A vehicle model for micro-traffic simulation in dynamic urban scenarios

Conference Paper *in* Proceedings - IEEE International Conference on Robotics and Automation · June 2011

DOI: 10.1109/ICRA.2011.5980101 · Source: IEEE Xplore

Wenda Xu
Carnegie Mellon University

Wen Yao
Peking University

Huijing Zhao
Peking University

2011 IEEE International Conference on Robotics and Automation
Shanghai International Conference Center
May 9-13, 2011, Shanghai, China

A Vehicle Model for Micro-Traffic Simulation in Dynamic Urban Scenarios

Wenda Xu, Wen Yao, Huijing Zhao and Hongbin Zha

Abstract— **In order to improve energy efficiency of transport systems, eco-driving strategies are studied world-widely. However, most literatures on eco-driving based on traditional traffic flow models, are greatly simplified, and can not evaluate the effects on detailed driving behaviors. By referring to robot motion planning approaches, in this research a microscopic vehicle model is developed and it can represent different driving behaviors, such as aggressive or conservative driving; a collision detection algorithm is proposed that takes O(1) time to check for a trajectory's collision, enabling realtime planning; and a traffic simulation system is developed by incorporating traffic rules, so that the driving behaviors such as observing or not observing traffic rules can also be represented. Experiments are conducted on the simulation platform, and the performance of different driving behaviors on travel time, mileage, comfort and eco is studied.**

I. INTRODUCTION

A. Motivation

Over the last two decades, greenhouse gas emissions from transportation have continued to rise, while the contributions of other sectors, such as industrial processes and agriculture, have gone down. This continuous rise is due to the significantly increased number of vehicles produced in this period and to the traffic congestion that is frequently observed in major cities all over the world. Although improvements in energy efficiency and vehicle design technology have reduced the fuel consumption of vehicles considerably, severe traffic congestion and variations in driving patterns have led to extra fuel consumption. For this reason, developing an optimal driving strategy is an important factor in improving the energy efficiency of transport systems.

Eco-driving is such a strategy. For the purpose of minimizing the impact of transportation on the environment, eco-driving has become an active area of worldwide research topic [1]. Eco-driving primarily consists of a variety of driving techniques that save fuel and lower emissions. Eco-driving programs attempt to change a driver's behavior through general advice, such as shifting gears as soon as possible, maintaining a steady and moderate speed, anticipating traffic flow and decelerating smoothly. Eco-driving is a measure that is mainly designed to reduce fuel consumption and the emission of carbon dioxide, but it is also effective at reducing travel time and costs associated with accidents due to the greater anticipation and less erratic and unpredictable behavior of drivers [2].

This work is partially supported by the NSFC Grants (No.90920304 and No.60975061)

W.Xu, W.Yao, H.Zhao and H.Zha are with the State Key Lab of Machine Perception (MOE), Peking University, Beijing, P.R. China. E-mail: xuwenda@cis.pku.edu.cn

In most of the literatures about eco-driving, only speed ([3], [4]) or both speed and gear [5] are considered as the decisive factors for explaining fuel consumption and greenhouse gas emissions, while most of the simulation systems use simplified traditional traffic flow model, which reduce the credibility of the simulation results. In this paper, a more accurate vehicle model and simulation system is developed that can evaluate eco-performance on different driving behaviors and lead to a more comprehensive eco-driving advice.

Traditional traffic flow theory for microscopic traffic behavior focuses on vehicle flux, the relationship between speed and density, headway distribution in lanes or at intersections, and vehicle behavior like following, lane changing and overtaking. There are numerous techniques used to simulate microscopic traffic, including cellular automata ([6], [7]) and agent-based methods ([8], [9]). However, most of these models are over-simplified, ignoring vehicle kinematic and dynamic constraints, so that vehicle statuses like displacement, velocity and acceleration cannot be obtained in realtime.

As a consequence, it is needed to build a precise microscopic model to describe the traffic behavior. Based on this requirement, we construct a behavior model of vehicle in urban environment referring to robot motion planning approaches. This paper presents a new vehicle model that can achieve realtime motion planning. Based on this model we examine the behavior of the vehicle which is useful for evaluating eco-driving, as well as other performance indexes such as travel time, mileage, comfort and eco.

B. Related work

To obtain more realistic simulation, we need to consider the difficulties in motion planning in dynamic urban environment.

First, the planning time and replanning interval must be short enough to ensure driving safety because the environment changes constantly. In addition, the computation cost of collision detection algorithm is widely accepted as one of the bottlenecks in realtime processing.

Second, the curvature and velocity of the trajectory must be continuous over time. Here, we emphasize the concept of trajectory. A trajectory is the path a moving object follows through space as a function of time. Many search-based methods such as D* [10] and AD* [11], can only generate paths, not trajectories. To get trajectories, we need to plan the speed based on the path, which will extend the planning time. The velocity obstacles (VO) ([12], [13]) presented by Fiorini can generate trajectories. VO uses the speed

978-1-61284-380-3/11/$26.00 ©2011 IEEE

information to determine potential conflicts. It represents a set of speeds, and a robot will collide with obstacles at certain time if a speed in the set is chosen. Simply by choosing the speed outside of VO and satisfying the dynamic constraints, we can get a feasible speed and then a collision-free trajectory. However, the curvature and velocity of a trajectory generated by VO are discontinuous over time, so VO is also inapplicable to vehicle motion planning in dynamic urban environment.

Third, to make the simulation more realistic, the trajectory should be consistent with the behaviors of drivers, and should, therefore, observe the traffic rules. For example, the vehicle orientation is generally parallel to the tangent direction of a center line. This requires that path generation be combined with the center line, which is not considered by most existing methods. [14] uses a robot motion planning approach to reconstruct and visualize continuous traffic flows and satisfy the above three requirements. First, this approach takes into account the kinematic and dynamic constraints of each vehicle. In addition, the approach constrains the motion of the vehicle to a pre-computed roadmap, and it searches for an optimal trajectory in the State-Time Space [15], so that the velocity of the trajectory is continuous over time. However, a vehicle can only travel on a predefined roadmap and the acceleration and lane-change curves are discretized. These simplifications reduce the realism of the simulation. Takahashi [16] and Papadimitriou [17] both use the polynomial as a function of time to represent the displacement. This approach can complete planning in a short time, and it can meet the demand for a continuous curvature and speed over time. However, the trajectory generated by this approach has no direct relation with the center line. This method can not guarantee that the vehicle orientation corresponds (is close) to the tangent direction of every shape of center line, so it cannot meet the requirements. On the other hand, this approach provides us with a feasible direction and it can meet the requirements with some simple improvements. Werling [18] applies the concept of the Frenet Frame to the method mentioned above. The Frenet Frame is a moving reference frame of two orthonormal vectors which are used to describe a curve locally at each point in a two-dimensional plane. Based on the Frenet Frame, we can use a polynomial as the function of time to represent the lateral and longitudinal movement. This method meets all of the requirements mentioned above.

By referring to robot motion planning methods, and especially the trajectory generation approach by [18], this research propose a microscopic vehicle model that can represent different driving behaviors, such as aggressive or conservative driving; a collision detection algorithm takes O(1) time to check for a trajectory's collision, so that it enables realtime planning; and a traffic simulation system that incorporate traffic rules, so that the driving behaviors such as observing or not observing traffic rules can also be represented.

The rest of this paper is organized as follows. First, we describe our motion planning approach in Section II, including a kinematic model of a car-like vehicle, the trajectory generation method and the collision detection methods. In Section III, we discuss the details of our traffic simulation platform, and we present traffic behavior evaluation in Section IV. Finally, we conclude and discuss future work in Section V.

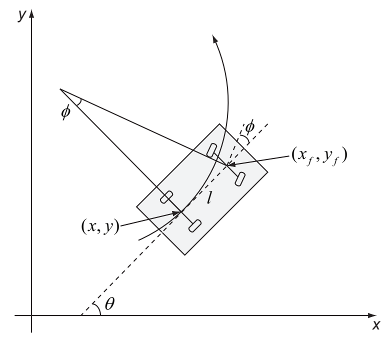


Fig. 1. The kinematic model of a car-like vehicle. (x, y) are the coordinates of the rear axle midpoint, θ is the orientation of car, ϕ is the steering angle and l is the wheelbase.

II. MOTION PLANNING IN DYNAMIC URBAN SCENARIOS

This section introduces the details of our motion planning approach. First, a kinematic model of a car-like vehicle that satisfies the non-holonomic constraint is reviewed. Then an optimal expression is obtained by using the optimal control theory. Third, the expression is extended to the center line of any road shape through the Frenet Frame. Finally the collision detection method is presented.

A. Kinematic model of a car-like vehicle

We will discuss the kinematic model of a car-like vehicle from the viewpoint of optimal control. The configuration of a car is defined by (x, y, θ, ϕ) where (x, y) are the coordinates of the rear axle midpoint, θ is the orientation of the car and ϕ is the steering angle (see Fig. 1). The vehicle is subject to non-holonomic constraints due to the rolling without slipping condition [19]. Denoting input $u = (v_1, v_2)$, where v_1 and v_2 are $driving$ and $steering$ velocity, and l the wheelbase respectively, then the kinematic model of a vehicle (rear-wheel) is:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} \cos\theta \\ \sin\theta \\ \tan\phi/l \\ 0 \end{bmatrix} v_1 + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} v_2 \qquad (1)$$

From a control viewpoint, $q = (x, y, \theta, \phi)$ are state variables, and $u = (v_1, v_2)$ are control variables. Eqn. 1 defines the state equation of vehicle motion. For the next step the cost function is determined. To travel along the trajectory smoothly, the vehicle's sudden acceleration change must be constrained. It also helps to approach the real trajectory. We choose the *jerk* (acceleration change) as an index and the cost function J is defined as the time integral of the square of *jerk* when the vehicle travels within the time interval T.

$$J = \frac{1}{2}\int_0^T L dt \tag{2}$$

where L is

$$L = \left(\frac{d^3x}{dt^3}\right)^2 + \left(\frac{d^3y}{dt^3}\right)^2 \tag{3}$$

The problem can be summarized as a nonlinear optimization problem as shown by Eqn. 4. The objective is to find T, $q(t)$ and $u(t)$ to minimize cost function J while subject to state equation Eqn. 1 and boundary constraints.

$$\begin{aligned} \text{find} \quad & T, q(t), u(t) \\ \text{minimizing} \quad & J(u(t), q(t), T) \\ \text{subject to} \quad & \text{Eqn. 1} \\ & q(0) = q_{start}, q(T) = q_{goal} \end{aligned} \tag{4}$$

There are three choices for solving this problem [20]: parameterize the trajectory $q(t)$ [18]; parameterize the control $u(t)$ [21]; or parameterize both $q(t)$ and $u(t)$. If $q(t)$ is parameterized, the trajectory can be solved directly, and the control $u(t)$ is calculated using Eqn. 1. If $u(t)$ is parameterized, it can be solved directly. In this way controlling the vehicle will be easy, but calculating the state $q(t)$ requires integrating the equations of motion (Eqn. 1). Because our goal is to simulate traffic behavior and we do not need to control the vehicle, we choose to parameterize the trajectory $q(t)$. A cost function similar to the form of Eqn. 2 can be solved by Euler-Lagrange equations. The cost function J is analytically assumed to be an extremum when $x(t)$ and $y(t)$ are the solutions of the following Euler-Lagrange equations.

$$\begin{aligned} \frac{\partial L}{\partial x} - \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{x}}\right) + \cdots + (-1)^n \frac{d^n}{dt^n}\left(\frac{\partial L}{\partial (x^{(n)})}\right) = 0 \\ \frac{\partial L}{\partial y} - \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{y}}\right) + \cdots + (-1)^n \frac{d^n}{dt^n}\left(\frac{\partial L}{\partial (y^{(n)})}\right) = 0 \end{aligned} \tag{5}$$

From the above equations and the performance index L, the following equations are derived.

$$\begin{aligned} \frac{d^3}{dt^3}\left(\frac{\partial L}{\partial (x^{(3)})}\right) = 0 \\ \frac{d^3}{dt^3}\left(\frac{\partial L}{\partial (y^{(3)})}\right) = 0 \end{aligned} \tag{6}$$

Then

$$\begin{aligned} \frac{d^6x}{dt^6} = 0 \\ \frac{d^6y}{dt^6} = 0 \end{aligned} \tag{7}$$

Consequently, we can conclude that both $x(t)$ and $y(t)$ are quintic polynomials. Because the first and second deviations of a quintic polynomial are continuous, the curvature and velocity of the trajectory generated by this method must be continuous over time. However, this trajectory can not satisfy the requirement that the angle between the direction of the trajectory and the center line should not be too large at any given time. Next, the Frenet Frame is used to achieve this requirement.

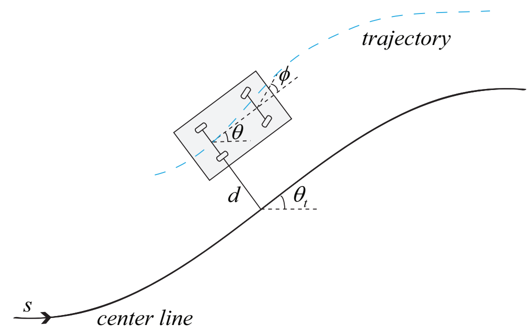


Fig. 2. Vehicle kinematic in Frenet Frame. s and d denote the longitudinal and lateral displacement, respectively. Let θ be the orientation of the vehicle, θ_t the angle between the the center line and x-axis, $\theta_p = \theta - \theta_t$ the difference in heading between the path and the vehicle, ϕ the steering angle, $\kappa = \frac{d\theta_t}{ds}$ the curvature of the center line.

B. Trajectory generation in the Frenet Frame

The Frenet Frame is a moving reference frame consisting of two orthonormal vectors. The trajectory of the vehicle can be decomposed into lateral and longitudinal (parallel and perpendicular to the center line respectively) movement by using the Frenet Frame (see Fig. 2). s and d denote the longitudinal and lateral displacement, respectively. Let θ be the orientation of the vehicle, θ_t the angle between the the center line and x-axis, $\theta_p = \theta - \theta_t$ the difference in heading between the path and the vehicle, ϕ the steering angle, $\kappa = \frac{d\theta_t}{ds}$ the curvature of the center line. The kinematic equations in terms of path coordinates $q_p = (s, d, \theta_p, \phi)$ in the Frenet Frame are derived as follows [19].

$$\begin{bmatrix} \dot{s} \\ \dot{d} \\ \dot{\theta}_p \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} \frac{\cos\theta_p}{1-d\kappa} \\ \sin\theta_p \\ \frac{\tan\phi}{l} - \frac{\kappa\cos\theta_p}{1-d\kappa} \\ 0 \end{bmatrix} v_1 + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} v_2 \tag{8}$$

Eqn. 1 and Eqn. 8 establish the transformational relation between the Frenet Frame coordinates (s, d) and the real coordinates (x, y). By replacing the state q in Eqn. 4 with q_p, the problem becomes parameterizing the state q_p. Similar to Eqn. 5 - Eqn. 7, we conclude that both $d(t)$ and $s(t)$ are quintic polynomials.

$$\begin{cases} d(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4 + a_5 t^5 \\ s(t) = b_0 + b_1 t + b_2 t^2 + b_3 t^3 + b_4 t^4 + b_5 t^5 \end{cases} \tag{9}$$

Trajectory generation method in this paper is developed by extending the method in [18]. The generation of lateral displacement $d(t)$ is considered first. The constraints of the start state $(d_0, \dot{d}_0, \ddot{d}_0)$ and end state $(d_T, \dot{d}_T, \ddot{d}_T, T)$ make up six linear equations. It is then easy to solve for the coefficients of $a_0 \dots a_5$. The end state is set to equal to $(d_T, 0, 0, T)$ as vehicles are expected to move parallel to the center line. By combining different discretized end states T and d_T, an entire trajectory set of lateral displacement is obtained.

The generation of longitudinal movement $s(t)$ is similar to the lateral one. However, $s(t)$ cannot be discretized due to the infinite longitudinal movement. If we want to generate a set of longitudinal trajectories independent of the positions of vehicles in the vicinity, we can only discretize the longitudinal speed $\dot{s}_T$. Here, the start state is $(s_0, \dot{s}_0, \ddot{s}_0)$ and the end state is $(\dot{s}_T, \ddot{s}_T, T) = (\dot{s}_T, 0, T)$, because we want to reach a constant speed.

Finally, we can calculate the position, orientation, curvature and other states of the final trajectory by combining the lateral and longitudinal trajectory sets through the use of Eqn. 8.

C. Collision detection

We develop a fast collision detection method extending the method from [22] in this subsection. One of the difficulties in this problem is that typical car-like vehicles (generally rectangular) cannot easily be approximated by a rotationally invariant disk shape, so the orientation of vehicle need to considered. Since collision detection for two circles is very simple, firstly, the vehicle shape is decomposed by several circles.

1) Representation of vehicle shape by circles: A car-like vehicle is approximated as a rectangular shape, but it can be reasonably approximated by a set of overlapping circles [22], as demonstrated by Fig. 3. A rectangle of length l and width w can be covered by n circles of radius with $r = \sqrt{\frac{l^2}{n^2} + \frac{w^2}{4}}$ and $d = 2\sqrt{r^2 - \frac{w^2}{4}}$.

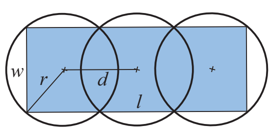


Fig. 3. Representation of vehicle shape by circles.

2) Collision detection in the Frenet Frame: After decomposing the vehicle shape into several disks, collisions can be easily determined by calculating the distance between two centers. However, trajectory generation is in the Frenet Frame while collision detection is in the Cartesian coordinates. Calculating Cartesian coordinates (x, y) for every point on the trajectory coordinate takes a lot of time. The following derivation will show that detecting collisions in the Frenet Frame directly is feasible. Using a fair approximation of reality, we will prove that it only takes O(1) time to check for a trajectory's collision.

Proof: Assuming that (s_1, d_1) and (s_2, d_2) are the coordinates of two vehicles in the Frenet Frame, while (x'_1, y'_1) and (x'_2, y'_2) are the coordinates in the Cartesian coordinates. (x_1, y_1) and (x_2, y_2) correspond to s_1 and s_2; the latter are longitudinal displacements along the center line. $\theta_1, \vec{t}_1, \vec{n}_1$ and $\theta_2, \vec{t}_2, \vec{n}_2$ are orientation, tangent vector, normal vector of the two vehicles respectively, as shown in Fig. 4. The coordinates of the vehicles in Cartesian coordinates can be formulated as follows.

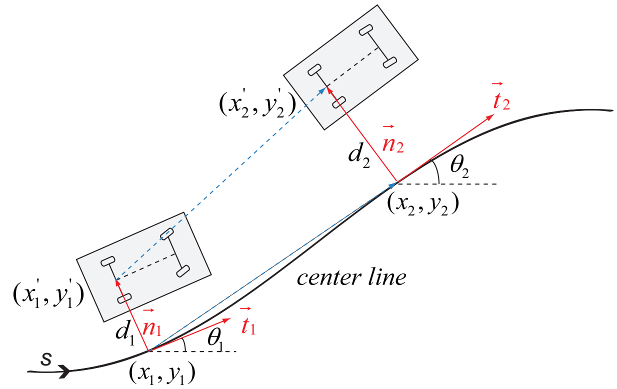


Fig. 4. Collision detection in the Frenet Frame. (s_1, d_1) and (s_2, d_2) are coordinates of two vehicles in the Frenet Frame, while (x'_1, y'_1) and (x'_2, y'_2) are coordinates in the Cartesian coordinates. (x_1, y_1) and (x_2, y_2) correspond to s_1 and s_2, the latter are longitudinal displacements along the center line. $\theta_1, \vec{t}_1, \vec{n}_1$ and $\theta_2, \vec{t}_2, \vec{n}_2$ are orientation, tangent vector, normal vector of two vehicles respectively.

$$
\begin{aligned}
(x'_1, y'_1) &= (x_1, y_1) + d_1 \vec{n}_1 \\
(x'_2, y'_2) &= (x_2, y_2) + d_2 \vec{n}_2
\end{aligned} \tag{10}
$$

It is easy to prove that the distance between (x_1, y_1) and (x_2, y_2) is less than $||s_1 - s_2||$ (A straight line is the shortest distance between two points). In addition, because the change of the road's tangential direction is generally small within a short distance (if two cars are far away from each other, they will not collide), we suppose $\theta_1 \approx \theta_2$. Therefore $\cos(\theta_1 - \theta_2) \approx 1$, and the following formulas are obtained.

$$
\begin{aligned}
&\|(x'_2, y'_2) - (x'_1, y'_1)\| \\
=&\|(x_2, y_2) + d_2 \vec{n}_2 - (x_1, y_1) - d_1 \vec{n}_1\| \\
=&\|(x_2 - x_1, y_2 - y_1) + (-d_2 \sin\theta_2 + d_1 \sin\theta_1, \\
&d_2 \cos\theta_2 - d_1 \cos\theta_1)\| \\
\leq&\|s_2 - s_1\| + \sqrt{d_1^2 + d_2^2 - 2 d_1 d_2 \cos(\theta_1 - \theta_2)} \\
\approx&\|s_2 - s_1\| + \|d_2 - d_1\|
\end{aligned} \tag{11}
$$

The collision-free condition for the two cars is that every pair of disks will not collide, that is:

$$
\|(x'_2, y'_2) - (x'_1, y'_1)\| \geq \|r_1 + r_2\| \tag{12}
$$

Finally, by combining Eqn. 11 and Eqn. 12, we get

$$
\|s_2 - s_1\| + \|d_2 - d_1\| - \|r_1 + r_2\| \geq 0 \tag{13}
$$

■

Collision in (s, d) plane can be checked using the result of the proof. A collision-free state is obtained if Eqn. 13 is constantly true for $t \in [0, T]$. $s(t)$ and $d(t)$ are quintic polynomials, thus the left part of Eqn. 13 is quintic polynomial too. The extreme values of a quintic polynomial can be obtained by solving the first derivative of itself, which is a quartic equation here. According to Ferrari's method, there are root formulas for a quartic equation. Consequently, for each trajectory it takes only O(1) time to check if host vehicle collides with another vehicle. Furthermore, in a freeway scenario, a vehicle only collides with its front and back vehicles in current lane and adjacent lanes, so the conclusion can be further expanded that it takes O(1) time to check if collision happens between host vehicle and all other vehicles.

III. MICRO TRAFFIC SIMULATION PLATFORM

In the previous section, a vehicle model is presented followed by the methods of trajectory generation and collision detection. The trajectory generation method enables a simulation of vehicle behavior (e.g., like following, changing lanes and parking) based on center line of any road shape, while the speed and curvature of trajectories remain continuous over time. As a result, this model is more realistic than the previous models. This section introduces the micro traffic simulation platform. Based on such a simulation system, different driving behaviors can be represented, and both the comfort and energy consumption of the vehicle can be evaluated simultaneously. Through the evaluation, we hope to give useful advice on vehicle driving.

Model design of simulation platform is introduced in detail as follows. It contains three models: environment model, vehicle model and rule model (Fig. 5). Among them, rule model is the premise of the whole system; they not only affect the definition of the environment model but also restrain the behavior of vehicles. Environment model includes the static environment, namely traffic channelization, and other traffic participants. In this research, traffic participants, as the dynamic factor of the environment, are simulated using traditional cellular automata ([6]) by given the parameters such as average flow speed, density, etc. Subject to the environmental and rule models, the vehicle model, containing vehicle states and actions, conducts simulation to host vehicles. Host vehicles can avoid other traffic participants while neither violate the traffic rules nor travel away from the road. Besides, it is able to simulate different driving behaviors, such as aggressive *vs.* conservative, and observing *vs.* not observing traffic rules.

A. Environment model

The environment model has two components: road structure and traffic participants. Our platform can simulate traffic behavior in urban environment, such as intersection and freeway. This paper takes freeway for example. The simulation environment according to rule model (as described in

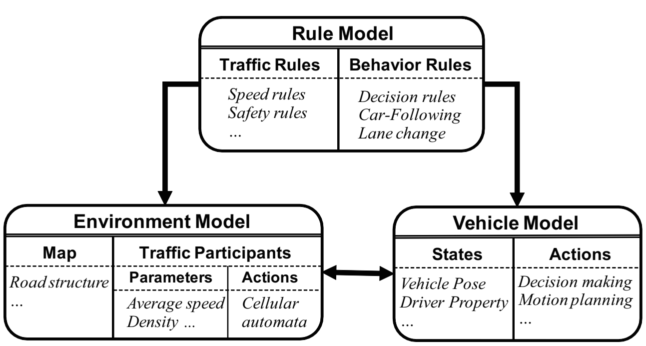


Fig. 5. Framework of micro traffic simulation platform.

Algorithm 1: Behavior Rule for Action Mode Decision

Input: $v_0\ v_t\ f_0\ b_0\ min_f\ s_f\ s_b\ p_c$;
Output: LANE_CHANGE or CAR_FOLLOW;
begin
  $v_0 \leftarrow$ current speed of host vehicle;
  $v_t \leftarrow$ target speed of host vehicle;
  $f_0 \leftarrow$ distance with front vehicle in current lane;
  $b_0 \leftarrow$ distance with back vehicle in current lane;
  $min_f \leftarrow$ minimum distance with front vehicles in adjacent lanes;
  $s_f \leftarrow$ safe distance with front vehicle;
  $s_b \leftarrow$ safe distance with back vehicle;
  $p_c \leftarrow$ probability of driver's intention to change lines without traffic condition constraints;
  if $v_0 < v_t$ *and* $f_0 \neq min_f$ **then**
    if $b_0 > s_b$ *and* $f_0 > s_f$ **then**
      with p_c **return** LANE_CHANGE;
  else
    return CAR_FOLLOW;
end

Section III-B) has at least two lanes: the left-most is passing lane, on which a vehicle cannot continuously drive more than a certain period; others are carriageways. There is a highest and a lowest allowed speed in each lane. The traffic participants (i.e. obstacle vehicles) are randomly generated on the basis of traffic density and average speed. Besides, the participant vehicles travel along a given lane following a cellular automata rule and will not change lane, and of course in accordance with the traffic rules. The host vehicles in simulation must avoid collision with the participant vehicles.

B. Rule model

1) Behavior rules: A host vehicle's driving can be parsed into a sequence of actions, which has mainly two modes: car following and lane changing. A behavior rule is defined for action mode decision. The one that implemented in this research is shown in Algorithm 1, where personal driving properties are represented by the parameters s_f, s_b and p_c. For example, aggressive vehicles have shorter safe distance and stronger motivation for lane changing.

2) Traffic rules: For freeway traffic rules, we refer to the Enforcement Regulations of the Road Traffic Safety Law of the People's Republic of China, which can be briefly described as follows.

- Speed rules
  - The highest speed allowed on the freeway is 120 km/h and the lowest is 60 km/h.
  - If there are more than four lanes in the same direction, the lowest speed is 110 km/h in the left-most lane, 90 km/h in the center lanes and 60 km/h in the right-most lane. In addition, the left-most lane is the passing lane, and the other lanes are carriageways.

- Safety rules
  - Only adjacent lanes can be used to overtake, and overtaking on the right is forbidden. After overtaking in the passing lane, the vehicle shall immediately drive back to the carriageway.
  - In the following two situations driving continuously is not allowed: driving on the lane line or driving in the passing lane.

C. Vehicle model

Driving can be generally divided into aggressive and conservative, which are greatly affected by the drivers' personality. This research intends to evaluate their impact through traffic simulation. In simulating different driving behaviors, the vehicle's target action (changing lane or following) is first decided by using the decision function (Algorithm 1) based on vehicle pose and driver property. The detail behavior is then planned by using our motion planning approach (Section II). In addition, it is important that vehicles strictly abide by traffic rules except an exceptional situation. That is when in a traffic jam, vehicles will unable to comply with the speed rules.

The flow chart of the motion planning algorithm is shown in the left part of Fig. 6, and example results are shown accordingly at the right part. They are explained in details below. A trajectory set is first generated by discretizing the end state (Fig. 6(a)). They are then sorted on cost (Eqn. 2). Their descending order are represented in color from yellow to red (Fig. 6(b)). The trajectory with the lowest cost is first selected, as shown in green (Fig. 6(c)). Two checks are then conducted to find whether the selected trajectory is in compliance with traffic rules and/or free of collision with other vehicle. Planning ends if both conditions are satisfied, otherwise another trajectory will be selected and examined from the remaining set of trajectories. In Fig. 6(d), a trajectory is abandoned after traffic rule or collision check (colored in blue), while a different trajectory is then selected and marked in green.

The results of traffic rule and collision check are further demonstrated using real simulation data in Fig. 7. There is one host vehicle in the simulation, which is shown in red. Other vehicles are traffic participants, which are shown in blue for neighboring ones, and gray for others. The set of trajectories are checked for its violation to traffic rules (Fig. 7(a)) or collision with other cars (Fig. 7(b)). In both sub-figures, invalid trajectories are denoted in blue, while others are in grey. The red line on road is the host vehicle's target center line. If they are at different lanes, it means that the host vehicle intends to change lane. In Fig. 7(a), the trajectories that passing from the right violate the traffic rules, so they are marked with blue; in Fig. 7(b), the trajectories collide with participant cars are marked with blue too.

IV. TRAFFIC BEHAVIOR-BASED EVALUATION

By using the simulation platform described above, microscopic traffic behavior evaluation is taken into account in urban environment under different settings. In the following,

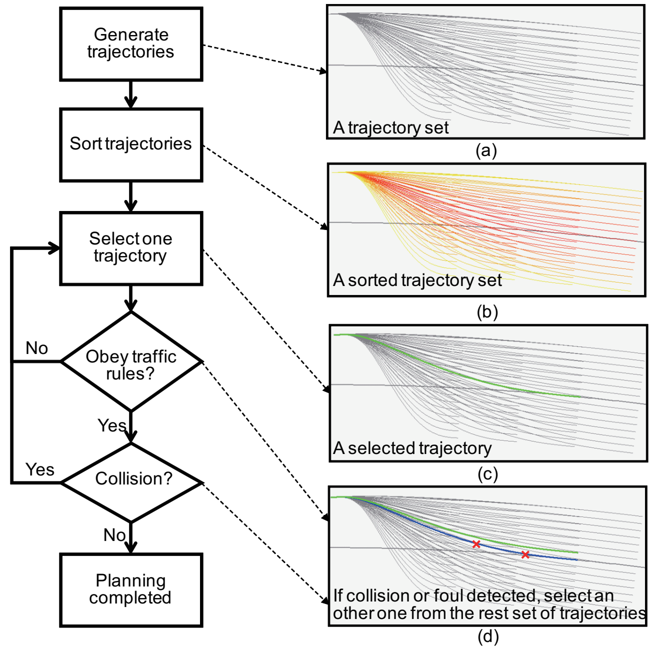


Fig. 6. Flow chart of motion planning.

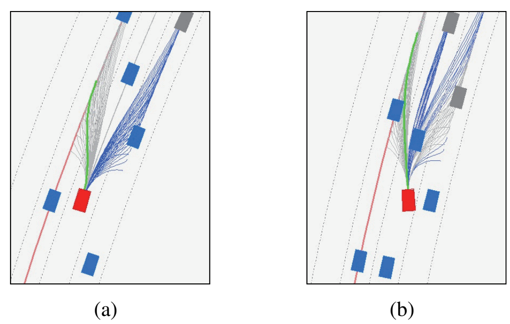


Fig. 7. Invalid trajectories (blue): (a) Violation of traffic rules; (b) Collision with other cars.

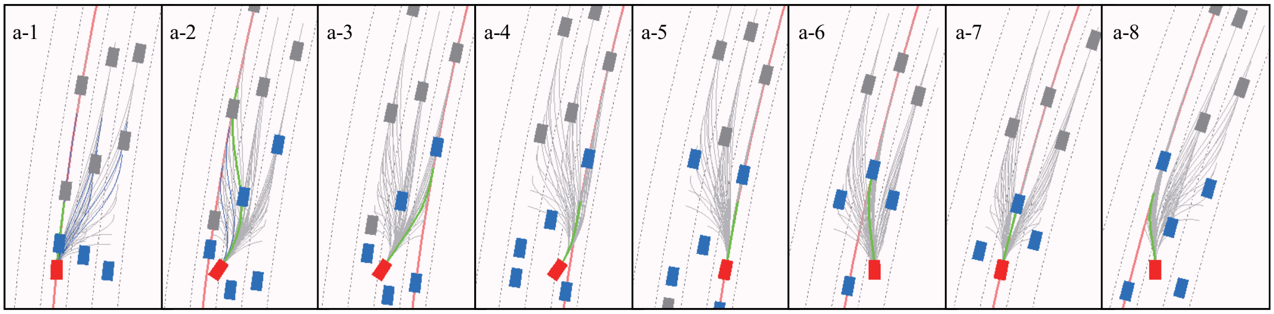
we show an example that compares adverse driving behaviors: aggressive *vs.* conservative. Given start and end states and based on the simulation platform, we comparatively evaluate travel time, mileage ($\int \sqrt{d^2 + s^2}dt$), discomfort (Eqn. 2) and energy consumption in the experiments. Among them, energy consumption is the most important performance index of eco-driving. This section firstly defines the performance index of energy consumption, then it introduces the experimental design and gives the experimental analysis and results.

A. Performance index of energy consumption

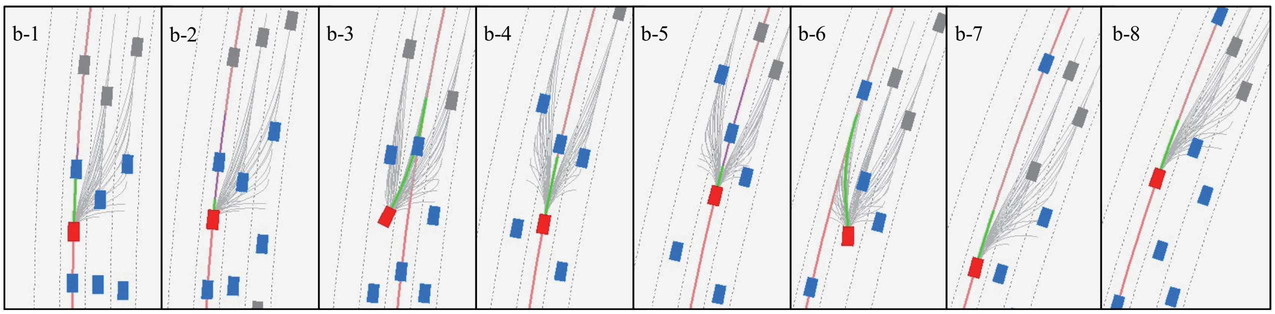
The simplest form of vehicle tractive power (kw) can be expressed as follows [4].

$$P_{tractive} = \frac{M}{1000}v(a + g\sin\theta) + \frac{v}{1000}(MgC_r + \frac{\rho}{2}v^2AC_a) \tag{14}$$

In the above formula, M is the vehicle mass (kg), v is the velocity(m/s), a stands for acceleration (m/s^2), g is the gravitational constant $(9.81m/s^2)$, θ is the road grade, C_r is the rolling resistance coefficient, ρ is air density $(1.225kg/m^3$,



(a) Aggressive driving behavior



(b) Conservative driving behavior

Fig. 8. Simulation snapshots in a traffic jam: (a) driving aggressively; (b) driving conservatively. The vehicle in red is a host vehicle, while others (blue and gray) are traffic participants. The green line in front of the host vehicle is the best valid trajectory, the blue ones are invalid, and gray ones are not checked trajectories. The dark red line in the road is the target center line of the host vehicle.

related to temperature and altitude), A is the windward area (m^2), and C_a is the air drag coefficient.

The following formula represents the conversion between traction power and engine power.

$$P_{engine} = \frac{P_{tractive}}{\eta_{tf}} + P_{accessories} \tag{15}$$

η_{tf} is the combined efficiency of the transmission and final drive, and $P_{accessories}$ is the engine power demand associated with the operation of accessories. We simply use P_{engine} as the total power of the vehicle and the energy consumption can be calculated by integrating the total power.

B. Simulation design

Two sequences of snapshots are given in Fig. 8, which are the simulation of two contradictory driving behaviors, i.e. aggressive (Fig. 8(a)) and conservative (Fig. 8(b)) driving, in heavy traffic conditions. We need to mention that in abnormal traffic situations, such as heavy traffic or jam, vehicle speeds are strongly restricted by their local traffic environment, thus are not subject to the speed rules that are defined in Section III-C. The same with Fig. 7, in each snapshot, the red car is the host one, while others are traffic participants. In different with Fig. 7, green trajectory is the one that is finally selected for vehicle motion; blue ones are those have lower costs than green while abandoned due to either violation to traffic rule or in collision with traffic participants; gray ones are those have high costs that are not examined at the end of motion planning. The red line on road is the host vehicle's target center line. When the host vehicle move forward, the participant vehicles near it will be marked with blue, and the others are marked with gray. Therefore, the colors of the participant vehicles are changing continuously. The lanes from right to left are Lane 1 to Lane 4 respectively.

As illustrated in Fig. 8(a), the host vehicle drives aggressively to get quicker speed and good driving conditions. In a-1 the host vehicle follow the front vehicle tightly on Lane 3, and then in a-2 it attempts to overtake the front vehicle aggressively. However, when it reaches Lane 2, it finds the driving conditions on Lane 1 is better, so it changes lane to Lane 1 as shown in a-3 and a-4. After that it follow the front vehicle on Lane 1 in a-5, and it changes lane to Lane 2 to get a quicker speed in a-6 while dose not consider the short distance with the back vehicle. Similar behavior appears in a-7 and a-8. Fig. 8(b) shows the behavior of a conservative host vehicle. In b-1 and b-2, it follows the front vehicle with a safe distance. Then in b-3, it changes lane to Lane 2 with an long distance with the back vehicle on Lane 2. Similarly, as shown from b-4 to b-6, it does not change lane to Lane 3 until there is enough safe distance with the back vehicle on Lane 3. Finally in b-7 and b-8, it follows the front vehicle and dose not change lane. Videos of this simulation can be found at **http://poss.pku.edu.cn/people/xuwd/**.

C. Evaluation

The experimental scenario is designed as a four-lane freeway with one-kilometer center line. The participant vehicles on each lane are randomly generated on the basis of given initial traffic density and average speed. As we want to see host vehicle's choice in critical scene, host vehicle's target speed is set as initial average speed of participant vehicles. In the experiment, the initial traffic density is fixed and

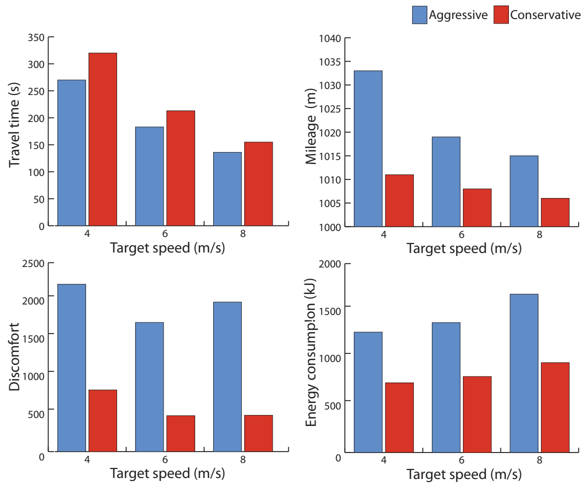


Fig. 9. Evaluation results: evaluation of travel time, mileage, comfort and energy consumption for different target speeds.

average speed has three different values. We simulate fifty times for aggressive and conservative driving behaviors respectively in each speed. Through realtime traffic simulation, the evaluation results in Fig. 9 reveal four examined indexes, which are travel time, mileage, discomfort and energy consumption. The horizontal coordinates stand for host vehicle's target velocity (m/s). The blue columns stand for the case driving aggressively while the red columns represent driving conservatively. The result clearly shows that, in each target speed, driving aggressively has shorter travel time, longer mileage, more discomfort, and more energy consumption. That is because host vehicles driving aggressively overtake participant vehicles at a higher frequency and have a greater tendency to reach the target speed. These results show that driving conservatively corresponds with eco-driving objectives. Furthermore, by using this platform, we not only obtain the realtime calculation of vehicle conditions such as position, velocity and acceleration, but also obtain the realtime evaluation of comfort and energy consumption. This analysis indicates that motion planning can be applied to all vehicles under our platform, though the experiments only used it for one vehicle in this paper. Applying motion planning for all vehicles will be included in future work.

V. CONCLUSIONS AND FUTURE WORKS

In this research a microscopic vehicle model is developed that can represent different driving behaviors, such as aggressive or conservative driving; a collision detection algorithm is proposed and it takes O(1) time to check for a trajectory's collision, enabling realtime planning; and a traffic simulation system is developed by incorporating traffic rules, so that the driving behaviors such as observing or not observing traffic rules can also be represented. Experiments in highway scene are conducted to evaluate eco-performance as well as other indexes using the simulation platform.

Future studies will be addressed on situation-based experiments in other dynamic urban scenarios, such as traffic jam vs. normal traffic at intersections, downtown streets, etc., the results of which are expected to give suggestions to a more comprehensive eco-driving strategy. A cross-validation of the simulation platform will also be conducted through future work using real driving data, so that to validate and improve simulation reliability.

REFERENCES

[1] N. L. J. Gense, "Driving style, fuel consumption and emissions," TNO Automotive, Delft, The Netherlands, Final report TNO report 00.OR.VM.021.1/NG, 2000.

[2] CIECA, "Internal project on eco-driving in category B driver training and the driving test," CIECA, Brussels, Final report, 2007.

[3] J. M. Starkey, S. Gray, and D. Watts, "Vehicle performance simulation and optimization including tire slip," *SAE Transaction*, no. 881733, 1988.

[4] M. Barth and K. Boriboonsomsin, "Energy and emissions impacts of a freeway-based dynamic eco-driving system," *Transportation Research Part D: Transport and Environment*, vol. 14, no. 6, pp. 400–410, 2009.

[5] Y. Saboohi and H. Farzaneh, "Model for developing an eco-driving strategy of a passenger vehicle based on the least fuel consumption," *Applied Energy*, vol. 86, no. 10, pp. 1925–1932, 2009.

[6] K. Nagel and M. Schreckenberg, "A cellular automaton model for freeway traffic," *J. Phys. I France*, vol. 2, pp. 2221–2229, 1992.

[7] K. Nagel, D. E. Wolf, P. Wagner, and P. Simon, "Two-lane traffic rules for cellular automata: A systematic approach," *Physical Review E*, vol. 58, no. 2, pp. 1425–1437, 1998.

[8] H. Wang, J. K. Kearney, J. Cremer, and P. Willemsen, "Steering behaviors for autonomous vehicles in virtual environments," in *IEEE Virtual Reality(VR)*, 2005.

[9] P. Willemsen, J. K. Kearney, and H. Wang, "Ribbon networks for modeling navigable paths of autonomous agents in virtual environments," *IEEE Transactions on Visualization and Computer Graphics*, pp. 331–342, 2006.

[10] A. Stentz, "Optimal and efficient path planning for partially-known environments," in *Proc. of The IEEE International Conference on Robotics and Automation (ICRA)*, vol. 4, May 1994, pp. 3310–3317.

[11] M. Likhachev, D. Ferguson, G. Gordon, A. Stentz, and S. Thrun, "Anytime dynamic A*: An anytime, replanning algorithm," in *Proceedings of the International Conference on Automated Planning and Scheduling(ICAPS)*, June 2005.

[12] P. Fiorini and Z.Shille, "Motion planning in dynamic environments using velocity obstacles," *The International Journal of Robotics Research*, vol. 17, no. 7, p. 760, 1998.

[13] J. van den Berg, M. Lin, and D. Manocha, "Reciprocal velocity obstacles for real-time multi-agent navigation," in *Proc. of IEEE Conference on Robotics and Automation*, 2008, pp. 1928–1935.

[14] J. Sewall, J. van den Berg, M. C. Lin, and D. Manocha, "Virtualized traffic: Reconstructing traffic flows from discrete spatiotemporal data," *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, pp. 26–37, 2011.

[15] T. Fraichard, "Trajectory planning in a dynamic workspace: a'state-time space'approach," *Advanced Robotics*, vol. 13, no. 1, pp. 75–94, 1998.

[16] A. Takahashi, T. Hongo, Y. Ninomiya, and G. Sugimoto, "Local path planning and motion control for agv in positioning," in *Proc. of The IEEE International Conference on Robotics and Automation (ICRA)*, Sep 1989, pp. 392–397.

[17] I. Papadimitriou and M. Tomizuka, "Fast lane changing computations using polynomials," in *American Control Conference*, 2003, pp. 48–53.

[18] M. Werling, J. Ziegler, S. Kammel, and S. Thrun, "Optimal trajectory generation for dynamic street scenarios in a frenet frame," in *Proc. of The IEEE International Conference on Robotics and Automation (ICRA)*, May 2010.

[19] J. Laumond, *Robot motion planning and control*. Springer, 1998.

[20] H. Choset, K. M. Lynch, S. Hutchinson, G. Kantorand, W. Burgard, L. E. Kavraki, and S. Thrun, *Principles of robot motion: theory, algorithms, and implementation*. The MIT Press, 2005.

[21] M. Rufli and R. Siegwart, "On the design of deformable input- / state-lattice graphs," in *Proc. of The IEEE International Conference on Robotics and Automation (ICRA)*, May 2010.

[22] J. Ziegler and C. Stiller, "Fast collision checking for intelligent vehicle motion planning," in *Intelligent Vehicles Symposium(IV)*, June 2010.