\frac{\Delta d}{\Delta t} = v, \frac{\Delta v}{\Delta t} = a, \frac{\Delta \beta}{\Delta t} = \omega \tag{15}$$

where $\Delta t$ is the time step length, $\Delta d$ denotes the length of a small path segment during the lane-changing procedure, $\Delta v$ indicates the velocity difference, $\Delta \beta$ is the steering angle difference, and $d$, $v$, and $\beta$ are the traversed path length, the velocity, and steering angle of a car, respectively.

On the basis of the real-world traffic or lane-changing reasons, the velocity $v$, acceleration $a$, and the steering speed $\omega$ are all limited: $v \in (0, v_{\max}]$, $a \in [0, a_{\max}]$, and $\omega \in [-\omega_{\max}, \omega_{\max}]$. The positive velocity assures that the current vehicle should only move forward. The corresponding acceleration is non-negative such that the car can make a lane change smoothly and steadily.

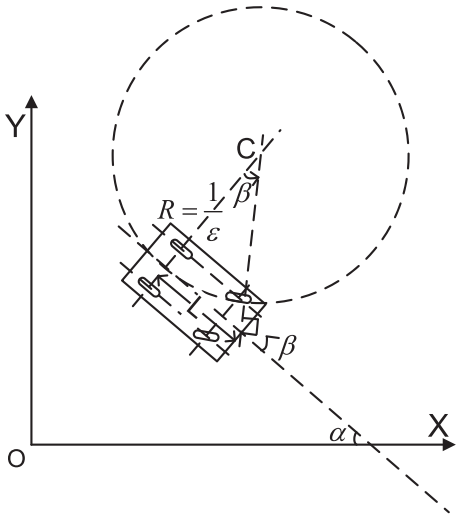



**Figure 4.** The dynamics of a lane-changing car. $\alpha$ is the direction relative to the $X$-axis, $\beta$ is the steering angle, and $\varepsilon$ and $R$ are the curvature and the radius of the traveled path, respectively.

Regarding the length of the path segment $\Delta d$, we use it to compute the homologous curvature, several trigonometric functions, the new steering angle $\alpha(d + \Delta d)$, and the new coordinate values $x(d + \Delta d)$, $y(d + \Delta d)$.

$$\begin{aligned} \frac{\Delta \alpha}{\Delta d} &= \varepsilon = \frac{\tan \beta}{L} \\ \frac{\Delta x}{\Delta d} &= \cos \alpha, \frac{\Delta y}{\Delta d} = \sin \alpha \end{aligned} \tag{16}$$

Here, the computation of the curvature $\varepsilon$ complies with its inherent definition; $\beta$ is the steering angle with $|\beta| \leq \beta_{\max}$, and $L$ is the distance between the rear axle and the front line of a vehicle (Figure 4); $\Delta x$ and $\Delta y$ are the length variations with respect to the segment length along the $X$-axis and $Y$-axis, respectively; $\alpha$ is the lane-changing direction relative to the $X$-axis; $x$, $y$ are coordinate values along the $X$, $Y$-axes.

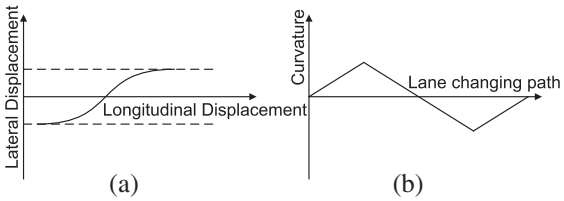



**Figure 5.** A lane-changing curve and its corresponding curvature whose derivative is constant. (a) The relation between lateral displacement and longitudinal displacement over a lane change. (b) The curvature varies along with the lane-changing path.

Note that a lane-changing curve Figure 5 is symmetric in its midpoint, because of the reversibility of the changing course. The curvature is zero at three points: the start point, the midpoint, and the end point. There are two maximal curvatures: one is positive and the other is negative.

## 5. RESULTS

All the results are collected on an Intel Core™ i7-3770 3.40-GHz CPU with a high-end independent graphics card. The road network is extracted from the real world, including flyovers, tunnels, and suspension bridges.

### 5.1. Scenarios

We simulate traffic congestion and the scenario described in the close-car-braking circumstance (see the numerical simulation in Section 5.2.1), lane-changing behavior, traffic at on/off ramps, and tunnel traffic passing through mountains. Please see the complementary videos (Supporting Information).

#### 5.1.1. Traffic Jams.

From a practical perspective, jams usually occur when the leading vehicles decelerate for certain reasons. We simulate traffic congestion on this basis (Figure 6). A stop-and-go phenomenon also takes place along with traffic congestion.

#### 5.1.2. Close-car-braking Scenario.

As described earlier, FVDM cannot handle a close-car-braking circumstance; however, our method is able to cope with this problem. We animate the same traffic flow with FVDM and our technique. The visual results are presented in Figure 7.

#### 5.1.3. Lane Changes.

We merge the lane-changing behavior into our system, and Figure 8 shows the visual effects of the lane-changing behavior.

#### 5.1.4. Traffic at On/Off Ramps.

In reality, ramp traffic is very common for choosing different driving directions (Figure 9). In our road web, there are eight on-ramps and eight off-ramps; please see complementary videos (Supporting Information).

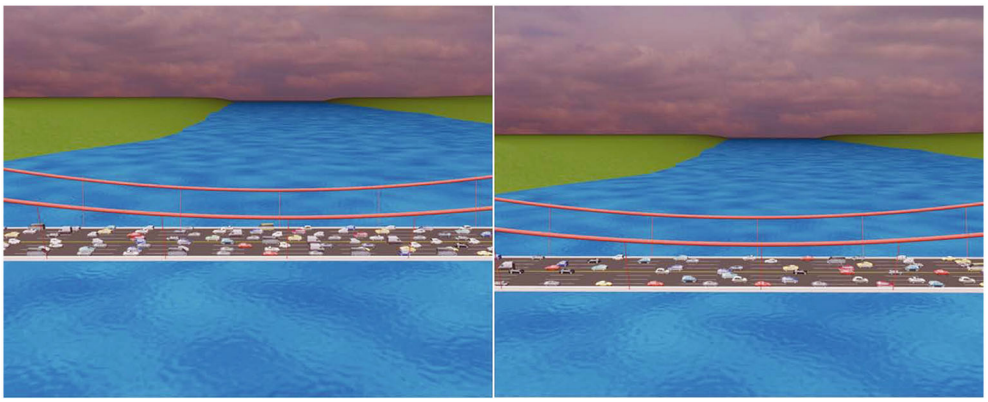
#### 5.1.5. Traffic in Tunnels.

Tunnel traffic is ubiquitous in rural places where mountains always lie. In this work, we also animate tunnel traffic passing through mountains (Figure 10).

### 5.2. Performance

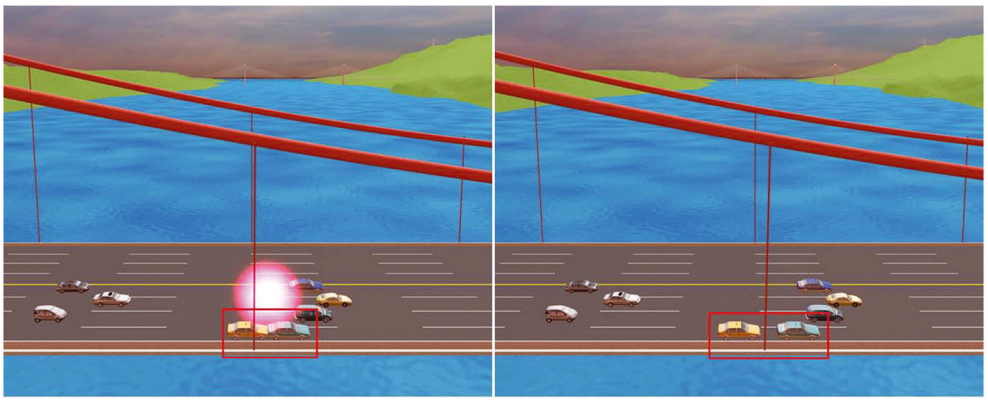
#### 5.2.1. Comparison with Full Velocity Difference Model.

The simulation scenario is described as follows: the velocities of the leader and the follower are 20.0 and 20.0 m/s at the beginning, and the initial netto interval (net spacing) between them is 10 m; the leading car decelerates at $-6$ m/s$^2$ until it completely stops and keeps stopping for several seconds before speeding up to 21 m/s. We



(a) Traffic jams. (b) Jams disappear.

**Figure 6.** Two snapshots about traffic congestion. (a) Leading cars decelerate for traffic lights or accidents in front, and jams occur. (b) Traffic flow becomes fluent again because leaders begin to accelerate and there is no block.



(a) Using FVDM. (b) Using our AA-FVDM.

**Figure 7.** Simulating scenarios for the close-car-braking circumstance with the full velocity difference model and our method under the same initial conditions. (a) Produces collisions. (b) Avoids collisions. FVDM, full velocity difference model; AA-FVDM, accident-avoidance full velocity difference model.

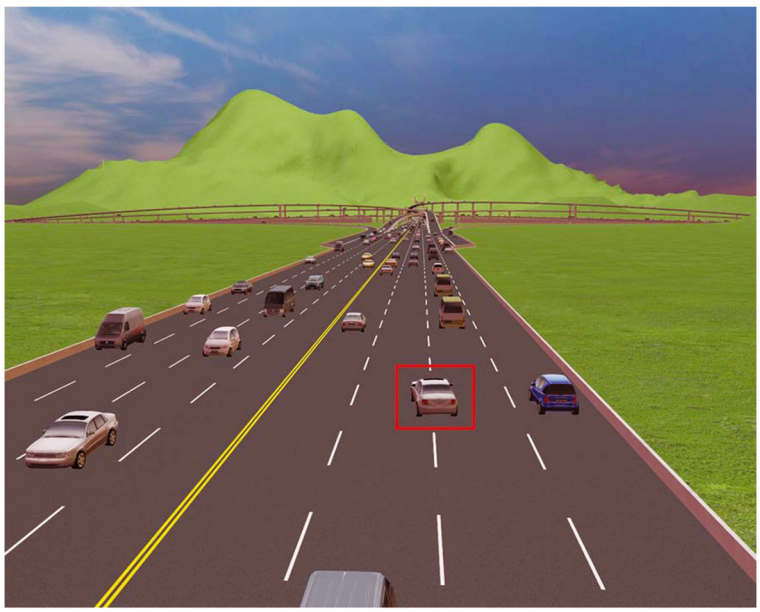


**Figure 8.** A car (enclosed in a red rectangle) performing a lane-changing behavior.

conduct two experiments under the same initial condition, employing FVDM and our method.

In such a situation, we can observe that FVDM is not capable of avoiding collisions or accidents, as it generates negative netto distance between the leading vehicle and the following vehicle, while the velocity of the follower is still positive. The leader and the follower get into an accident at 4.6 s in the FVDM (Figure 11).

We also test our method under the same situation as described earlier. The simulation results prove that the AA-FVDM is able to cope with such a problem without producing accidents (Figure 12). We can see from Figure 12 that the minimal netto distance is roughly 2 m over the whole process. The follower with a small interval can timely decelerate to zero and avoid collisions when the leader brakes steeply for an accident ahead.

### 5.2.2. Performance Tests.

First, we test the update frequency of our method with frames per second and memory usage. Then, we investigate how the percentage of lane-changing vehicles varies along with traffic density. Finally, we choose two typical parameters and perform some parameter tests, involving the probability variable $q$ (Section 4.3.1) and the combined


*Comp. Anim. Virtual Worlds* 2014; **25**:83–97 © 2013 John Wiley & Sons, Ltd.
DOI: 10.1002/cav

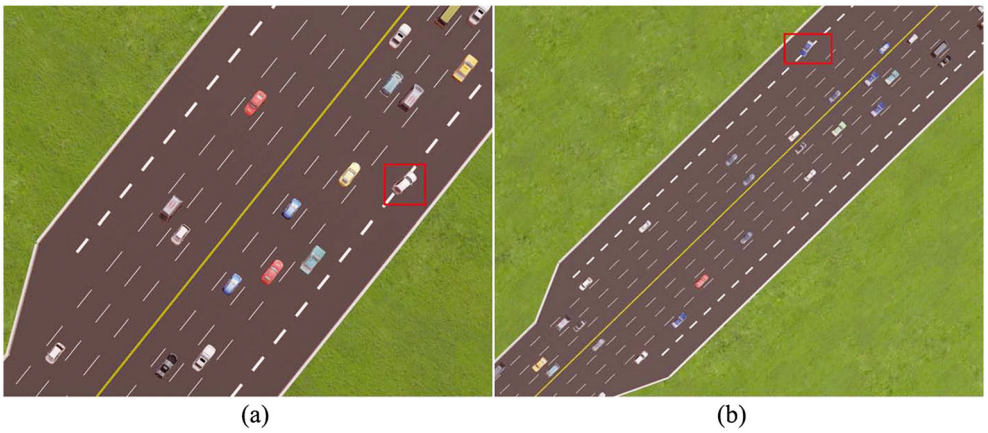



**Figure 9.** (a) A car (enclosed in a red rectangle) entering a ramp. (b) A car (enclosed in a red rectangle) exiting a ramp.

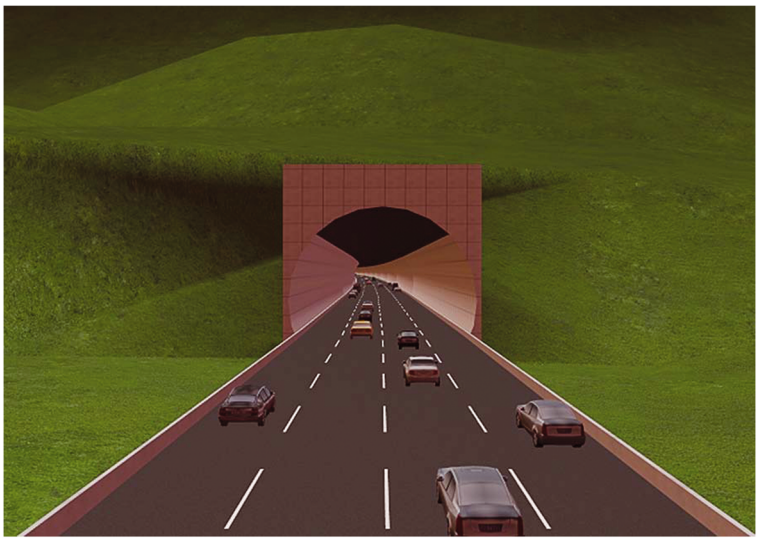


**Figure 10.** Vehicular traffic in tunnels passing through mountains.

radii $r_{i,i-1}$ (Section 4.2). Each test has been run for a constant period for 80 times.

Without taking lane-changing behavior into account (Figure 13), our method may update more than 185,000 vehicles in real time. The usage of memory increases linearly with the number of vehicles. An analogous linear growth can also be observed from Figure 13; however, when lane changes are considered, the maximum number of vehicles that can be updated at interactive rates is around 85,000, which is much smaller than that in Figure 13(a).

From real-world experience, we know that dense traffic will bring little opportunity for vehicles to perform lane changes. Our simulation results live up to this experience, which can be validated by Figure 14.

The first parameter we test is the probability variable $q$ (Section 4.3.1). We set the boundaries of $q$ as $[0.7, 1.0]$ in this study, and we test $q = 0.7$, $q = 0.8$, $q = 0.9$, and $q = 1.0$. It is not hard to see that if $q = 0.7$, only vehicles assigned $P \geq 0.7$ can perform lane changes (these vehicles must meet the safety and incentive criteria). If $q = 1.0$, it means that vehicles whose probability must be $P = 1.0$ may make lane changes. Therefore, a smaller $q$ results in a larger percentage of vehicles that perform lane-changing behavior (Figure 14).

The second parameter that we choose is the combined radii $r_{i,i-1}$. The difference of $r_{i,i-1}$ leads to different results. From our method, we can realize that a larger $r_{i,i-1}$ implies earlier repulsive forces between the two cars nearest to each other in the same lane. We test this idea in the same scenario, and the result (Figure 15) reveals that a larger $r_{i,i-1}$ leads to a greater gap between the leader and the follower.

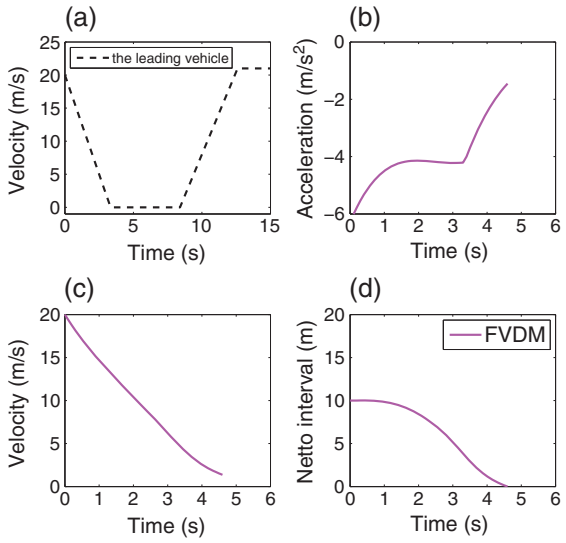



**Figure 11.** Simulations for the full velocity difference model (FVDM) in the close-car-braking circumstance. (a) Velocity of the leading vehicle, (b) deceleration of the following car, (c) speed of the following car, and (d) the netto interval between the leader and the follower.

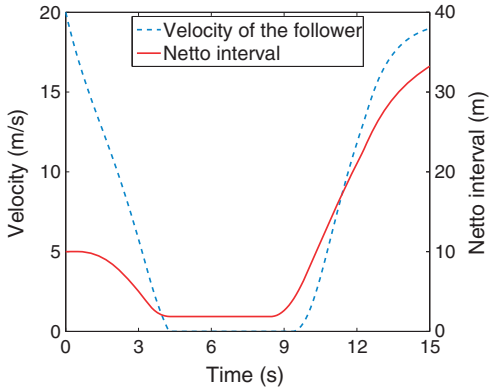



**Figure 12.** Simulations for our method—accident-avoidance full velocity difference model. With our method, the velocity of the follower is able to decelerate to zero to avoid collisions with the leader. The following car will accelerate if the leading vehicle moves again.

### 5.3. Validation with Real Traffic Data

In order to validate our method, we compute the standard deviations between simulation results and real-world traffic data for different traffic models and make a comparison with the outcomes. The Next Generation Simulation (NGSIM) project [35] is used as the real-world dataset source to firstly calibrate the parameters with respect to various traffic models including the GFM, FVDM, intelligent driver model [36], MITSIM (a well-known traffic simulator [37]), and our method—AA-FVDM. Secondly, the minimal standard deviation can be obtained for each model by fitting the simulation results to real-world data.

The NGSIM program was initiated by the US Department of Transportation Federal Highway Administration in the early 2000s and collected high-quality primary traffic and trajectory data. We utilize an effective calibration method called the trust region algorithm [38] because it provides a numerical solution to the problem of minimizing a function, generally nonlinear, over a space of parameters of the function and achieves an optimal parameter set. After the calibration process, it is necessary to make some comparison with the calibration results.

In our work, the US 101 dataset is a promising choice in the NGSIM datasets, because the data are collected on a segment of the US Highway 101 (Hollywood Freeway) in the Universal City neighborhood of Los Angeles, California, which matches our scenarios quite well. The vehicle trajectory data contain vehicle ID, frame ID, time, position, velocity, acceleration, preceding vehicle ID, space headway, and other information for each vehicle at a frame rate of 10 fps; see an example in Table 1.

In the following, we make a simple introduction for the calibration problem.

$$\begin{aligned} E(\sigma) &= f(e_1, e_2, ..., e_m; \xi_1, \xi_2, ..., \xi_n) \\ &= f\left(\vec{e}, \vec{\xi}\right) \end{aligned} \tag{17}$$

where $\vec{e} = (e_1, e_2, ..., e_m)$, $e_1, e_2, ..., e_m$ are independent variables, $\vec{\xi} = (\xi_1, \xi_2, ..., \xi_n)$, $\xi_1, \xi_2, ..., \xi_n$ are the population values of parameters, and $E(\sigma)$ is the expected value of the dependent variable $\sigma$.

Our aim is to achieve the parameter values $\vec{\xi} = (\xi_1, \xi_2, ..., \xi_n)$, so we need a set of empirical datum pairs of independent and dependent variables $(\vec{e}, \vec{\xi})$ so that the sum of the deviation squares $S(\vec{\xi})$ becomes minimal.

$$S(\vec{\xi}) = \sum_{i=1}^{N} \left[\sigma_i - f\left(\vec{e}, \vec{\xi}\right)\right]^2 \tag{18}$$

We set the initial values for parameters before the first iteration, and the iteration details can be referenced from [38]. The iteration will continue until there is a convergent value.

To evaluate the results of different traffic models, we bring in the standard deviation, which is also named root mean square error (RMSE). A smaller RMSE indicates a better agreement with real traffic data. RMSE is computed as

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N} (E(\sigma_i) - \sigma_i)^2}{N - 1}} \tag{19}$$

where $N$ denotes the sample size; in this study, we choose the vehicles' velocity as $\sigma_i$; $E(\sigma_i)$ is the computed value using traffic simulation models. Given the same initial information (position, velocity, etc.) as real-world vehicles and a traffic model, traffic evolves by itself. We know

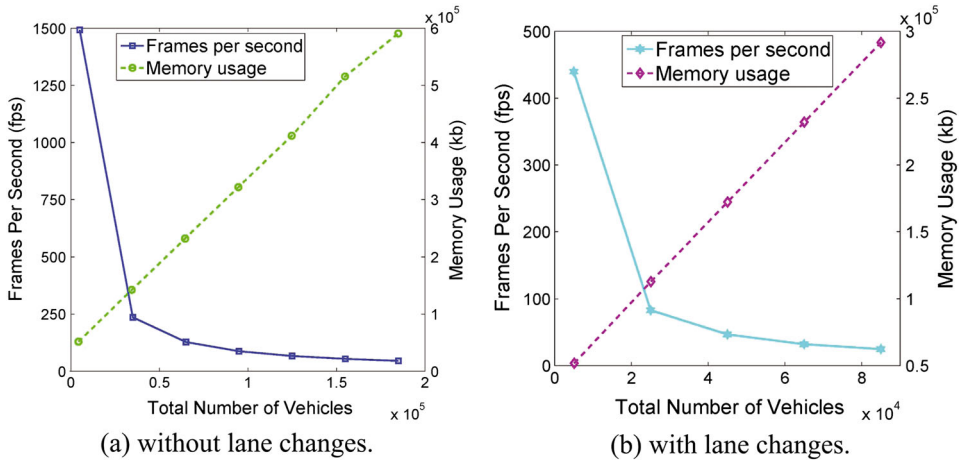



**Figure 13.** Frames per second and approximate memory usage without and with lane changes. (a) Frames per second and approximate memory usage for updating various numbers of vehicles without lane-changing behavior. The update frequency is about 45 fps when there are 185,000 vehicles. (b) Frames per second and memory usage for differing numbers of vehicles with lane-changing behavior. The update is roughly 25 fps when the number of vehicles is 85,000.

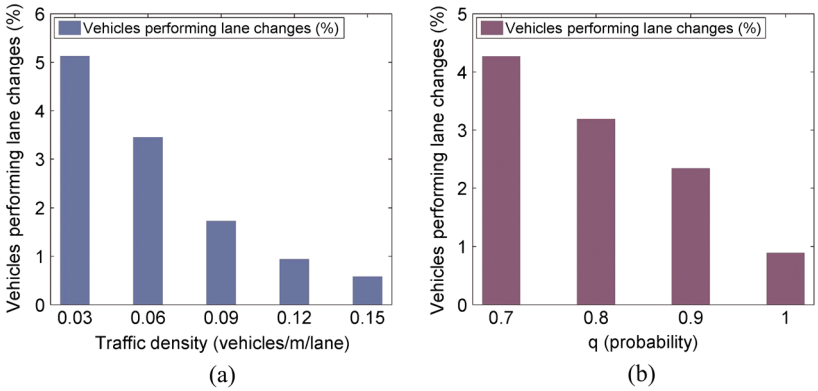



**Figure 14.** (a) The percentage of vehicles that perform lane changes along with traffic density. Vehicles performing lane changes occupy a greater percentage when traffic becomes sparse. (b) The impact of $q$ (probability) on the percentage of vehicles that make lane changes. A larger probability leads to a smaller lane-changing percentage of total vehicles.

that $v(t + \Delta t') = v(t) + a\Delta t'$; in the US 101 datasets, the time step is 0.1 s, which only induces a small variation in velocity; therefore, we choose 1 s as the time step size ($\Delta t' = 1$ s). Table 2 shows the RMSE of differing traffic models, and Table 3 exhibits the resulting optimal parameters for our method.

In Figure 16, we can observe that all models except MITSIM compare well with real traffic data. MITSIM has remarkable overshoots, which indicate too large accelerations. Our method has a similar trend as FVDM, because ours is on top of it. It is interesting that our method outperforms FVDM, even though there is little difference, as shown in Table 2. Table 3 demonstrates that the parameters from the optimal velocity function ($V(s_i)$) have analogous values to those in Helbing's work [13], which further validates our method.

## 6. CONCLUSION AND FUTURE WORK

We have set up a rural road web with diverse road structures including flyovers, suspension bridges, curving tunnels, and other straight or curved roads. Currently, our system simulates but not limited to different types of medium-size vehicles, allows for diverse speed limits on various road segments, and supports lane-changing behavior. Please note that our method can simulate different sizes of vehicles, as vehicle sizes do not make any difference. Our method can simulate but not limited to rural traffic and can also simulate urban traffic if we do some other work, for example, intersection control, calibration of model parameters using real-world urban traffic data.

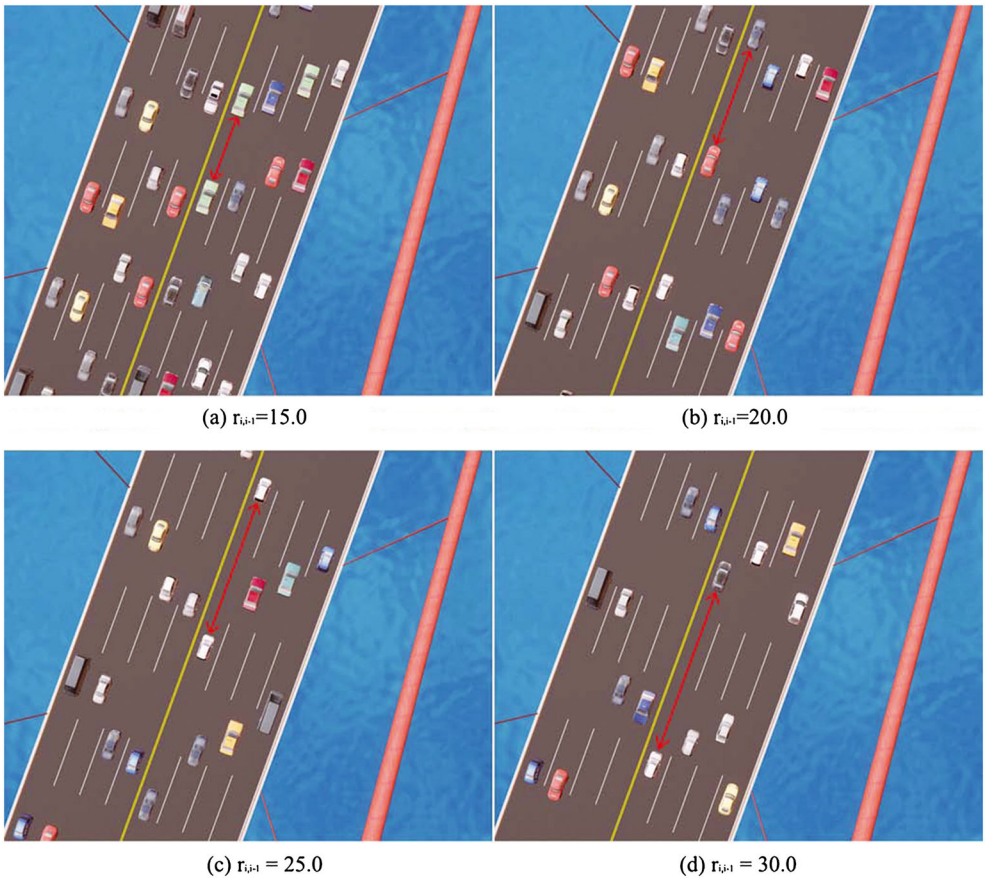



**Figure 15.** Parameter tests with different $r_{i,i-1}$; a longer $r_{i,i-1}$ results in a larger gap between the leading and following vehicles.

**Table 1.** A random extract from the US 101 dataset of vehicle trajectory data.

| Data | Value |
|---|---|
| Vehicle ID | 196 |
| Frame ID | 829 |
| Total frames | 474 |
| Local *X* (ft) | 18.937 |
| Local *Y* (ft) | 1560.577 |
| Vehicle length (ft) | 16.0 |
| Vehicle width (ft) | 6.4 |
| Vehicle velocity (ft/s) | 50.71 |
| Lane | 2 |
| Preceding vehicle | 184 |
| Following vehicle | 203 |
| Spacing (ft) | 103.82 |

**Table 2.** Minimal values of RMSE between real traffic data and simulation results that were reached for various traffic models by the trust region algorithm.

| | Model | | | | |
|---|---|---|---|---|---|
| | Ours | FVDM | IDM | GFM | MITSIM |
| RMSE | 0.479 | 0.504 | 0.590 | 0.604 | 1.007 |

FVDM, full velocity difference model; IDM, intelligent driver model; GFM, generalized force model; RMSE, root mean square error.

On the basis of FVDM, we present a novel method inspired by the force concept [33,34] and is able to address some emergency—the close-car-braking circumstance. A force perspective makes those two force terms easy and clear to interpret. However, the cost is that the new method may lead to overshooting deceleration when an emergency occurs. It is inevitable because when there is an emergency ahead, only with strong deceleration can the current car cut its speed quickly and avoid accidents. We also represent simple rules, the kinetics, and constraints on lane-changing behavior. While believable and realistic lane-changing behavior can be simulated with our technique, one restriction is that we do not consider vehicles' types when performing a lane change, for example, a truck should need a longer lane change than a small car.

Then, we conducted different experiments to test the AA-FVDM. Because of the agent-based property of our method, the update duration is much longer than that for


DOI: 10.1002/cav

**Table 3.** The optimal parameter values for our method (accident-avoidance full velocity difference model).

| | Parameters | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | $\kappa$ ($s^{-1}$) | $V_1$ (m/s) | $V_2$ (m/s) | $C_1$ ($m^{-1}$) | $C_2$ | $\lambda$ ($s^{-1}$) | $r_{i,i-1}$ (m) | $C$ ($m/s^2$) | $k$ ($s^{-2}$) |
| Values | 0.486 | 8.31 | 9.87 | 0.155 | 1.212 | 0.421 | 27.797 | 0.544 | 0.1 |

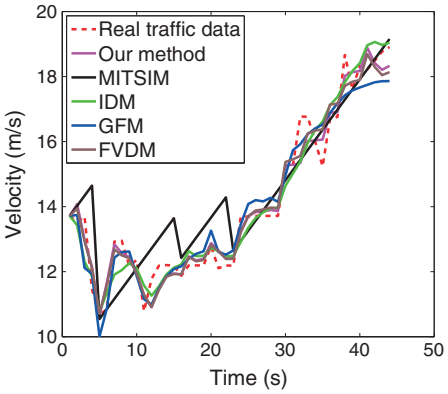



**Figure 16.** Comparison of real-world traffic data and simulated velocity using different traffic models. FVDM, full velocity difference model; IDM, intelligent driver model; GFM, generalized force model.

continuum-based models. However, our method can simulate anisotropic drivers and individualistic behavior with complex dynamics. Furthermore, the performance is fairly efficient, and it is able to simulate tens of thousands vehicles with our method at interactive rates. Besides, we validate our method using real-world traffic data and compare with other traffic methods. The matching results show that our method remarkably outperforms others (FVDM, intelligent driver model, GFM, and MITSIM). However, there are still some limitations in our current framework. In line with the reality that passing rules are different in diverse areas such as the USA, China, and Germany, lane changes can be classified into two classes: asymmetric and symmetric. We plan to simulate both types in different scenarios. Another limitation is that our current approach does not support path planning for an individual vehicle, and vehicles choose random directions at intersections. Although some efforts were carried out to simulate both pedestrians and traffic flows [39], they primarily focused on pedestrians and only presented simple traffic flows. As a future work, we plan to investigate novel schemes to simulate both crowds and vehicles in great detail in complex dynamic environments.

## ACKNOWLEDGEMENTS

This research is supported by National Science and Technology Support Program (Grant 2013BAH23F01), Natural Science Foundation of China (Grant 60970125 and 61202207), China Postdoctoral Science Foundation (Grant 2012M520067 and 2013T60706), Research Fund for the Doctoral Program of Higher Education of China (Grant 20124101120005), and the Open Project Program of the State Key Lab of CAD&CG at Zhejiang University (grant A1209).

## REFERENCES

1. Sumo—simulation of urban mobility, 2012. http://sumo.sourceforge.net/.
2. Lighthill MJ, Whitham GB. On kinematic waves. II. A theory of traffic flow on long crowded roads. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences* 1955; **229**(1178): 317–345.
3. Richards PI. Shock-waves on the highway. *Operations Research* 1956; **4**(1): 42–51.
4. Payne HJ. Models of freeway traffic and control, In *Mathematical Models of Public Systems*, 1971; 51–60.
5. Aw A, Rascle M. Resurrection of "second order" models of traffic flow. *SIAM Journal on Applied Mathematics* 2000; **60**(3): 916–938.
6. Zhang HM. A non-equilibrium traffic model devoid of gas-like behavior. *Transportation Research Part B-Methodological* 2002; **36**(3): 275–290.
7. Sewall J, Wilkie D, Merrell P, Lin MC. Continuum traffic simulation. *Computer Graphics Forum* 2010; **29**(2): 439–448.
8. Ngoduy D. Instability of cooperative adaptive cruise control traffic flow: a macroscopic approach. *Communications in Nonlinear Science and Numerical Simulation* 2013; **18**(10): 2838–2851.
9. Jiang R, Wu QS, Zhu ZJ. Full velocity difference model for a car-following theory. *Physical Review E* 2001; **64**(1): 017101.
10. Gerlough DL. Simulation of freeway traffic on a general-purpose discrete variable computer, *Ph.D. Thesis*, University of California, Los Angeles, 1955.
11. Newell GF. Nonlinear effects in the dynamics of car following. *Operations Research* 1961; **9**(2): 209–229.
12. Bando M, Hasebe K, Nakayama A, Shibata A, Sugiyama Y. Dynamical model of traffic congestion and numerical simulation. *Physical Review E* 1995; **51**(2): 1035–1042.
13. Helbing D, Tilch B. Generalized force model of traffic dynamics. *Physical Review E* 1998; **58**(1): 133–138.

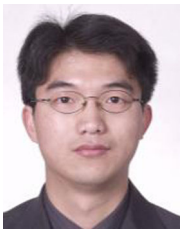
14. Algers S, Bernauer E, Boero M, Breheret L, Taranto CD, Dougherty M, Fox K, Gabard JF. Smartest project: review of micro-simulation models, 1997. EU project No: RO-97-SC 1059.
15. Helbing D. Traffic and related self-driven many-particle systems. *Reviews of Modern Physics* 2001; **73**(4): 1067–1141.
16. Nagel K, Schreckenberg M. A cellular automation model for freeway traffic. *Journal De Physique I* 1992; **2**(12): 2221–2229.
17. Kesting A, Treiber M, Helbing D. Enhanced intelligent driver model to access the impact of driving strategies on traffic capacity. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 2010; **368**(1928): 4585–4605.
18. Gipps PG. A model for the structure of lane-changing decisions. *Transportation Research, Part B (Methodological)* 1986; **20B**(5): 403–414.
19. Ahmed KI, Ben-Akiva M, Koutsopoulos HN, Mishalani. R G. Models of freeway lane changing and gap acceptance behavior, In *Proceedings of the 13th International Symposium on the Theory of Traffic Flow and Transportation*. Pergamon, New York, NY, 1996; 501–515.
20. Yang Q, Koutsopoulos HN. A microscopic traffic simulator for evaluation of dynamic traffic management systems. *Transportation Research Part C—Emerging Technologies* 1996; **4**(3): 113–129.
21. Kesting A, Treiber M, Helbing D. General lane-changing model mobil for car-following models. *Transportation Research Record: Journal of the Transportation Research Board* 2007; **1999**: 86–94.
22. Hidas P. Modelling vehicle interactions in microscopic simulation of merging and weaving. *Transportation Research Part C—Emerging Technologies* 2005; **13**(1): 37–62.
23. Moridpour S, Sarvi M, Rose G. Lane changing models: a critical review. *Transportation Letters: The International Journal of Transportation Research* 2010; **2**(3): 157–173.
24. Go J, Vu TD, Kuffner JJ. Autonomous behaviors for interactive vehicle animations. *Graphical Models* 2006; **68**(2): 90–112.
25. Sewall J, Van Den Berg J, Lin MC, Manocha D. Virtualized traffic: reconstructing traffic flows from discrete spatiotemporal data. *IEEE Transactions on Visualization and Computer Graphics* 2011; **17**(1): 26–37.
26. Sewall J, Wilkie D, Lin MC. Interactive hybrid simulation of large-scale traffic. *ACM Transactions on Graphics* 2011; **30**(6): 12 pages.
27. Shen J, Jin X. Detailed traffic animation for urban road networks. *Graphical Models* 2012; **74**(5): 265–282.
28. Prigogine I, Andrews FC. A Boltzmann-like approach for traffic flow. *Operations Research* 1960; **8**(6): 789–797.
29. Nelson P, Bui DD, Sopasakis A. A novel traffic stream model deriving from a bimodal kinetic equilibrium, 1997.
30. Shvetsov V, Helbing D. Macroscopic dynamics of multilane traffic. *Physical Review E* 1999; **59**(6): 6328–6339.
31. Nagel K, Wolf DE, Wagner P, Simon P. Two-lane traffic rules for cellular automata: a systematic approach. *Physical Review E* 1998; **58**(2): 1425–1437.
32. Huang DW. Lane-changing behavior on highways. *Physical Review E* 2002; **66**(2): 026124.
33. Helbing D, Molnar P. Social force model for pedestrian dynamics. *Physical Review E* 1995; **51**(5): 4282–4286.
34. Helbing D, Farkas I, Vicsek T. Simulating dynamical features of escape panic. *Nature* 2000; **407**(6803): 487–490.
35. NGSIM—Next Generation Simulation, 2012. http://ngsim-community.org/.
36. Treiber M, Hennecke A, Helbing D. Congested traffic states in empirical observations and microscopic simulations. *Physical Review E* 2000; **62**(2): 1805–1824.
37. MITSIM. MIT Intelligent Transportation Systems, 2012. http://mit.edu/its/mitsimlab.html.
38. Coleman TF, Li Y. An interior trust region approach for nonlinear minimization subject to bounds. *SIAM Journal on Optimization* 1996; **6**(2): 418–445.
39. Treuille A, Cooper S, Popovic Z. Continuum crowds. *ACM Transactions on Graphics* 2006; **25**(3): 1160–1168.

## AUTHORS' BIOGRAPHIES

**Xuequan Lu** is a Ph.D. candidate in the College of Computer Science and Technology at Zhejiang University, China. His research interests include crowd-related simulation, computer animation and computer graphics.

**Wenzhi Chen** born in 1969, received his Ph.D. degree from Zhejiang University, Hangzhou, China. He is now a Professor and Ph.D. supervisor of the College of Computer Science and Technology of Zhejiang University. His areas of research include computer graphics, computer architecture, system software, embedded system, and security.

**Mingliang Xu** is a lecturer in the School of Information Engineering at Zhengzhou University, China, and the secretary of the VR Committee for the China Society of Image and Graphics. His research interests include computer animation, virtual and augment reality, and mobile computing. Xu has a Ph.D. in computer science from the State Key Lab of CAD&CG at Zhejiang University.



**Zonghui Wang** , born in Mar. 1979, is a lecturer in the college of Computer Science and Engineering at Zhejiang University in Hangzhou, China. He received his Ph.D. from the College of Computer Science and Technology at Zhejiang University in 2007. His research interests focus on cloud computing, distributed system, computer architecture, and computer graphics.

**Zhigang Deng** is currently an Associate Professor of Computer Science at the University of Houston (UH) and the Founding Director of the UH Computer Graphics and Interactive Media (CGIM) Lab. His research interests include computer graphics, computer animation, virtual human modeling and animation, and human–computer interaction. He earned his Ph.D. in Computer Science at the Department of Computer Science at the University of Southern California in 2006. Prior to that, he also completed his B.S. degree in Mathematics from Xiamen University (China) and M.S. in Computer Science from Peking University (China). He is a senior member of IEEE, a member of ACM, and a Founding Board member of the International Chinese Association of Human Computer Interaction.

**Yangdong Ye** is currently a professor in the School of Information Engineering at Zhengzhou University, China. His research interests include database systems, machine learning, and intelligent systems. Ye has a Ph.D. in computer science from the China Academy of Railway Science.