This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIV.2018.2843178, IEEE Transactions on Intelligent Vehicles

Lane-change Intention Estimation for Car-following Control in Autonomous Driving

Yihuan Zhang[1], Qin Lin[2], Jun Wang[1*], Sicco Verwer[2] and John M. Dolan[3]

***Abstract*—Car-following is the most general behavior in highway driving. It is crucial to recognize the cut-in intention of vehicles from an adjacent lane for safe and cooperative driving. In this paper, a method of behavior estimation is proposed to recognize and predict the lane change intentions based on the contextual traffic information. A model predictive controller is designed to optimize the acceleration sequences by incorporating the lane-change intentions of other vehicles. The public dataset of Next Generation Simulation are labeled and then published as a benchmarking platform for the research community. Experimental results demonstrate that the proposed method can accurately estimate vehicle behavior and therefore outperform the traditional car-following control.**

***Index Terms*—cooperative car-following, driving behavior estimation, lane change prediction, model predictive control.**

I. Introduction

Recently, many research institutes and vehicle manufacturers have focused on the commercialization of autonomous driving systems. Safety and reliability are fundamental for self-driving cars on roads. Most car crashes are caused by human mistakes, and many of these occur during lane changes [1], [2]. Furthermore, fewer than 50% of drivers use turn signals when they change lanes [3]. In order to guarantee the safety of driving, it is important for self-driving cars to estimate the driving behavior of surrounding vehicles and predict their intension of lane change before they cross lane lines.

Fig. 1 illustrates the scenarios in highway driving. The self-driving car is noted as the host vehicle in blue (*Veh-h*), the target vehicle is in red (*Veh-t*), the proceeding vehicle (*Veh-p*) is in front of the host vehicle, *Veh-ft* and *Veh-rt* represent the front and rear vehicles in the target lane. Assume that the red vehicle is following the leading vehicle and intends to merge. In this case, if the host vehicle cannot estimate the merge intention of the red vehicle, a sudden change of accelerations may occur, which leads to an uncomfortable or even dangerous situation. Human drivers predict the behavior of surrounding vehicles (merging into their lane or not) based on their observations and driving experiences. A self-driving car uses a computational model to mimic human beings and estimate the states of its own and surrounding vehicles.

The cut-in intention of the target vehicle should be estimated to ensure a safe and comfort car-following for the host vehicle.

Yihuan Zhang and Jun Wang are with the Department of Control Science and Engineering, Tongji University, Shanghai 201804, P. R. China.

Qin Lin and Sicco Verwer are with the Department of Intelligent Systems, Delft University of Technology, Delft 2628 CD, the Netherlands.

John M. Dolan is with the Robotic Institute, Carnegie Mellon University, Pittsburgh, PA, United States of America.

*Corresponding author junwang@tongji.edu.cn

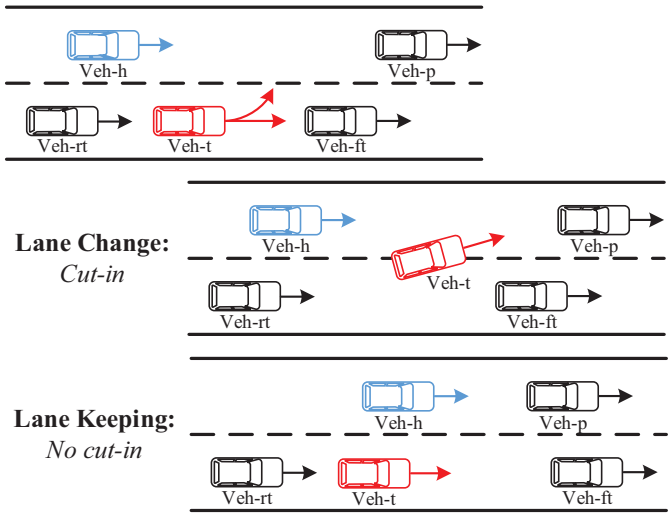


Fig. 1: Multi-lane car-following scenarios.

The contextual information of the four surrounding vehicles is used to model the driving behavior of the target vehicle. In this paper, we need to recognize/classify the observations (vehicle positions, lateral accelerations, etc.) into lane change or lane keeping. It is a standard multivariate-time-series classification based on the observations, i.e., to assign a label to a complete sequence of lane change or lane keeping. This work aims at an even more challenging task of predicting such a label (i.e., intention) in advance for the intervention of control.

Although the Adaptive Cruise Control (ACC) systems have been in market since 1995 [4], their performance in terms of smoothness is frequently interrupted by cut-in vehicles from adjacent lanes. More attention should be paid to the intention of other vehicles for a more reliable ACC. In this paper, an intention-based car-following control method is proposed by integrating the cut-in intention of surrounding vehicles.

The framework of the proposed method is shown in Fig. 2. First, a scenario extraction method is used to obtain two classes of driving sequences: lane change and lane keeping. Then, the continuous Hidden Markov Models (HMMs) integrated with the Gaussian Mixture Models (GMMs) are used to model the behavior of lane change and lane keeping, respectively. A likelihood function is employed to estimate the behavior in an online manner. Finally, a framework of model predictive control is proposed to consider the predicted cut-in intention.

The major contributions of this paper are as follows:

- To the best of our knowledge, this is the first work to fuse traffic contextual information into the driving behavior

2379-8858 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIV.2018.2843178, IEEE Transactions on Intelligent Vehicles

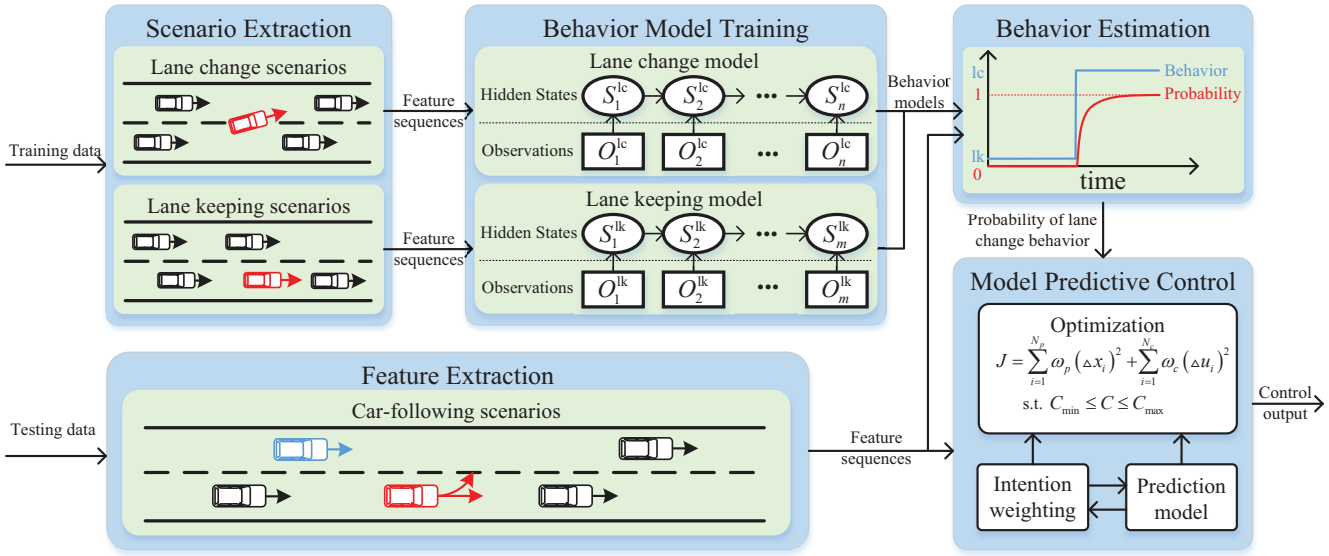


Fig. 2: Framework of proposed method.

estimation of target vehicles by using continuous HMMs.

- A threshold-based method is used to estimate driving behavior of a target vehicle in a streaming fashion, which is able to predict the behavior of lane change before the target vehicle crosses the lane line.
- A novel car-following control method integrating the cut-in intention estimation is proposed and achieves superior performance in terms of comfort and safety.

The remainder of this paper is organized as follows. Related work is introduced in Section II. The proposed method is detailed in Section III. The experiments are carried out in Section IV. Conclusions and future work are presented in Section V.

II. Related Work

The related work is divided into two parts: one is on the estimation and prediction of driving behavior by using various kinds of information, the other is on car-following control including mathematical models, control methods and ACC systems.

A. Driving behavior classification

Many researches focused on the classification and prediction of driving behavior. In [5], the behavior of following and passing a vehicle was modeled and recognized using HMMs and Gaussian mixture model. In [6], a maneuver-based method was proposed to estimate the driving state of a driver and to predict the future trajectory considering the information of its leading vehicle. In car-following scenarios, it is important to monitor the situation in the adjacent lanes to deal with the behavior of lane change.

The behavior estimation or intention recognition of Lane change can be classified into two categories based on its input signals. The first one uses ***internal*** information of a target vehicle such as throttle pedal pressure, brake pedal pressure and steering wheel angles to identify driving behavior. It is mainly used in advanced driver assistance systems. In [7], an accuracy of 93.3% over the 47 recorded lane-change scenarios was achieved based on the data of vehicle accelerations, brakings and steerings. In [8], lateral accelerations, steering wheel angles and steering angles were used to classify the maneuvers of lane keeping and lane change by continuous HMMs and the average recognition rate of lane change was over 90%. In [9], lane change maneuvers were recognized by using the features extracted from vehicle states and driver operation signals. The dataset was recorded from different drivers under varying driving conditions and the recognition rate was 88.2%. In [10], some additional features like eye movements and head dynamics were added to the behavior recognition for improved accuracy. In [11], the signals of heart electrocardiogram, galvanic skin responses and respiration were utilized to train a multi-layer neural-network model. The prediction of lane change was achieved about 2 seconds before the target vehicle actually crossed lane lines.

The other category uses ***external*** information of a target vehicle, for example, vehicle speeds, lateral offsets, distances, etc. It is possible for self-driving cars to estimate the behavior of surrounding vehicles because all parameters are measurable by sensors on board. In [12], lateral positions and relative heading angles were used as features to train a support vector machine (SVM) and a Bayesian filter was used to obtain the probability of driving behaviors. However, the effect of surrounding vehicles on the behavior of the target vehicle should not be ignored. More and more researchers have considered the surrounding traffics when studying driving behaviors. In [13], the lane change intention was estimated based on the driver's head motions, internal signals and the information of the surrounding vehicles. The classifier was able to provide the intention of the driver more accurately.

The dataset of Next Generation SIMulation (NGSIM) have been adopted to explore the characteristics of the vigilant lane-change process. In [14], a fuzzy inference system was used to make a decision of lane change based on distances

2379-8858 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIV.2018.2843178, IEEE Transactions on Intelligent Vehicles

and relative speeds. In [15], a neural-network based learning method was applied to model the behavior of lane change. The SVM-based classification as a classical machine learning method can deal with high-dimensional input features. In [16], [17], SVMs were used to classify different situations of lane-change behavior, and different input features of surrounding vehicles were used to train the SVMs. In addition, a probabilistic classification method based on a Bayesian network was applied in [18], [19]. The time-to-collision between a target vehicle and surrounding vehicles was used as an input feature to obtain the probability of lane-change behavior. An exponential probability model of lane-change was proposed in [20] by using NGSIM data. Various factors were claimed to affect the decision of lane change, including the relative speeds between the target and original lanes and the distances between the target vehicle and the surrounding ones.

B. Car-following control

The first work on car-following can be dated back to the 1950s. In [21], a linear follow-the-leader model was proposed to calculate the desired acceleration by using the relative speeds between the following and the leading vehicles. Another widely-used linear model, knows as the Helly model, was proposed in [22]. Alternatively, a non-linear Gazis-Herman-Rothery model introduced the power operators of ranges and speeds [23]. An intelligent driver model was introduced in [24] to simulate freeway and urban traffics. In our recent work [25], a human-like car-following controller was designed to mimic human driving behavior. These works are essentially feed-forward models that are more suitable for simulating car-following behavior than real-time control.

The ACC system as an upgration of cruise control improves the convenience and safety of driving. Many control methods have been applied to the ACC systems, e.g., proportional-integral (PI) control [26], fuzzy control [27], and model predictive control (MPC) [28], [29]. The MPC method can be used to deal with multiple objective optimizations of driving safety, fuel efficiency and ride comfort. In [30], a scenario MPC method was proposed that enabled predictive and anticipatory driving in multi-lane and multi-vehicle scenarios. By using a stochastic modeling approach, the lane-change probability of surrounding traffic participants was determined and integrated into the optimization. Simulations illustrated the much smoother control of speeds and accelerations than PI control. In [31], a car-following gap model generated from the data of highway naturalistic driving, and the cut-in probability was incorporated into the algorithm of MPC control. Simulated scenarios demonstrated the smoothness of vehicle driving. Although these methods have considered the behavior of vehicles in adjacent lanes, the methods of intention estimation were only tested by simulated data rather than real traffic data. In this work, the models of driving behavior are learned from the real data, and all the testings are conducted in real driving scenarios.

In summary, the smooth and reliable performance of ACC systems tends to be interrupted by cut-in vehicles from adjacent lanes. A model of behavior estimation is crucial for improving the performance of ACC systems. This paper focuses on predicting the cut-in intention at "any time" (i.e., an online fashion) from the external information of surrounding vehicles. The inferred cut-in probability is integrated into the framework of MPC control to efficiently deal with the sudden behavior change of target vehicles.

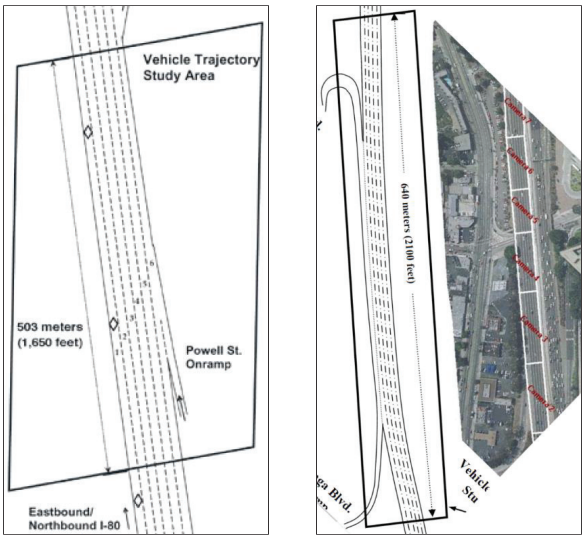
III. Proposed Method

In the NGSIM dataset, separated scenarios for each vehicle are extracted where surrounding vehicles remain the same. Two types of behavior models, i.e., lane keeping and lane change, are learned using GMM-HMMs. In the testing phase, the likelihood of sequences is computed using a forward algorithm and is compared with a threshold for the final recognition. The probability of lane-change is calculated and integrated into the MPC framework to control the car-following behavior of the host vehicle.

A. Scenario definition and extraction

In the folowing, the NGSIM dataset is described in detail and the scenarios used in this paper are defined.

1) Data Description: This paper uses the public datasets of individual vehicle trajectories from NGSIM [32], a program funded by the U.S. Federal Highway Administration. These trajectory data are thus far unique in the history of traffic research and provide a valuable basis for the research of driving behavior on structured roads. All the experiments are performed on the datasets of I-80 and US-101. The road structures of both scenarios are shown in Fig. 3. The labeled scenario data are open-sourced.[1]



(a) I-80 scenario (b) US-101 scenario

Fig. 3: Overview of study area on two NGSIM datasets [32].

The I-80 dataset consists of three 15-minute periods: 4:00 pm to 4:15 pm, 5:00 pm to 5:15 pm, and 5:15 pm to 5:30 pm. These periods represent respectively a buildup of congestion, a

[1]All the labeled scenario data can be found in our online repository: https://bitbucket.org/stzyhian/beta-ngsim.

2379-8858 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIV.2018.2843178, IEEE Transactions on Intelligent Vehicles

transition between uncongested and congested conditions, and full congestion. A total of 45 minutes of data are available in the US-101 dataset, which are segmented into three 15-minute periods: 7:50 am to 8:05 am, 8:05 am to 8:20 am, and 8:20 am to 8:35 am. The vehicle trajectories in both datasets data include the precise location of each vehicle within the study area and the data were sampled at a rate of 10 Hz.

2) Scenario segmentation: The segmented scenarios in Fig 1 have the following properties:

- In each scenario, the surrounding vehicles (*Veh-h*, *Veh-p*, *Veh-ft*, *Veh-rt*) of a target vehicle (*Veh-t*) remain the same.
- We set the relative distance to 150 m and the relative speed to 0 for any missing surrounding vehicles.
- A scenario ends when a target vehicle crosses a lane line (merge), passes *Veh-p*, or yields to *Veh-h*.
- A new scenario restarts immediately once the preceeding scenario is finished to ensure continuity between driving scenarios.
- The segmented scenarios last at least two seconds to ensure complete lane-change or lane-keeping behavior.

TABLE I: Scenario segmentations.

Dataset	Lane change	Lane keeping
I-80-1	212 (avg. dur. 6.12s)	16997 (avg. dur. 6.01s)
I-80-2	159 (avg. dur. 6.13s)	16972 (avg. dur. 6.16s)
I-80-3	167 (avg. dur. 6.30s)	16536 (avg. dur. 6.26s)
US-101-1	242 (avg. dur. 8.07s)	15683 (avg. dur. 7.99s)
US-101-2	156 (avg. dur. 8.56s)	17254 (avg. dur. 8.07s)
US-101-3	154 (avg. dur. 7.44s)	17796 (avg. dur. 7.71s)

The summary of the segmented sequences in both datasets is shown in Table I. The average duration of each scenario segmentation is about 6 to 8 seconds. The highly imbalanced data, i.e., much higher proportion of lane keeping than lane change, pose another significant challenge to behavior recognition. However, the proportion of data is consistent with daily driving. According to References [14], [15], the features listed in Table II are deemed relevant and are extracted.

TABLE II: Features of scenario segmentation.

Symbols	Descriptions
v_{x}	Longitudinal speed of *Veh-t*
d_{o}	Lateral speed of *Veh-t*
d_{o}	Lateral offset from target lane line to *Veh-t*
$\Delta v_{\mathrm{t,p}}$	Longitudinal speed difference between *Veh-t* and *Veh-p*
$\Delta v_{\mathrm{t,h}}$	Longitudinal speed difference between *Veh-t* and *Veh-h*
$\Delta v_{\mathrm{t,ft}}$	Longitudinal speed difference between *Veh-t* and *Veh-ft*
$\Delta v_{\mathrm{t,rt}}$	Longitudinal speed difference between *Veh-t* and *Veh-rt*
$\Delta x_{\mathrm{t,p}}$	Longitudinal distance between *Veh-t* and *Veh-p*
$\Delta x_{\mathrm{t,h}}$	Longitudinal distance between *Veh-t* and *Veh-h*
$\Delta x_{\mathrm{t,ft}}$	Longitudinal distance between *Veh-t* and *Veh-ft*
$\Delta x_{\mathrm{t,rt}}$	Longitudinal distance between *Veh-t* and *Veh-rt*

B. Behavior model

HMMs have been widely used to model driving behavior due to their powerful ability of describing dynamic processes and inferring unobserved (hidden) states [33], [5]. GMMs are used to model the probabilities of the continuous observations such as speeds.

1) GMM: The variables in Table II can be classified into three categories as follows:

$$\begin{aligned}\xi_t = \Big[& \left[v_{\mathrm{x}}(t), v_{\mathrm{y}}(t), d_{\mathrm{o}}(t)\right], \\ & \left[\Delta v_{\mathrm{t,p}}(t), \Delta v_{\mathrm{t,h}}(t), \Delta x_{\mathrm{t,p}}(t), \Delta x_{\mathrm{t,h}}(t)\right], \\ & \left[\Delta v_{\mathrm{t,ft}}(t), \Delta v_{\mathrm{t,rt}}(t), \Delta x_{\mathrm{t,ft}}(t), \Delta x_{\mathrm{t,rt}}(t)\right] \Big]^{\mathrm{T}}\end{aligned}$$

Note that ξ_t is used to model the behaviors in this paper, and the first group $\left[v_{\mathrm{x}}(t), v_{\mathrm{y}}(t), d_{\mathrm{o}}(t)\right]^{\mathrm{T}}$ is used to build the model which only considers the information of target vehicles. In this paper, we assume that the distribution of the observation ξ is a weighted sum of multivariate Gaussian distribution functions:

$$\begin{aligned} p(\xi_t;\theta) &= \sum_{k=1}^{K} \omega_k \mathcal{N}(\xi_t;\mu_k,\Sigma_k) \\ &= \sum_{k=1}^{K} \frac{\omega_k \cdot \exp\left(-\frac{1}{2}(\xi_t-\mu_k)^{\mathrm{T}}\Sigma_k^{-1}(\xi_t-\mu_k)\right)}{\sqrt{(2\pi)^{11}\det(\Sigma_k)}} \end{aligned} \tag{1}$$

where $\theta = \{\theta_k\}_{k=1}^K = \{\omega_k, \mu_k, \Sigma_k\}_{k=1}^K$ are the parameters of the GMMs, $\mathcal{N}(\xi_t;\mu_k,\Sigma_k)$ is the multivariate Gaussian distribution with the mean center $\mu_k \in \mathcal{R}^{11\times 1}$ and covariance matrix $\Sigma_k \in \mathcal{R}^{11\times 11}$, and K is the number of GMM components which can be determined using the Bayesian information criterion (BIC) [34]. As $\omega_k \in (0,1]$ is the weight of the k^{th} Gaussian component, we have $\sum_{k=1}^K \omega_k = 1$.

Given a data sequence $\xi_{1:n}$, the maximum-likelihood estimation method is used to find a θ that maximizes the likelihood of the GMM function:

$$\mathcal{L}(\theta) = \sum_{t=1}^{n} \ln(p(\xi_t;\theta)) \tag{2}$$

The expectation-maximization algorithm is utilized in this paper to search for the optimal parameter

$$\theta^* = \arg\max_{\theta} \mathcal{L}(\theta)$$

The estimation of θ at Step j is denoted by $\hat{\theta}^j$. The iteration from $\hat{\theta}^j$ to $\hat{\theta}^{j+1}$ is achieved by the following *E-step* and *M-step* [35].

- *E-step*: For each iteration, the posterior probability for each component k is calculated by using the previous estimation $\hat{\theta}^j$:

$$P_k^{j+1}(\xi_t) = \frac{\hat{\omega}_k^j \cdot \mathcal{N}(\xi_t;\hat{\mu}_k^j,\hat{\Sigma}_k^j)}{\sum_{l=1}^K \hat{\omega}_l^j \cdot \mathcal{N}(\xi_t;\hat{\mu}_l^j,\hat{\Sigma}_l^j)} \tag{3}$$

- *M-step*: The model parameters are then updated by

$$\begin{aligned} \hat{\omega}_k^{j+1} &= \frac{1}{n}\sum_{t=1}^{n} P_k^{j+1}(\xi_t) \\ \hat{\mu}_k^{j+1} &= \frac{\sum_{t=1}^{n}(\xi_t \cdot P_k^{j+1}(\xi_t))}{\sum_{t=1}^{n} P_k^{j+1}(\xi_t)} \\ \hat{\Sigma}_k^{j+1} &= \frac{\sum_{t=1}^{n}\left(P_k^{j+1}(\xi_t)(\xi_t-\hat{\mu}_k^{j+1})(\xi_t-\hat{\mu}_k^{j+1})^{\mathrm{T}}\right)}{\sum_{t=1}^{n} P_k^{j+1}(\xi_t)} \end{aligned}$$

2379-8858 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIV.2018.2843178, IEEE Transactions on Intelligent Vehicles

In the end of each iteration, the log-likelihood $\mathcal{L}(\hat{\theta}^{j+1})$ is calculated by

$$\mathcal{L}(\hat{\theta}^{j+1}) = \sum_{t=1}^{n} \mathcal{L}(\hat{\theta}^{j}) \tag{4}$$

The iteration will continue until the likelihood difference between two consecutive estimated models is less than a threshold, which is set to 10^{-10} in this paper.

2) HMM: Two separate HMMs are built to represent the behavior of lane change and lane keeping. In this paper, the structure of the HMM is left-to-right, as shown in Fig. 2. The HMM is represented by

$$\lambda = \{\mathcal{S}, \mathcal{Z}, \mathcal{A}, \mathcal{B}, \pi\}$$

where

- $\mathcal{S} = \{s_1, \cdots, s_N\}$ represents a finite set of N hidden states.
- $\mathcal{Z} = \{\xi_t\}$ is the set of all observed states ξ at time t and each ξ consists of the eleven elements included in the GMM.
- $\mathcal{A} = [a_{ij}]$ is the state transition matrix and a_{ij} is defined as the probability of a transition from state s_i to state s_j.
- $\mathcal{B} = \{b_i(\xi)\}$ is the observation model and $b_i(\xi)$ represents the probability of observing ξ while being in state s_i.
- $\pi = \{\pi_i\}$ is the initial state distribution where π_i represents the probability of the state s_i being the initial state.

Readers can referred to [36] for a more detailed formulation and applications of HMM. HMM is a dual stochastic model: one is a Markov model for stochastic state transition, the other is the stochastic observation in each state. Three hidden states are chosen to represent the underlying dynamic processes of the lane-change and lane-keeping behavior. The continuous observation model $\mathcal{B}$ is defined by

$$b_i(\xi) = \sum_{k=1}^{K} \omega_k \mathcal{N}(\xi; \mu_k, \Sigma_k) \tag{5}$$

The Baum-Welch algorithm [37] is used to estimate λ of the two HMMs. It is an approximate iterative optimization technique for maximizing the likelihood of the observations. A random set of initial parameters are chosen and improved by gradient updating.

3) Behavior recognition: In the testing phase, a binary recognition, i.e., lane change or lane keeping, is achieved in a receding horizon manner. Assume that the sequence $\xi_{1:n}$ is a complete period of lane change/keeping, where n is the length of the sequence. The shortest sequence with a size s implies the least information to distinguish two kinds of behavior. A prediction can be achieved if $s < n$. The streaming data $\xi_{1:t}$ where $t \geq s$ is fed as the real-time input to λ_{lk} and λ_{lc} separately for likelihood computation. λ_{lk} and λ_{lc} respectively represent the HMM of lane keeping and lane change. $P(\xi_{1:t}|\lambda_i)$ is obtained by a forward algorithm [36]:

$$P(\xi_{1:t}|\lambda_i) = \sum_{i=1}^{N} \alpha_t(i) \tag{6}$$

where

$$\begin{aligned} \alpha_{t+1}(j) &= \left(\sum_{i=1}^{N} \alpha_t(i) \cdot a_{ij} \right) b_j(\xi_{t+1}) \\ \alpha_1(j) &= \pi_j b_j(\xi_1) \end{aligned} \tag{7}$$

As there is no prior acknowledge of the driving behavior of a specific driver, we assume the prior probabilities of each model are identical. After the calculation of $P(\xi_{1:t}|\lambda_{\text{lk}})$ and $P(\xi_{1:t}|\lambda_{\text{lc}})$, we are able to set a threshold to estimate the current behavior of the target vehicle:

$$\mathcal{R} = \frac{P(\xi_{1:t}|\lambda_{\text{lc}})}{P(\xi_{1:t}|\lambda_{\text{lk}})} \tag{8}$$

where $\mathcal{R}$ indicates whether the classification is more likely to be lane change or keeping.

C. Model predictive control

Once the behavior model is built, a probability of lane change is calculated and integrated into the framework of model predictive control.

1) Intention estimation: The probability of the lane change intention is calculated as follows:

$$P_{\text{c}} = \begin{cases} \tanh\left(\omega_{\text{c}} \cdot \dfrac{\mathcal{R} - \mathcal{R}_{\text{T}}}{\mathcal{R}_{\text{m}} - \mathcal{R}_{\text{T}}}\right), & \mathcal{R} > \mathcal{R}_{\text{T}} \\ 0, & \mathcal{R} \leq \mathcal{R}_{\text{T}} \end{cases} \tag{9}$$

where $\mathcal{R}_{\text{T}}$ is the threshold of the classification, $\mathcal{R}_{\text{m}}$ is the maximum ratio obtained from the training data and ω_{c} is a span parameter indicating the range of the ratio. The likelihood is thus normalized as a probability ranging from 0 to 1. The function "tanh" is selected because the values of random variable $\frac{\mathcal{R}-\mathcal{R}_{\text{T}}}{\mathcal{R}_{\text{m}}-\mathcal{R}_{\text{T}}}$ in the training dataset follows such a distribution with the smallest fitting error.

2) Prediction model: In this paper, the longitudinal motion of the vehicle is expressed by

$$\begin{aligned} x(t+1) &= x(t) + v(t)\Delta t + 0.5a(t)\Delta t^2 \\ v(t+1) &= v(t) + a(t)\Delta t \end{aligned} \tag{10}$$

where x, v, a are respectively the positions, speeds and accelerations of the host vehicle, and Δt is the sampling time. Then the following variables are defined:

- Distances: $\Delta x = x_{\text{f}} - x_{\text{h}}$ where x_{f} is the longitudinal position of the virtual leading vehicle, and $x_{\text{f}} = P_{\text{c}} x_{\text{t}} + (1 - P_{\text{c}}) x_{\text{p}}$. Note that, if the probability P_{c} is 1, then the host vehicle will assume the target vehicle to be the leading vehicle.
- Relative speeds: $\Delta v = v_{\text{f}} - v_{\text{h}}$ where v_{f} is the longitudinal speed of the virtual leading vehicle, and $v_{\text{f}} = P_{\text{c}} v_{\text{t}} + (1 - P_{\text{c}}) v_{\text{p}}$.
- Accelerations: $a_{\text{h}} = j_{\text{h}} \Delta t$ where j_{h} is the jerk of the host vehicle.

Due to the uncertainty of the vehicle motions, we assume that the accelerations of the surrounding vehicles remain the same in the prediction step as in Reference [31]. Such an assumption is reasonable because the prediction window of the MPC is continuously receding to the next time point when the real status of the leading vehicles is updated.

2379-8858 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIV.2018.2843178, IEEE Transactions on Intelligent Vehicles

3) Receding horizon optimization: The cost function of the MPC is designed to meet the following objectives:

- Tracking errors: The objective of car-following control is to follow the speed of the leading vehicle while keeping a safe distance. The distance is defined as a constant time headway policy [30]:

$$d_{\text{des}} = d_0 + \tau_{\text{h1}} v_{\text{h}} + \tau_{\text{h2}} \Delta v$$

where d_0 denotes the desired distance at stand still, τ_{h1}, τ_{h2} are constant time headway parameters.

$$J_{\text{T}} = \omega_{\text{d}} \left(d_{\text{des}} - \Delta x\right)^2 + \omega_{\text{v}} \Delta v^2 \tag{11}$$

- Comfort and smoothness: The host vehicle should realize a comfortable and economic driving style by minimizing its accelerations and jerks.

$$J_{\text{C}} = \omega_{\text{a}} a_h^2 + \omega_{\text{j}} j_h^2 \tag{12}$$

where ω_{d}, ω_{v}, ω_{a} and ω_{u} are the weight values of the cost function.

Considering the nonholonomic constraints of the vehicle and the car-following scenario, the following constraints should also be considered in the MPC design:

- The speed of the host vehicle is bounded by

$$0 \leq v_{\text{h}} \leq v_{\text{max}}$$

- The minimum gap from the leading vehicle is constrained by

$$d_{\text{safe}} \leq \Delta x$$

where $d_{\text{safe}} = \tau_0 v_{\text{h}}$ is the minimum time headway.

- The acceleration constraint of the host vehicle is

$$a_{\text{min}} \leq a_{\text{h}} \leq a_{\text{max}}$$

- The jerk constraint of the host vehicle is

$$j_{\text{min}} \leq j_{\text{h}} \leq j_{\text{max}}$$

The optimization problem can now be write as:

$$\min_{j_{\text{h}}(k), k=1,\cdots,N_{\text{p}}} J = J_{\text{T}} + J_{\text{C}} \tag{13}$$

where N_{p} is the prediction step. The optimization problem is subject to the above constraints. Note that the optimal solution is a vector of control values with the length N_{p}. The MPC method only takes the first value and then moves to the next time point and re-starts the optimization.

IV. Experimental results

The effectiveness of the proposed method is demonstrated by a 5-folder cross validation experiment. In order to balance the data proportion of lane change and keeping, an equal number of data, i.e. 538 sequences, are randomly chosen. First, the BICs are calculated to determine the number of GMM components, where n_t is the length of training data and $\hat{\mathcal{L}}$ is the maximum log-likelihood. When fitting GMMs, it is possible to increase the likelihood by increasing K, which may result in over-fitting. Moreover, the log-likelihood may have a large negative value and the two parts in this equation may not be of the same order of magnitude. A normalization step is set to make a trade-off between number of parameters and log-likelihood:

$$\widehat{\text{BIC}} = \ln(n_t) \cdot \frac{K - K_{\text{min}}}{K_{\text{max}} - K_{\text{min}}} - 2 \cdot \frac{\ln(\hat{\mathcal{L}}) - \ln(\hat{\mathcal{L}})_{\text{min}}}{\ln(\hat{\mathcal{L}})_{\text{max}} - \ln(\hat{\mathcal{L}})_{\text{min}}}$$

In this step, all the sequences in each dataset are used to calculate the BICs with K varying from 1 to 20. There is usually a reasonable range for the elbow-like parameter selection [38]. The final parameter K is chosen based on the minimal normalized BIC. Then $K = 3$ is selected for both lane change and lane keeping in the I-80 dataset; $K = 4$ is selected for lane change and $K = 3$ for lane keeping in the US-101 dataset.

A. Classification evaluation

In order to highlight the effects of surrounding vehicles, the model only considering the information of target vehicles is also studied in the following experiments, which is designated "tgt" for only considering *target* vehicle. The proposed method is designated "srd" for considering *surrounding* vehicles.

A receiver-operating-characteristic curve is a standard analysis tool to score the performance of a binary classifier system with a varying threshold, i.e. $\mathcal{R}_{\text{T}}$ in this paper. The area under the curve (AUC) is equal to the probability that a classifier will rank a randomly chosen positive instance higher than a negative one (assuming positives rank higher than negatives) [39]. In this paper, the AUC means the classification performance of the behavior estimation. The accuracy of behavior estimation is higher when AUC (ranging from 0 to 1) is larger. As shown in Table. III, the AUCs of the "srd" method are higher than the "tgt" method, i.e., the classification results considering surrounding vehicles are more accurate than the results only considering the information of target vehicles.

TABLE III: Comparison of AUCs.

Cases	I	II	III	IV	V
srd-I-80	**0.9603**	**0.9475**	**0.9418**	**0.9575**	**0.9356**
tgt-I-80	0.9282	0.9064	0.9325	0.9046	0.9182
srd-US-101	**0.9173**	**0.9295**	**0.9270**	**0.9358**	**0.9163**
tgt-US-101	0.9065	0.9167	0.9058	0.8980	0.9007

Besides the AUC evaluation, the following quantitative metrics are also introduced for a comprehensive evaluation:

- True Positive Rate (TPR), also named *Recall*, is the fraction of events classified correctly out of all true events, i.e.

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

where TP means true positive and FN means false negative (missed detection).

- False Positive Rate (FPR) is the fraction of events classified wrongly out of all false events, i.e.

$$\text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}}$$

where FP means false positive (false alarm) and TN means true negative.

2379-8858 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIV.2018.2843178, IEEE Transactions on Intelligent Vehicles

TABLE IV: Performance index comparison at FPR = 5%.

Dataset		I-80						US-101					
Cases		I	II	III	IV	V	Average	I	II	III	IV	V	Average
TPR	srd	**0.9158**	**0.8055**	0.8056	**0.8425**	**0.8037**	**0.8346**	**0.8091**	0.7478	0.8108	**0.8091**	**0.7727**	**0.7898**
	tgt	0.7757	0.7407	**0.8241**	0.5278	0.6168	0.6971	0.6546	**0.8378**	**0.8738**	0.5909	0.7636	0.7441
FPR	srd	**0.0654**	**0.0648**	0.0740	0.0740	**0.0654**	**0.0688**	**0.0636**	**0.0811**	**0.0811**	**0.0909**	**0.0727**	**0.0778**
	tgt	0.0841	0.0741	**0.0648**	**0.0648**	**0.0654**	0.0706	0.0909	0.0991	0.0901	0.1000	0.0909	0.0942
ACC	srd	**0.9252**	**0.8703**	0.8657	**0.8842**	**0.8691**	**0.8829**	**0.8727**	0.8333	0.8648	**0.8591**	**0.8501**	**0.8561**
	tgt	0.8458	0.8333	**0.8796**	0.7315	0.7757	0.8132	0.7818	**0.8694**	**0.8919**	0.7455	0.8364	0.8249
PRE	srd	**0.9333**	**0.9255**	0.9157	**0.9191**	**0.9247**	**0.9237**	**0.9271**	**0.9022**	**0.9091**	**0.8989**	**0.9139**	**0.9103**
	tgt	0.9022	0.9091	**0.9271**	0.8906	0.9041	0.9066	0.8781	0.8942	0.9066	0.8553	0.8936	0.8855
F_1	srd	**0.9245**	**0.8614**	0.8571	**0.8792**	**0.8600**	**0.8765**	**0.8641**	0.8177	0.8571	**0.8516**	**0.8374**	**0.8456**
	tgt	0.8342	0.8163	**0.8725**	0.6627	0.7333	0.7838	0.7501	**0.8651**	**0.8899**	0.6989	0.8235	0.8055

- Accuracy (ACC) is the fraction of correctly classified events out of all testing events. It is defined by

$$\text{ACC} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

- Precision (PRE) is the fraction of events classified correctly out of all events predicted to be positive, i.e.

$$\text{PRE} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

- F1 Score is the harmonic mean of the precision and the recall, i.e.

$$F_1 = 2 \times \frac{\text{PRE} \times \text{TPR}}{\text{PRE} + \text{TPR}}$$

Note that the thresholds are determined by choosing FPR $=$ 5% in the training data. The thresholds are then used for the final evaluation in the testing set (see the results reported in Table IV). The evaluation results show that the proposed method considering the information of surrounding vehicles achieves better performance than the method only considering the target vehicle.

B. Lane change prediction

A further challenge is to predict lane change before the target vehicle crosses lane lines. In this paper, the prediction time is defined as

$$\tau_{\text{t}} = t_{\text{e}} - t_{\text{p}}$$

where t_{e} represents the ending time of a scenario and t_{p} is the first instant when a label of lane change is reached. In the testing dataset, t_{e} is the time when the target vehicle crosses the lane lines and t_{p} is the time when the behavior of lane change is estimated. When the ratio $\mathcal{R}$ changes across the threshold, the final driving behavior is estimated as lane change. In addition, if the final output behavior remains lane change until the end of the scenario, the prediction time is obtained as the period between t_{p} and t_{e}.

TABLE V: Lane change prediction time τ_{t} in second.

Cases	I	II	III	IV	V	Average
srd-I-80	5.16	5.21	4.97	3.11	3.49	**4.39**
tgt-I-80	4.12	3.42	2.99	2.58	2.49	3.12
srd-US-101	4.67	4.96	5.38	4.24	4.43	**4.73**
tgt-US-101	2.67	2.81	3.12	2.23	2.41	2.65

Table V compares the average prediction time between "srd" and "tgt", which demonstrates that the proposed method is able to predict the intention of target vehicles earlier. Moreover, a comparison of lane change prediction using SVM [12] is conducted. The results of the proposed method using GMM-HMM are better because the driving behavior is a time series and previous states are related to current and future states. SVM is a classifier that can only input constant dimensions of variables and is thus unable to model the effects of time series effects. The comparison results are shown in Fig. 4 and Fig. 5. The proposed method has an approximately 80% true positive rate of predicting the behavior of lane change 0.5s in advance and retains a 60% true positive rate up to 4s before the lane change occurs. Furthermore, the proposed method also has the lowest false positive rate while the SVM method produces over 20% false positive rate.

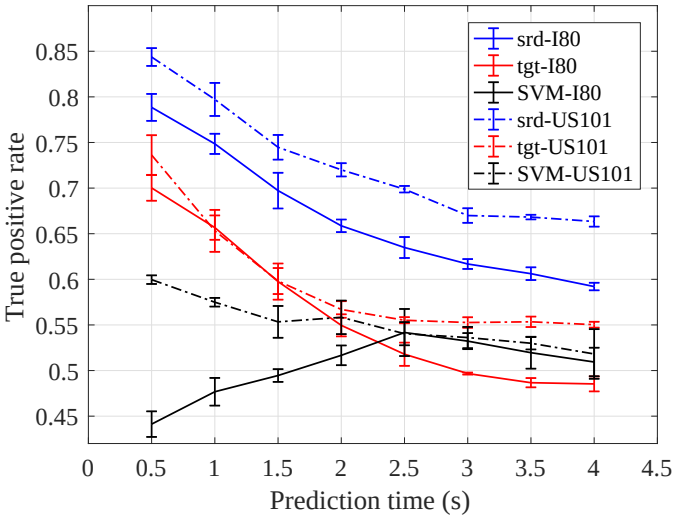


Fig. 4: Prediction time and true positive rate of lane-change behavior in both dataset.

A detailed example in Fig. 6 shows that the driving behavior cannot be correctly estimated by only considering the information of the target vehicle. In Fig. 6(d), lane keeping is 0 and lane change is 1. In the first second of this scenario, the target vehicle is shifting to the right, but it is not lane-change behavior because there is a vehicle in the lane to its right. If only the information of the target vehicle is considered, the algorithm may estimate that the target vehicle is changing lanes even though the target vehicle cannot do so. Therefore, accurate behavior estimation requires considering the traffic situation around the target vehicle. Moreover, due to the lack of

2379-8858 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIV.2018.2843178, IEEE Transactions on Intelligent Vehicles

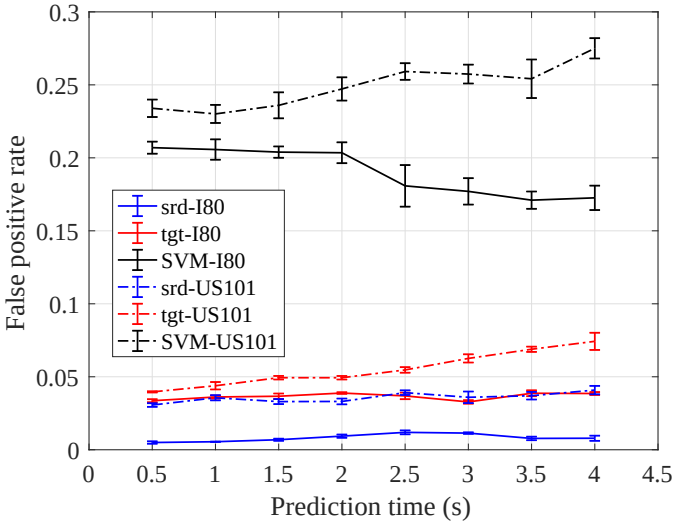


Fig. 5: Prediction time and false positive rate of lane-change behavior in both dataset.

modeling of the time-series sequences, the SVM is not stable and cannot make any estimation or prediction without filtering.

C. Car-following testing results

The scenarios containing the host and target vehicles in the NGSIM dataset are extracted for the car-following control test. The information of the surrounding vehicles are used as the observation of the host vehicle. The parameters of the MPC are listed in Table. VI.

TABLE VI: Parameters in MPC.

Variables	Values	Units
Δt	0.1	s
N_p	20	–
ω_c	10	–
d_0	6	m
v_max	30	m/s
τ_0, τ_h1, τ_h2	0.5, 1, 3	s
ω_d, ω_v, ω_a, ω_j	0.01, 0.02, 0.01, 0.05	–
a_min, a_max	−4, 6	m/s^2
j_min, j_max	−0.3, 0.3	m/s^3

As shown in Table VII, five metrics are selected to evaluate the proposed method and three methods are compared to demonstrate the influences of the cut-in situations. The proposed method is denoted by "srd-MPC", which means the intention of the target vehicle is estimated by considering the information of all the surrounding vehicles. The method "tgt-MPC" represents the MPC controller with the intention estimated only using the information of the target vehicle. The method "Only-MPC" is the pure MPC method without considering the cut-in intentions of target vehicles. The speeds, accelerations and jerks listed in Table VII are the average value in each test. The hazard index is defined as

$$\mathrm{HI} = \exp\left(-(\Delta x/h_1)^{h_2}\right)$$

which represents the degree of a rear end collision [40]. The values of h_1 and h_2 are fitted by the highway naturalistic driving data in [31]. The collision rate (CR) represents the

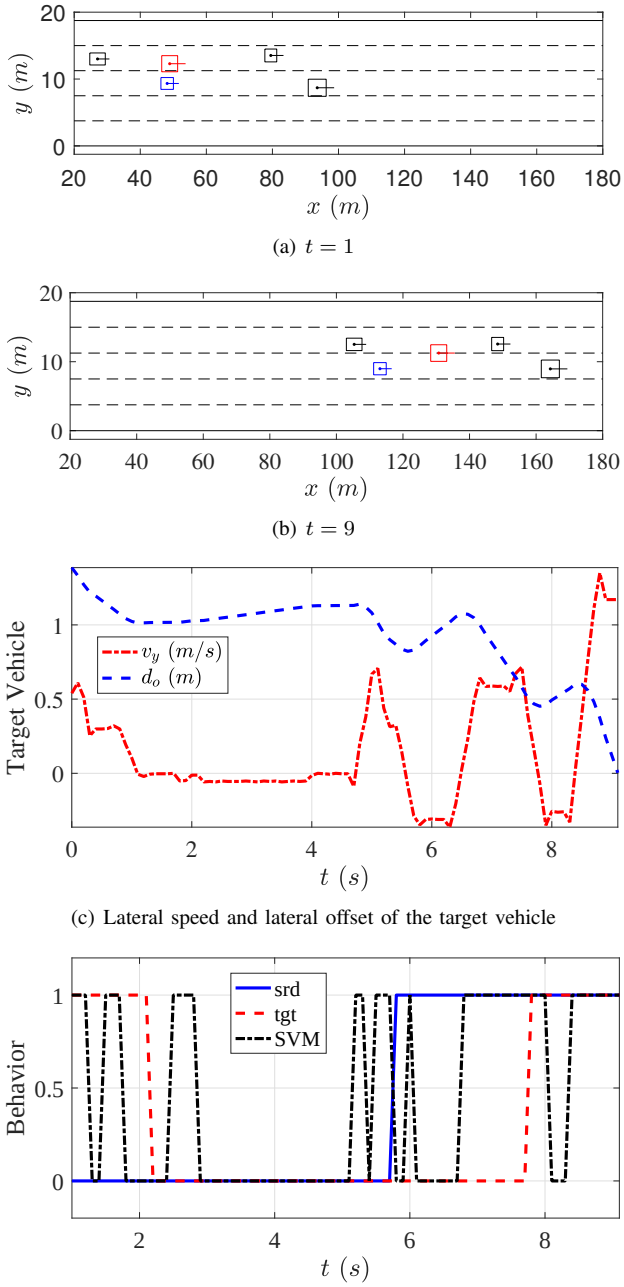


Fig. 6: An example of the proposed behavior estimation method.

collision numbers in the simulation of the host vehicle. The results show that the average speed of the proposed method is close to the traditional MPC. With the intention estimation of the target vehicle, the effect of a sudden change of the leading vehicle is smoothed. Meanwhile, the hazard index and the collision rate of the proposed method is much lower than the other methods. Note that the trajectories of cut-in vehicles are used as real stochastic inputs, though fixed in the dataset, to experimentally demonstrate the collision avoidance control of the proposed method. The on-line interaction between host vehicles and cut-in vehicles are omitted as a fundamental assumptions.

2379-8858 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIV.2018.2843178, IEEE Transactions on Intelligent Vehicles

TABLE VII: Performance index comparison of MPCs.

Dataset		I-80						US-101					
Cases		I	II	III	IV	V	Average	I	II	III	IV	V	Average
v_{h} (m/s)	srd-MPC	6.3667	7.5950	6.2163	6.0926	6.1791	6.4899	10.1505	10.4960	10.2020	10.9406	9.7350	10.3048
	tgt-MPC	6.3292	7.5994	6.1411	5.8433	5.9667	6.3759	10.3989	10.7071	10.5165	11.0393	9.8605	10.5045
	Only-MPC	6.9295	7.5845	6.2827	6.0988	6.2823	**6.6356**	10.6237	10.9072	10.6093	11.2079	9.8176	**10.6331**
a_{h} ($\mathrm{m/s^2}$)	srd-MPC	1.1624	1.0795	1.1589	1.1845	1.2646	**1.1700**	1.1609	1.3325	1.0661	1.7717	1.4109	**1.3484**
	tgt-MPC	1.1974	1.5522	1.3786	1.2096	1.4739	1.3623	1.1632	1.6061	1.4135	1.8874	1.4795	1.5099
	Only-MPC	1.4067	1.5785	1.4746	1.3482	1.4555	1.4527	1.4798	1.6183	1.4058	1.9159	1.7556	1.6351
Δa_{h} ($\mathrm{m/s^3}$)	srd-MPC	0.1253	0.1399	0.1378	0.1409	0.1548	**0.1397**	0.1245	0.1526	0.1145	0.1787	0.1550	**0.1451**
	tgt-MPC	0.1263	0.1836	0.1569	0.1498	0.1783	0.1590	0.1320	0.1710	0.1511	0.1841	0.1619	0.1600
	Only-MPC	0.1625	0.1892	0.1732	0.1606	0.1734	0.1718	0.1698	0.1730	0.1515	0.1827	0.1850	0.1724
HI	srd-MPC	0.0310	0.0214	0.2641	0.3888	0.4185	**0.2248**	0.2664	0.3675	0.1517	1.0301	0.4928	**0.4617**
	tgt-MPC	0.1245	0.4692	0.2650	0.4297	0.8305	0.4238	0.2865	0.8062	0.7836	1.2306	0.7578	0.7729
	Only-MPC	0.6393	0.4705	0.2821	0.6097	0.9959	0.5995	0.6062	0.8269	1.1758	1.3458	1.0394	0.9988
CR	srd-MPC	0/29	0/22	1/25	2/25	2/21	**0.0430**	2/35	2/28	1/40	6/31	2/36	**0.0805**
	tgt-MPC	1/29	3/22	1/25	3/25	3/21	0.0947	2/35	6/28	5/40	8/31	4/36	0.1531
	Only-MPC	4/29	3/22	1/25	4/25	4/21	0.1330	4/35	6/28	8/40	8/31	6/36	0.1907

Two detailed examples from the testing data are illustrated to explain the advantage of the proposed method in Fig. 7 and Fig. 8, where the real data is from human drivers in the dataset. The first example is a cut-in scenario in the I-80 dataset as shown in Fig. 7. In this scenario, the cut-in behavior happens when the target vehicle is slow and wants to give way to a faster following vehicle. As shown in Fig. 7(d), the lane change intention of the target vehicle is detected at 1.8 s by the proposed method, and the target vehicle crosses the lane lines at 7.3 s, where the sudden change of relative distance is shown in Fig. 7(b). Such an intention is detected at 6.6 s using the target vehicle information only. By using the proposed method, the host vehicle is able to take an earlier intervention control of slowing down before the cut-in, therefore obtains smooth accelerations and avoids a hard brake.

Another example from the US-101 dataset is shown in Fig. 8. The target vehicle in this scenarios is trying to merge into the lane of the host vehicle to speed up. The proposed method estimates the cut-in behavior at 1.1 s, while the target vehicle crosses the lane lines at 8.2 s. Similarly to the last scenarios, an earlier and smoother control can be seen in the Jerk subplot. Without the intention estimation, the host vehicle controlled by the pure MPC fails to avoid the collision due to the sudden cut-in.

V. Conclusions

This paper develops a car-following control method with the estimation of the lane-change behavior of other traffic participants. Multivariate time series data from the target vehicle and its surrounding vehicles are used to build two continuous HMMs representing the behavior of lane change and land keeping. A threshold-based classification method is used to estimate the target vehicle's behavior. In the meantime, a cut-in probability is calculated based on the behavior estimation and the MPC method is then applied to optimize the car-following behavior of the host vehicle. The behavior model of the target vehicle is able to achieve over 85% of the true positive rate and the lane change behavior is predicted about 4 seconds before the target vehicle crosses the lane lines. The proposed intention-based MPC achieves superior performance of safety and ride comfort.

In future, we will investigate the strategies based on intention prediction in more complicated scenarios like at intersections. The interpretation of the complicated model is also a research line. The insightful model like timed automaton would act as a promising alternative solution.

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China under Grant No. 61473209, Technologiestichting STW VENI Project 13136 (MANTA), and NWO Project 62001628 (LEMMA).

References

[1] H. Lum and J. A. Reagan, "Interactive highway safety design model: accident predictive module," *Public Roads*, vol. 58, no. 3, 1995.

[2] M. Peden, R. Scurfield, D. Sleet, D. Mohan, A. A. Hyder, E. Jarawan, C. D. Mathers, *et al.*, "World report on road traffic injury prevention," 2004.

[3] R. Dang, F. Zhang, J. Wang, S. Yi, and K. Li, "Analysis of Chinese driver's lane change characteristic based on real vehicle tests in highway," in *The 16th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, 2013, pp. 1917–1922.

[4] R. Rajamani, "Adaptive cruise control," *Encyclopedia of Systems and Control*, pp. 13–19, 2015.

[5] D. Meyer-Delius, C. Plagemann, and W. Burgard, "Probabilistic situation recognition for vehicular traffic scenarios," in *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2009, pp. 459–464.

[6] M. Schreier, V. Willert, and J. Adamy, "An integrated approach to maneuver-based trajectory prediction and criticality assessment in arbitrary road environments," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 10, pp. 2751–2766, 2016.

[7] A. Pentland and A. Liu, "Modeling and prediction of human behavior," *Neural Computation*, vol. 11, no. 1, pp. 229–242, 1999.

[8] H. Hou, L. Jin, Q. Niu, Y. Sun, and M. Lu, "Driver intention recognition method using continuous hidden markov model," *International Journal of Computational Intelligence Systems*, vol. 4, no. 3, pp. 386–393, 2011.

[9] G. Li, S. E. Li, Y. Liao, W. Wang, B. Cheng, and F. Chen, "Lane change maneuver recognition via vehicle state and driver operation signalsresults from naturalistic driving data," in *Intelligent Vehicles Symposium (IV)*. IEEE, 2015, pp. 865–870.

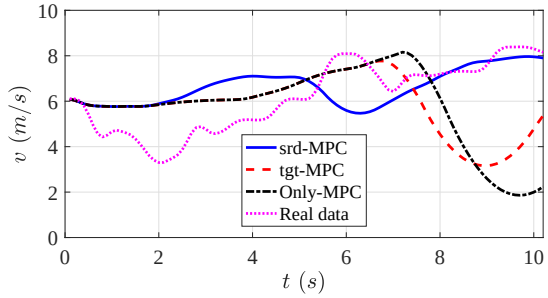
[10] A. Doshi and M. M. Trivedi, "On the roles of eye gaze and head dynamics in predicting driver's intent to change lanes," *IEEE Transactions on Intelligent Transportation Systems*, vol. 10, no. 3, pp. 453–462, 2009.

[11] X. Wang, Y. L. Murphey, and D. S. Kochhar, "Mts-deepnet for lane change prediction," in *The International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2016, pp. 4571–4578.

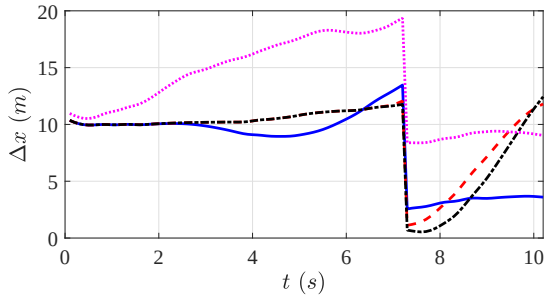
[12] P. Kumar, M. Perrollaz, S. Lefevre, and C. Laugier, "Learning-based approach for online lane change intention prediction," in *Intelligent Vehicles Symposium (IV)*. IEEE, 2013, pp. 797–802.

2379-8858 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

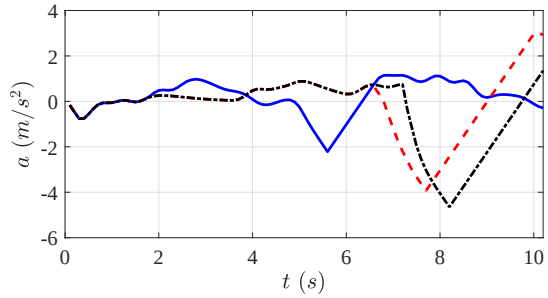
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIV.2018.2843178, IEEE Transactions on Intelligent Vehicles



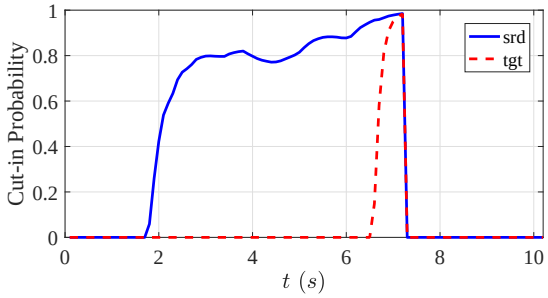
(a) Speed



(b) Relative distance

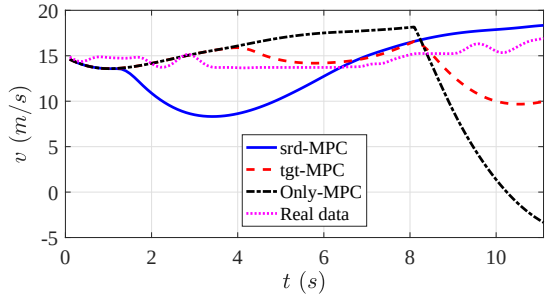


(c) Acceleration

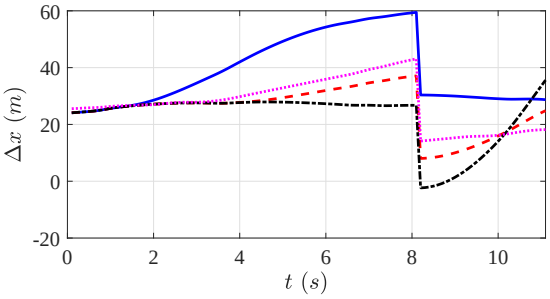


(d) Cut-in Probability

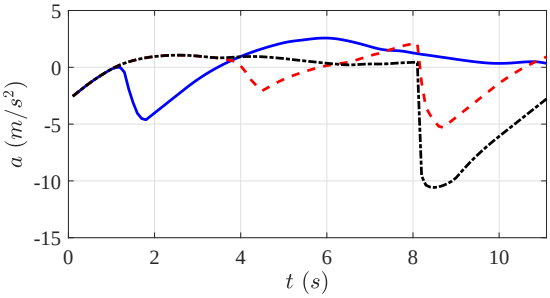
Fig. 7: An example of the car-following simulation in the I-80 dataset.



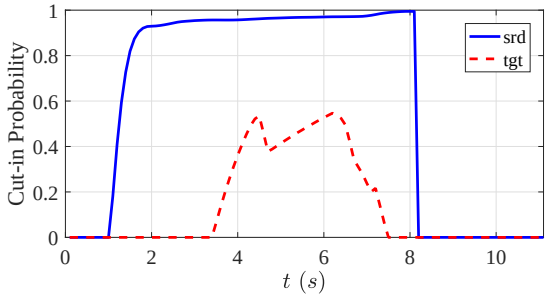
(a) Speed



(b) Relative distance



(c) Acceleration



(d) Cut-in Probability

Fig. 8: An example of the car-following simulation in the US-101 dataset.

[13] B. Morris, A. Doshi, and M. Trivedi, "Lane change intent prediction for driver assistance: On-road design and evaluation," in *Intelligent Vehicles Symposium (IV)*. IEEE, 2011, pp. 895–901.

[14] E. Balal, R. L. Cheu, and T. Sarkodie-Gyan, "A binary decision model for discretionary lane changing move based on fuzzy inference system," *Transportation Research Part C: Emerging Technologies*, vol. 67, pp. 47–61, 2016.

[15] H. Bi, T. Mao, Z. Wang, and Z. Deng, "A data-driven model for lane-changing in traffic simulation," in *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. Eurographics Association, 2016, pp. 149–158.

[16] J. Nie, J. Zhang, X. Wan, W. Ding, and B. Ran, "Modeling of decision-making behavior for discretionary lane-changing execution," in *The 19th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2016, pp. 707–712.

[17] H. Woo, Y. Ji, H. Kono, Y. Tamura, Y. Kuroda, T. Sugano, Y. Yamamoto, A. Yamashita, and H. Asama, "Dynamic potential-model-based feature for lane change prediction," in *The Proceedings of the 2016 IEEE*

2379-8858 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIV.2018.2843178, IEEE Transactions on Intelligent Vehicles

International Conference on Systems, Man, and Cybernetics, 2016.

[18] F. Yan, M. Eilers, A. Lüdtke, and M. Baumann, "Developing a model of driver's uncertainty in lane change situations for trustworthy lane change decision aid systems," in *Intelligent Vehicles Symposium (IV)*. IEEE, 2016, pp. 406–411.

[19] T. Rehder, W. Muenst, L. Louis, and D. Schramm, "Learning lane change intentions through lane contentedness estimation from demonstrated driving," in *The 19th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2016, pp. 893–898.

[20] J. Lee, M. Park, and H. Yeo, "A probability model for discretionary lane changes in highways," *KSCE Journal of Civil Engineering*, vol. 20, no. 7, pp. 2938–2946, 2016.

[21] L. A. Pipes, "An operational analysis of traffic dynamics," *Journal of Applied Physics*, vol. 24, no. 3, pp. 274–281, 1953.

[22] W. Helly, "Simulation of bottlenecks in single-lane traffic flow," in *Proceedings of the Symposium on Theory of Traffic Flow*. New York: Elsevier, 1959, pp. 207–238.

[23] D. C. Gazis, R. Herman, and R. W. Rothery, "Nonlinear follow-the-leader models of traffic flow," *Operations Research*, vol. 9, no. 4, pp. 545–567, 1961.

[24] M. Treiber, A. Hennecke, and D. Helbing, "Congested traffic states in empirical observations and microscopic simulations," *Physical Review E*, vol. 62, no. 2, p. 1805, 2000.

[25] Y. Zhang, Q. Lin, J. Wang, S. Verwer, and M. J. Dolan, "A data-driven behavior generation algorithm in car-following scenarios," in *The 25th Symposium of the International Association for Vehicle System Dynamics (IAVSD)*, 2017.

[26] R. Rajamani, *Vehicle dynamics and control*. Springer Science & Business Media, 2011.

[27] S. P. Sathiyan, S. S. Kumar, and A. I. Selvakumar, "Optimised fuzzy controller for improved comfort level during transitions in cruise and adaptive cruise control vehicles," in *The 2015 International Conference on Signal Processing And Communication Engineering Systems (SPACES)*. IEEE, 2015, pp. 86–91.

[28] R. Schmied, H. Waschl, and L. del Re, "Extension and experimental validation of fuel efficient predictive adaptive cruise control," in *American Control Conference (ACC)*. IEEE, 2015, pp. 4753–4758.

[29] M. A. S. Kamal, S. Taguchi, and T. Yoshimura, "Efficient vehicle driving on multi-lane roads using model predictive control under a connected vehicle environment," in *Intelligent Vehicles Symposium (IV)*. IEEE, 2015, pp. 736–741.

[30] R. Schmied, D. Moser, H. Waschl, and L. del Re, "Scenario model predictive control for robust adaptive cruise control in multi-vehicle traffic situations," in *Intelligent Vehicles Symposium (IV)*. IEEE, 2016, pp. 802–807.

[31] K. Liu, J. Gong, A. Kurt, H. Chen, and U. Ozguner, "A model predictive-based approach for longitudinal control in autonomous driving with lateral interruptions," in *Intelligent Vehicles Symposium (IV)*. IEEE, 2017, pp. 359–364.

[32] NGSIM, "U.S. Department of Transportation, NGSIM - Next generation simulation," http://www.ngsim.fhwa.dot.gov, 2007.

[33] K. Tang, S. Zhu, Y. Xu, and F. Wang, "Modeling drivers' dynamic decision-making behavior during the phase transition period: An analytical approach based on hidden markov model theory," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 1, pp. 206–214, 2016.

[34] D. F. Findley, "Counterexamples to parsimony and bic," *Annals of the Institute of Statistical Mathematics*, vol. 43, no. 3, pp. 505–514, 1991.

[35] J. A. Bilmes *et al.*, "A gentle tutorial of the em algorithm and its application to parameter estimation for gaussian mixture and hidden markov models," *International Computer Science Institute*, vol. 4, no. 510, p. 126, 1998.

[36] L. R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[37] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the Royal Statistical Society. Series B (methodological)*, pp. 1–38, 1977.

[38] S. Salvador and P. Chan, "Learning states and rules for detecting anomalies in time series," *Applied Intelligence*, vol. 23, no. 3, pp. 241–255, 2005.

[39] T. Fawcett, "An introduction to roc analysis," *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861–874, 2006.

[40] Y. Dou, D. Ni, Z. Wang, J. Wang, and F. Yan, "Strategic car-following gap model considering the effect of cut-ins from adjacent lanes," *IET Intelligent Transport Systems*, vol. 10, no. 10, pp. 658–665, 2016.

Yihuan Zhang has been a doctoral student in the Department of Control Science & Engineering at Tongji University since September 2013. He is now a visiting scholar at Carnegie Mellon University. His research interests include environment perception, behavior estimation and decision making for self-driving cars.

Qin Lin has been a PhD Student in the Department of Intelligent Systems at Delft University of Technology since 2015. His research interests include machine learning, time series data mining, and syntactic pattern recognition.

Jun Wang (S'98–M'03–SM'12) received his PhD in control engineering from the University of Leeds (UK) in 2003. He has been the Professor in Control Engineering in the Department of Control Science & Engineering at Tongji University since September 2010. His research interests are smart sensing and intelligent control in the areas of autonomous vehicles and renewable energy. Prof. Wang is in the Editorial Board of *The International Journal of Driving Science*.

Sicco Verwer received the PhD degree from Delft University of Technology. He is an assistant professor of computer science at Delft University of Technology. His research interests include the theory and practice of machine learning and state machine learning in particular. His interests within this focus area are diverse. He has published papers on learning state machines, discrimination-aware data mining, software testing, fraud detection, mechanism design, and combinatorial solvers in machine learning. In particular, he is interested in machine learning algorithms resulting in models that are useful for subsequent tasks such as visualization, testing, verification, data integration, control, and optimization.

John M. Dolan received the BSE degree from Princeton University, Princeton, NJ, USA, and the ME and PhD degrees from Carnegie Mellon University, Pittsburgh, PA, USA, all in mechanical engineering, in 1980, 1987, and 1991, respectively. He is a Principal Systems Scientist with the Robotics Institute, CMU. He was the behaviors leader for Carnegie Mellon's Tartan Racing team in the 2007 DARPA Urban Challenge. His research interests include autonomous driving, multi-robot cooperation, human-robot interaction, robot reliability, and sensor-based control.

2379-8858 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.