The Journal of Intelligent and Connected Vehicles publishes research covering all aspects of intelligent and connected vehicles technology, reflecting the most interesting and strategically important research and development activities from around the world. Readers can stay at the forefront of industry development by accessing a variety of peer reviewed research articles, technology reviews and specially commissioned case studies. Research papers accepted for publication in JICV are double blind refereed to ensure academic rigour and integrity.

EDITOR-IN-CHIEF
Jianqiang Wang
State Key Laboratory of Automotive Safety and Energy, Tsinghua University, China
E-mail: wjqlws@tsinghua.edu.cn
Homepage: http://www.emeraldgrouppublishing.com/services/publishing/jicv/index.htm

ASSOCIATE EDITORS
Shengbo Li
Associate professor, Department of Automotive Engineering, Tsinghua University, China

Daxin Tian
Associate professor, School of Transportation Science and Engineering, Beihang University, China

Mónica Menéndez, PhD
Institute for Transport Planning and Systems (IVT), ETH Zürich, Switzerland

Pasquale Pace, PhD
Department of Computer Engineering, Modelling, Electronics and Systems (DIMES), University of Calabria, Italy

Dongsuk Kum, Ph.D
The Cho Chun Shik Graduate School of Green Transportation (GSGT), Republic of Korea

Pongsathorn Raksincharoensak, Dr. Eng.
Department of Mechanical Systems Engineering, Tokyo University of Agriculture and Technology, Japan

Xiaobo Qu, PhD
School of Civil and Environmental Engineering, University of Technology Sydney, Australia

Lingxi Li
Associate professor, Indiana University, Department of Electrical and Computer Engineering, Purdue University Indianapolis (IUPUI), USA

Keqiang Li
Department of Automotive Engineering, Tsinghua University, China

Diange Yang
Department of Automotive Engineering, Tsinghua University, China

Hong Chen
State Key Laboratory of Automotive Simulation and Control, Jilin University, China

Xiaoyun Lu
PATH, Institute of Transportation Studies, University of California Berkeley, USA

Wenlong Jin
Department of Civil and Environmental Engineering, University of California Irvine, USA

Dongpu Cao
Department of Automotive Engineering, Cranfield University, U.K.

Ludovic Leclercq
Laboratory of Transport Traffic Engineering (UCIT), French Institute of Transport and Development Sciences and Technologies (IFSTTAR), University of Lyon, France

Professor Guodong Yin
School of Mechanical Engineering, Southeast University, China

Emerald Publishing Limited
Howard House, Wagon Lane, Bingley BD16 1WA, United Kingdom
Tel +44 (0) 1274 777700; Fax +44 (0) 1274 785201
E-mail emerald@emeraldinsight.com

For more information about Emerald’s regional offices please go to http://www.emeraldgrouppublishing.com/offices

Customer helpdesk:
Tel +44 (0) 1274 785278; Fax +44 (0) 1274 785201
E-mail support@emeraldinsight.com

The Publisher and Editors cannot be held responsible for errors or any consequences arising from the use of information contained in this journal; the views and opinions expressed do not necessarily reflect those of the Publisher and Editors, neither does the publication of advertisements constitute any endorsement by the Publisher and Editors of the products advertised.

Emerald is a trading name of Emerald Publishing Limited
Printed by CPI Group (UK) Ltd, Croydon, CR0 4YY

ISSN 2399-9802
© 2018 State Key Laboratory of Automotive Safety and Energy, Tsinghua University

Guidelines for authors can be found at: http://www.emeraldgrouppublishing.com/services/publishing/jicv/authors.htm

The Journal of Intelligent and Connected Vehicles
Indexed and abstracted by:
The British Library
A novel intelligent vehicle risk assessment method combined with multi-sensor fusion in dense traffic environment

Xunjia Zheng and Bin Huang
Tsinghua University, Beijing, China

Daiheng Ni
Department of Civil and Environmental Engineering, Amherst College of Engineering, University of Massachusetts, Amherst, Massachusetts, USA, and

Qing Xu
Tsinghua University, Beijing, China

Abstract

Purpose – The purpose of this paper is to accurately capture the risks which are caused by each road user in time.

Design/methodology/approach – The authors proposed a novel risk assessment approach based on the multi-sensor fusion algorithm in the real traffic environment. Firstly, they proposed a novel detection-level fusion approach for multi-object perception in dense traffic environment based on evidence theory. This approach integrated four states of track life into a generic fusion framework to improve the performance of multi-object perception. The information of object type, position and velocity was accurately obtained. Then, they conducted several experiments in real dense traffic environment on highways and urban roads, which enabled them to propose a novel road traffic risk modeling approach based on the dynamic analysis of vehicles in a variety of driving scenarios. By analyzing the generation process of traffic risks between vehicles and the road environment, the equivalent forces of vehicle–vehicle and vehicle–road were presented and theoretically calculated. The prediction steering angle and trajectory were considered in the determination of traffic risk influence area.

Findings – The results of multi-object perception in the experiments showed that the proposed fusion approach achieved low false and missing tracking, and the road traffic risk was described as a field of equivalent force. The results extend the understanding of the traffic risk, which supported that the traffic risk from the front and back of the vehicle can be perceived in advance.

Originality/value – This approach integrated four states of track life into a generic fusion framework to improve the performance of multi-object perception. The information of object type, position and velocity was used to reduce erroneous data association between tracks and detections. Then, the authors conducted several experiments in real dense traffic environment on highways and urban roads, which enabled them to propose a novel road traffic risk modeling approach based on the dynamic analysis of vehicles in a variety of driving scenarios. By analyzing the generation process of traffic risks between vehicles and the road environment, the equivalent forces of vehicle–vehicle and vehicle–road were presented and theoretically calculated.

Keywords Automated vehicles, Advanced vehicle safety systems, Autonomous driving, Connected vehicles, Environment perception, Sensor information fusion

Paper type Research paper

1. Introduction

In 2015, nearly 190,000 crashes were reported in China, causing more than 58,000 fatalities and 200,000 injuries (TMBPSM, 2016). Traffic accidents are a major public-safety problem in developing countries such as China, which also cause enormous economic losses and can even destroy families.

Fortunately, intelligent driving technologies such as advanced driver assistance systems (ADAS) and autonomous vehicles have been developed in an effort to avoid vehicle

The current issue and full text archive of this journal is available on Emerald Insight at: www.emeraldinsight.com/2399-9802.htm

© Xunjia Zheng, Bin Huang, Daiheng Ni and Qing Xu. Published in Journal of Intelligent and Connected Vehicles. Published by Emerald Publishing Limited. This article is published under the Creative Commons Attribution (CC BY 4.0) licence. Anyone may reproduce, distribute, translate and create derivative works of this article (for both commercial and noncommercial purposes), subject to full attribution to the original publication and authors. The full terms of this licence may be seen at http://creativecommons.org/licenses/by/4.0/legalcode

This work was supported by the National Science Fund for Distinguished Young Scholars (51625503), the National Natural Science Foundation of China, the General Project (51475294) and the Major Project (61790561). We appreciate Yinghan Wang, Biao Xu and Yougang Bian for their valuable comments and helpful work.

Received 21 February 2018
Revised 4 May 2018
Accepted 22 June 2018
crashes and minimize the impact of accidents (Ji et al., 2017). With the development of ADAS and autonomous driving (AD), the perceptual ability of vehicles, such as detection and tracking of objects, has been greatly improved. However, there is still a long way to go, especially regarding applications of these technologies in real dense traffic environment. When achieving dynamic object perception in complicated traffic conditions, we are facing challenges such as a large number of false positives or negatives, the limited sensing range and the diversity of dynamic obstacles.

Multi-sensor fusion is a basic method (Alessandretti et al., 2007; Li et al., 2014) proposed to deal with the above challenges. Generally, sensor fusion for tracking can be set up as track-level or detection-level fusion (Duraisamy et al., 2013). Tracks are delivered from the local tracker of each sensor to the fusion center in the track-level fusion to increase computational and communicational efficiency, while detections are sent directly to the fusion center in detection-level fusion, which uses more information and can be applied in challenging scenarios by advanced computation units. To understand the detection-level fusion of multi-object tracking better, we divide it into four steps. First, we need to predict the state of the object based on the historical tracking results by normally using the motion model. Second, dealing with data association between tracks and detections is indispensable. In this step, Bell and Stone (2014) exploited the function of data association based on joint probabilistic data association (JPDA) to solve the problem of best matching. Also, Dempster–Shafer theory (DST) is used in Fan et al.’s (2016) study to associate the current data with the predicted data with the format \([x, y, z, \text{type}]\) without extended information, such as speed. Then, the object states are updated based on Kalman Filter (KF) (Ligorio and Sabatini, 2015), extended KF (EKF) (Kim et al., 2015) or particle filter (Xiao et al., 2016), using the associated sensor observations. Last but not the least, to further generate multi-object tracking trajectories, the tracklets which belong to one specific target need to be considered, and then different tracklets from different targets need to be handled (Luo et al., 2014). To get a complete target trajectory, the concept of lifecycle management of dynamic objects is used broadly. Huang et al. (2008) designed an entry–exit map to predict the start and end of tracking tracklets. Furthermore, Luber et al. (2011) used an approach including three stages, i.e. generation, validation and dead, to manage the life cycle of different objects, by using multi-hypothesis tracking framework. However, it is difficult to achieve long-term stable tracking, considering the enormous variability of dynamic multi-object perception. More states, such as four subspaces of states (i.e. active, tracked, inactive and lost), are used in the model with Markov decision processes to manage the lifetime of a track (Xiang et al., 2015), but an inactive target remains so forever without considering false missing cases. Besides, rather than just monitoring frontal objects (Chavez-Garcia et al., 2014), we consider the perception of objects in 360 degrees in this paper, which is more difficult to keep their tracks constantly.

Another key technology in ADAS products and AD is traffic risk identification. Recently, the collision distance and collision time logic algorithm are mainly used in longitudinal risk identification. The typical collision distance-based algorithms include Mazda model, Honda model, JHU model, Jaguar model, fixed car-following distance model and critical safety distance model (Van Winsum, 1999; Lee and Peng, 2005; Pei et al., 2012). The typical collision time-based algorithms usually take the time to collision (TTC) into account such as TTC, inverse TTC (TTCi), THW (time headway) and so on (Ward et al., 2015; Balas and Balas, 2006; Sharifi et al., 2016). For lateral safety, car’s current position, time to lane cross and variable rumble strip are mainly used in driver assistance products (Risack et al., 2000; Plutti and Ulsoy, 2003; Mammar et al., 2004). However, traffic risk cannot be described as a continuous variable by using these methods which are artificially divided into longitudinal and lateral directions.

Some researchers studied the traffic risk from a statistical perspective, such as road accident rate analysis (Kuliczewska, 2016) and process analysis of traffic conflicts and crashes (Davis et al., 2011), but these methods fail to assess traffic risk dynamically. Another way to capture traffic risk is surrogate safety measures modeling (Pirdavani et al., 2010, 2011; Wu and Jovanis, 2012a, 2012b). However, many of these measures have not been used in models because of the structure of the model or difficulties in measuring them in existing models (Young et al., 2014).

Our previous research (Wang et al., 2014, 2015, 2016) presented a driving safety field theory based on the artificial potential field concept. This method is able to represent risks of driving caused by drivers, vehicles, roads and other traffic factors. Nevertheless, the driving safety field model contains a number of undetermined constants, calibration of which is difficult by using existing technologies.

In this paper, we firstly use DST to handle uncertainty information in multi-sensor fusion for dynamic object perception on road by considering target position, type and velocity, especially in aspects of track management and data association. Then, we present a novel road traffic risk modeling approach according to the “artificial potential field” concept. Traffic risk caused by a vehicle was quantitatively described according to kinetic energy of the vehicle after considering the distance between the vehicle and other road users or environment. Its influence range is expressed by considering the vehicle’s dynamical state and traffic environment conditions. Hence, traffic risk can be described by a relatively accurate method. Finally, an intelligent vehicle platform is built to test perception accuracy in dense traffic environment, and experiment illustrates that the intelligent vehicle can drive autonomously based on our road traffic risk model.

The rest of this paper is arranged as follows. The architecture of dynamic object perception approach and the details of fusion approach, including the basic mathematical concept, track management and data association, are introduced in the Section 2. Section 3 presents a novel traffic risk recognition approach, including the traffic risk range model and the concept of traffic safety field. In section 4, a vehicle platform equipped with multi-sensors is established and the real traffic scenarios and simulation experiments are described to verify the effectiveness of this approach and to analyze its results. Section 5 presents the discussions of this study. Conclusion is drawn in Section 6.
2. Multi-object perception in dense traffic environment

2.1 Approach architecture

We propose a generic framework of detection-level fusion approach, as depicted in Figure 1. The main part of this framework is to track management, including data associated with detections and tracks, track prediction and track update. The integrated track management is expected to reliably achieve multi-target tracking (MTT) in dense traffic scenarios based on detection-level fusion. Its input is the detection list provided by the data buffer, which converts asynchronous and heterogeneous detection data from different sensors into a uniform format. The output is the selected track list from the stage of track management. On the basis of fusing multi-sensor information and managing tracks, we can provide stable and accurate tracking of multiple dynamic objects to the decision layer of ADAS or AD to make a further decision on trajectory planning.

How to better manage tracks in case of a mass of Radar clutters, vision target occlusion and over/under-segmentation of LiDAR detection in heavy traffic environment is full of challenge. To solve this problem, a novel DST-based track management framework is designed. It represents uncertain track state as four types, i.e. unsure, mature and important, mature but unimportant and erased, updated by using DS combination principle. Firstly, the data association between detections and tracks that are predicted in one step by track prediction module is performed by using type, position and velocity of detected objects to reduce erroneous matching and inaccurate state estimation. Then, for unassigned tracks, the basic belief of its erased state increases and can be a criterion whether one should be erased. Processes such as track maturing, track grouping and track selection in the pre-tracking module search potential new tracks from the unassigned detections. Meanwhile, the states of the assigned tracks, including the DS basic belief of target type and track state, and kinematical state are updated by the currently associated tracks. After the next frame of detections is received, these tracks will be predicted using constant acceleration (CA) model and input into the association part for matching new detections.

2.2 Dempster-Shafer theory-based detection-level fusion

As mentioned above, DST makes contributions to both overall track management and its data association part of detection-level fusion for multi-object perception. In this section, the basic concepts of DST are reviewed. Then, it is adapted to our track management and data association. The former models the lifetime of tracks which can naturally handle the transition of four states, and the latter takes the information of position, velocity and even object type into consideration to reduce erroneous data association between tracks and detections.

2.2.1 Basic concepts of Dempster-Shafer theory

Derived from Shafer (1976), DST is practically more flexible than the Bayesian theory that requires probabilities for each concern, while dealing with the uncertainty in multi-target detection (Ayoun and Smets, 2011). For one thing, it is good at combining evidences from different information sources and historical data by its combination principle, which is similar to the recursive Bayesian updating. For another, it is equipped with a rational process for the management of conflict and unknown information.

Let \( \Theta = \{ \theta_1, \theta_2, \ldots, \theta_n \} \) be a finite set of mutually exclusive elements, \( i = 1, 2, \ldots, n \), namely, the frame of discernment. A mass belief function, also called the basic belief assignment (BBA), is essentially a mapping \( m \) from \( 2^{\Theta} \) to \([0, 1]\), to assign the evidence of all propositions over the power set \( 2^{\Theta} \), which is described as follows:

\[
m(\emptyset) = 0 \quad \text{and} \quad \sum_{A \subseteq \Theta} m(A) = 1.
\]  

where \( m(A) \) denotes the BBA assigned to the proposition of \( A \), and the BBA assigned to the zero set is zero. To achieve a combination of evidence from different information sources, several combination principles were proposed. One of the most popular principles is Dempster’s rule mentioned in the transferable belief model (Smets and Kruse, 1997). It uses the conjunctive rule of combination, represented as:

\[
m^{\Theta}(A) = \sum_{A_1 \cap \cdots \cap A_n = A} m_1^\Theta(A_1) \times \cdots \times m_n^\Theta(A_n).\]

The calculated BBAs are supposed to be normalized to remove the influence of conflict information:

\[
m_{12 \ldots n}^\Theta(A) = \frac{m_1^\Theta(A)}{1 - K_{12 \ldots n}}, \quad \forall A \subseteq \Theta, A \neq \emptyset.
\]

where \( K_{12 \ldots n} \) is the degree of conflict of all \( A_i, i = 1, 2, \ldots, n \) denoted as:

\[
K_{12 \ldots n} = \sum_{A_1 \cap \cdots \cap A_n = \emptyset} m_1^\Theta(A_1) \times \cdots \times m_n^\Theta(A_n).
\]

However, this normalization process is irrational when the value of \( K_{12 \ldots n} \) is high, as a small BBA can be normalized to a very large value. To deal with this case, Yager’s (1987) combination rule is a feasible solution to solve this problem by assigning \( K_{12 \ldots n} \) to an unknown proposition, i.e. the BBA for the whole frame of discernment, expressed as:

\[
m_{12 \ldots n}^\Theta(A) = m_1^\Theta(A), \quad A \neq \emptyset, A \neq \emptyset.
\]

\[
m_{12 \ldots n}^\emptyset(\emptyset) = m_1^\emptyset(\emptyset) + K_{12 \ldots n}.
\]

where \( m_{12 \ldots n}^\emptyset(\emptyset) \) is the BBA for unknown proposition superimposed by the degree of conflict, according to Yager’s combination rule.
A so-called pignistic probability denoted by $\text{BetP}$ is used as a probability measure for decision-making, defined as:

$$\forall \theta_i \in \Theta \Rightarrow \text{BetP}^\theta(\theta_i) = \sum_{\theta_j \in A \subseteq \Theta} \frac{m_{12}^{\theta}(A)}{|A|}.$$  

(7)

where $\theta_i$ is one of the elements consisted by $\Theta$, and $|A|$ is the number of elements of $\Theta$ in $A$. This transformation between $m^\theta$ and $\text{BetP}^\theta$ is called the pignistic transformation. Generally, the element with the largest value of $\text{BetP}$ is the optimal solution.

2.2.2 Track management

In a general track management framework, three track states including young, mature and erased are used (Milan et al., 2017; Linder et al., 2016). A young track transits to a mature one if it is associated with detections successful in several frames after its generation. And when no detections are assigned to a mature track for a set amount of time, its state transits to an erased one, being deleted from the track list.

However, this general framework does not work well for MTT in dense traffic environment, as the performance of all sensors become very poor. For instance, in addition to interested dynamic objects on the road, the Radar detects other metallic reflectors, e.g. the guardrails, traffic signs and parked cars at roadside and low overpasses. Although these stationary objects can be discarded at detection level according to their near to zero speed, this approach may accidentally ignore interested objects such as a vehicle waiting at a traffic light in front of ego car. What is worse, the point clouds become dense in this situation, which results in higher over and under-segmentation rates when clustering them to model objects. These imperfect performances of sensors can affect data association and further have a negative impact on the decision and planning of AD.

To address these issues, a novel DST-based track management framework is proposed, which includes four states in the frame of discernment of track life, namely, new ($N$), mature and important ($MI$), mature but unimportant ($MU$) and erased ($E$). In particular, the state of $MU$ represents that the track belongs to an object that is not interested and unnecessary to output to the decision layer. The transition among these four states is realized using equations (5) and (6), according to the sources of historical BBAs and current BBAs for these states.

To instantiate it, let $\Theta_{\text{track}} = \{N, MI, MU, \ldots, E\}$ be the frame of discernment towards the problem of track management. The combined BBA for each non-empty subset $A$ which belongs to $\Theta_{\text{track}}$ is calculated using Yager’s combination rule:

$$m_{12}^{\Theta_{\text{track}}}(A) = \sum_{A_1 \cap A_2 = A} m_1^{\Theta_{\text{track}}}(A_1) \times m_2^{\Theta_{\text{track}}}(A_2).$$  

(8)

$$m_{12}^\Theta(\emptyset) = m_{12}^{\Theta_{\text{max}}}(\emptyset) + K_{12}.$$  

(9)

where $m_1^{\Theta_{\text{track}}}(A_i)$ and $m_2^{\Theta_{\text{track}}}(A_j)$ are BBAs from two different sources, which are the historical track state from the last frame and the current track state respectively. The latter sets BBA according to some modules in track management.

As discussed briefly in Section 2, the track management involves detection/track data association, pre-tracking, track prediction, track updating and track deleting. The state transition of track life realized in the track management is illustrated in Figure 2. After data association, the state of all tracks obtains clear evidence to be updated. To begin with, the sensor detections not associated with any tracks are used to generate new tracks, of which the BBA for $N$ almost equal to 1. For tracks which are not associated with any detections, the BBA for $E$ in the current track state is set higher than it for assigned tracks. Especially, if a track fails to match with detections for some frames, its BBA for $E$ will be increased gradually. Once its value is greater than a specific threshold, the track deleting module will erase this track. On the contrary, an assigned track tends to be stable; as a result, its BBAs for $MI$ and $MU$ in current track state are set a high value in track updating module. As for a new track, track maturing process checks if it is mature enough (e.g. after 5 frames) to be a mature one. As for a mature track, it keeps its state of $MI$ or $MU$ after associating successfully. Even a track tented to be the state of $E$, as long as it is re-associated currently again, it can also be a mature one. Similarly, the BBAs for these two states increase during the process of track maturing in the pre-tracking module when a new track is matched.

Besides, some importance judgments are designed to judge whether a track is important and necessary to be output to the decision layer by setting the BBAs for $MI$ and $MU$, i.e. track selection from $MU$ to $MI$ and track discarding from $MI$ to $MU$. For instance, a frontal object which is nearly still and only detected by Radar is likely to generate an unimportant track and to be discarded, e.g. a traffic sign or a low overpass; an object with a high velocity is quite important to be tracked; an object which was detected to be moved but still at the moment is possible to be important, as it might be a car waiting for a traffic light near the ego car. The BBA update of $MI$ and $MU$ is done during the process of track selection in the pre-tracking module and also in track updating module, which can guarantee a sustaining concern on the importance of a track before it is erased.
2.2.3 Data association

Illuminated but not limited by Chavez-Garcia et al.’s (2014) study, a detection-level data association which uses the evidence of type, position, and velocity of detected objects is proposed. In dense traffic scenarios, some sensors may provide incorrect type information of detection to the fusion center. On this occasion, fusion at detection level should be robust to avoid erroneous assignment between tracks and detections. DST is a rational method to deal with this kind of uncertainty and conflict information. Hence, instead of only one certain class hypothesis of tracks and detections, the type feature is represented by an evidence mass distribution of all kinds of types. On top of this, object detections and tracks include position and velocity information, which serves as the kinematic evidence of association.

In the fusion framework, the track prediction module provides a list of \( m \) predicted tracks denoted by \( T = t_1, t_2, \ldots, t_m \), while the data buffer module outputs a list of \( n \) detections denoted by \( D = d_1, d_2, \ldots, d_n \). We consider a frame of discernment to describe the association between \( T \) and \( D \), which is expressed by \( \Theta_{\text{asso}} = \{ 0, 1 \} \). And three BBAs imply three propositions are defined as:

- \( m_{n,d}(\{1\}) = 1 \): when \( t_i \) and \( d'_j \) are from the same object;
- \( m_{n,d}(\{0\}) = 1 \): when \( t_i \) and \( d'_j \) are from different objects; and
- \( m_{n,d}(\{0, 1\}) = 1 \): when we know nothing about the object source of \( t_i \) and \( d'_j \).

As discussed before, three kinds of evidence sources can be used to determine if they are matched, which are type similarity \( m^t \), position similarity \( m^p \), and velocity similarity \( m^v \). According to Yager’s combination rule, \( m_{n,d} \) can be represented by \( m^t, m^p \) and \( m^v \) as:

\[
    m_{n,d}(A) = \sum_{A_1 \subset A, A_2 \subset A} m^t_{i,d}(A_1) \times m^p_{i,d}(A_2) \times m^v_{i,d}(A_3) \tag{10}
\]

where \( A, A_1, A_2 \) and \( A_3 \) are nonempty subsets of \( \Theta_{\text{asso}} \).

If the value of \( \text{BetP}(\{1\}) \) is over a specific threshold, the track \( t_i \) and detection \( d_j \) are matched together. Afterwards, global nearest neighbor and JPDA are used to get an optimal or a statistically most possible update from all of the matched candidates for the association to a track. The former method is applied to the detection from the camera, of which the false detection rate and over-segmentation rate are low. The latter is used in the detection from Radar and LiDAR, as they easily detect clutters and sometimes generate more than one detections from an object.

As for type similarity, it is not fully convincing to imply that a track and a detection are from the same object, even if their object types are same. This is because of possible wrong detection of the sensors and many targets with same types in a dense driving scenario. However, it provides a strong evidence to conclude that they are probably unmatched if their object types are different. Considering the frame of discernment \( \Theta_{\text{type}} = \{ \text{car, truck, pedestrian, cyclist} \} \), the BBA \( m^t_{n,d} \) is set as follows:

\[
\begin{align*}
    m^t_{n,d}(\{1\}) &= 0, \\
    m^t_{n,d}(\{0\}) &= \sum_{A \in \Theta_{\text{type}}} m^t_{n,d}(A) \times m^t_{d,A}(B), \\
    \forall A, B \subset \Theta_{\text{type}} \\
    m^t_{n,d}(\{0, 1\}) &= 1 - m^t_{n,d}(\{0\}).
\end{align*}
\tag{13}
\]

where \( m^t_{i,d} \) is updated based on the type BBA of a track from the last frame and the type BBA of the matched detection using Yager’s combination rule, and \( m^t_{d,A} \) is set according to the detection information of each sensor, which is discussed below.

Radar sensors can provide an accurate estimation of the position and velocity of a detected object. However, they cannot tell exactly what type it belongs to. Still, estimation can be derived from experience that a high-velocity object is more likely to be a car or a truck, and an object with not too low speed is improbable to be a pedestrian. The camera used in our vehicle platform mentioned in Section 2 has a built-in detection algorithm, which outputs the information of frontal targets including type, position, velocity and shape. As a result, the BBA for one type \( m^t_{d,A}(A) \) will be set a high value if it is the type of an object implied by the camera detection. What’s more, the LiDAR provides abundant information of three-dimension shape after a clustering process for its point clouds. Hence, \( m^v_{d,A}(A) \) can be set based on the shape size, width and height from the LiDAR detection.

As for position similarity, the smaller the Mahalanobis distance \( d_{n,d} \) between a predicted track \( t_i \) and a detection \( d_j \) is, the more likely they are from the same object. Thus, the BBA for position \( m^p_{n,d} \) is distributed as follows:

\[
\begin{align*}
    m^p_{n,d}(\{1\}) &= \alpha f(d_{n,d}), \\
    m^p_{n,d}(\{0\}) &= \alpha(1 - f(d_{n,d})), \\
    m^p_{n,d}(\{0, 1\}) &= 1 - \alpha f.
\end{align*}
\tag{14}
\]

where \( \alpha \in [0, 1] \) is a discounting factor for velocity precision of a sensor, and \( f(d_{n,d}) \) is a function describing the negative relationship between \( d_{n,d} \) and the velocity similarity.

In this part, the architecture of the detection-level perception approach based on DST is proposed. To avoid tracking uninterested targets and losing important targets, the concept of multi-state track life is integrated into a generic fusion framework to improve the performance of multi-object perception. Moreover, the information of object type, position and velocity is used to reduce wrong data association between tracks and detections. It is important to point it out that the output from this perception approach includes information of motion states, appearance and type of the surrounding targets.

3. Traffic risk assessment

After the multi-object perception is processed and the information of kinetic states, appearance and type of the surrounding targets is obtained, this section presents a novel
traffic risk assessment method based on the dynamic analysis of vehicles in different driving scenarios. The results of multi-object perception will be the input in this section.

3.1 Road traffic risk

While economic development and social progress are increasingly dependent on transportation, traffic accidents attract more and more attention. Traffic risk, the likelihood of having a traffic accident, is an integral part of human-vehicle-road environment closed-loop traffic system. In addition, traffic risk even exists in near-crash scenarios. To reduce traffic risks, we need to analyze influencing factors in human, vehicle, and road environment. Statistics show that 80 per cent of the road traffic accidents occurred in the straight road scenario (TMBPSM, 2016). Therefore, this scenario became the main research object in this paper.

3.1.1 The road traffic risk caused by a single moving object

The essence of the collision process is converting kinetic energy into frictional heat energy, elastic and plastic deformation. In a traffic system, moving objects such as vehicles, pedestrians and cyclists have kinetic energy. Therefore, traffic risk can be described in terms of the kinetic energy of moving objects. The kinetic energy of a moving object is:

\[ E_i = \frac{1}{2} m_i v_i^2 = \frac{1}{2} m_i v_i \cdot v_i = \frac{1}{2} m_i v_i \cdot (v_i - 0) \cdot \Delta x_i \]  

(15)

where \( E_i \), \( m_i \) and \( v_i \) are the kinetic energy, mass and velocity of moving object \( i \), respectively, and the \( \Delta x_i \) is the distance between \( i \) and an arbitrary point in front of it.

Let \( F_i = 1/2 m_i v_i^2 \cdot (v_i - 0)/\Delta x_i \), therefore,

\[ E_i = F_i \cdot \Delta x_i \]  

(16)

where \( F_i \) denotes the equivalent force in the traffic environment exerted by the moving object \( i \). It is measured in Newton.

When an obstacle \( j \) appears in front of the moving object \( i \), a relationship between the two emerges. Here, we use \( E_{ij} \) to describe this relationship.

\[ E_{ij} = \frac{1}{2} m_i v_i \cdot (v_i - v_j) \cdot \frac{|x_i - x_j|}{|x_i - x_j|} = \frac{1}{2} m_i v_i \cdot v_i - v_j \cdot |x_i - x_j| \]  

(17)

where \( v_j = 0 \), it is the velocity of the obstacle \( j \). \( x_j \) is the longitudinal position of the obstacle \( j \). According to equation (17), the \( (v_i - v_j)/|x_i - x_j| \) means the result of dividing relative velocity by relative distance between the moving object \( i \) and the obstacle \( j \). This physical quantity in automotive engineering represents the TTC\( i \). Therefore, equation (17) can be written as follows:

\[ E_{ij} = \frac{1}{2} m_i v_i \cdot \text{TTC} \cdot |x_i - x_j| \]  

(18)

Similarly, we set \( F_{ij} = 1/2 m_i v_i^2 \cdot \text{TTCi} \), which indicates the internal equivalent force between the moving object \( i \) and the obstacle \( j \). It is measured in Newton.

3.1.2 The road traffic risk in car-following scenario

The car-following scenario is a typical scenario in the traffic environment. With the rapid increase of auto ownership, vehicles frequently appear in platoons or clusters on city roads and highways. A car-following scenario is shown in Figure 3. \( s_{ij} \) denotes the space occupied by vehicles \( i \) and \( j \) in the traffic environment, \( s_j \) the space headway of this two vehicles, \( L_j \) the length of vehicle \( j \) and \( x_i \) and \( x_j \) denote the longitudinal position of the vehicle \( i \) and vehicle \( j \), respectively.

The traffic risk caused by vehicle \( i \) and vehicle \( j \) is the same as the single moving object form and is defined as follows:

\[ E_i = \frac{1}{2} m_i v_i \cdot (v_i - 0) \cdot \Delta x_i \]  

(19)

\[ E_j = \frac{1}{2} m_j v_j \cdot (v_j - 0) \cdot \Delta x_j \]  

(20)

Next, a collision event only occurs between the front of the following vehicle \( i \) and the rear of the leading vehicle \( j \) in this car-following scenario. In other words, if we set Event A to denote the scenario that vehicle \( i \) crashes into vehicle \( j \) and Event B to denote that vehicle \( j \) crashes into vehicle \( i \), the probability of Event A must be greater than zero and the probability of Event B is virtually zero. Therefore, we define the following vehicle as an active-collision participant (ACP) and the leading vehicle as a passive-collision participant (PCP). The traffic risk between the ACP and PCP are defined as follows:

\[ E_{ij} = \frac{1}{2} m_i v_i \cdot (v_i - v_j) \cdot |x_i - x_j| = \frac{1}{2} m_i v_i \cdot \text{TTCi} \cdot |x_i - x_j| \]  

(21)

Similarly, we set \( F_{ij} = 1/2 m_i v_i^2 \cdot \text{TTCi} \), which indicates the internal equivalent force between the vehicle \( i \) and the vehicle \( j \). Its unit is Newton.

Hence, the traffic risk of the road environment in the car-following scenario can be defined as follows:

\[ E = E_i + E_j + E_{ij} \]  

(22)

3.1.3 The road traffic risk in arbitrary scenarios

The above section described the relationship between two vehicles in a car-following scenario. However, the car-following scenario is often disturbed by vehicles cutting in. A cut-in scenario is shown in Figure 4, where \( (x_i, y_i), (x_j, y_j) \), \( v_i \) and \( v_j \) denote the positions and velocities of vehicle \( i \) and vehicle \( j \), \( v_j \) and \( d_j \) denote the relative velocity and distance between vehicle

![Figure 3 Car-following scenario](image-url)
and vehicle \( j \), respectively, \( d_i^* \) the minimum relative distance, \( \theta_{ij} \) the interior angle between \( v_i \) and \( d_i \) and \( \theta_{ij}^* \) the angle from \( v_i \) to \( d_i \) with counterclockwise being the positive direction. Therefore, the maximum force on vehicle \( j \) exerted by vehicle \( i \) is calculated as follows:

\[
F_{ij,max} = \frac{1}{2} m_i v_i \cdot \frac{v_j \cos(\theta_{ij} + \theta_{ij}^*)}{\sqrt{d_i^*-d_i^2}} = \frac{1}{2} m_i v_i \cdot \frac{v_j \cos(\theta_{ij} + \theta_{ij}^*)}{d_i \cos \theta_{ij}} \tag{23}
\]

In addition, Figure 4 shows an instantaneous scenario; the cut-in action of vehicle \( j \) is a continuous process; all variables in Figure are time-varying; \( F_{ij,max} \) has the same properties as well. The arbitrary two-vehicle scenario (Figure 5) can be analyzed by using the same method. Hence, The traffic risk between vehicle \( i \) and vehicle \( j \) is derived as follows:

\[
E_{ij} = F_{ij,max} \cdot d_i = \frac{1}{2} m_i v_i \cdot \frac{v_j (\cos \theta_{ij}^* - \tan \theta_{ij} \sin \theta_{ij}^*)}{d_i} \tag{24}
\]

Similarly, the traffic risk of the road environment in the arbitrary two-vehicle scenario can also be described as equation (22).

### 3.2 The range of road traffic risk

The road traffic risk is always resulted from road users and the road traffic environment. It is related to the motion states of road users and the road environment conditions. A road traffic accident occurs because the road user does not recognize the traffic risk caused by others or does not take the traffic risk caused by itself under control in advance. In response to this situation, the range of road traffic risk is proposed in this paper; meanwhile, the mathematic model of this risk range is established.

We assume that the road user obeys traffic rules, e.g. forward driving without reversing and turning and changing lanes as appropriate. Vehicle velocity and steer angle are treated as continuous variables. As such, the positions of the vehicle can be predicted over time, based on which its trajectory can be projected as illustrated in Figure 6 where red dots are prediction positions and blue curves are the projected trajectories.

Symbol \( F \) indicates the influence of vehicle \( i \) at each position, and turning radius \( R \) can be calculated using the equivalent linear two-wheel vehicle model as follows:

\[
R(t) = \left[ 1 + K \kappa^2_i(t) \right] \frac{L}{\dot{\delta}(t)} \tag{25}
\]

where \( K \) indicates the stability factor, \( L \) is the wheelbase of vehicle \( i \), \( \delta \) denotes the steering angle. \( v_i \) is the velocity of vehicle \( i \).

When vehicle \( i \) drives at a constant velocity with negligible side slip angle, the predicted positions \( (x_{ip}, y_{ip}) \) at a time horizon \( t_p \) with a command steer angle \( \delta \) can be calculated as follows:

\[
\begin{bmatrix}
x_{ip} \\
y_{ip}
\end{bmatrix} = 
\begin{bmatrix}
v_i \\
\int_{t_0}^{t_p} v_i(t) \cdot \cos \frac{\nu_i(t) \cdot \Delta t}{R(t)} dt \\
\int_{t_0}^{t_p} v_i(t) \cdot \sin \frac{\nu_i(t) \cdot \Delta t}{R(t)} dt
\end{bmatrix} \tag{26}
\]

where \( \Delta t \) denotes the unit time, in addition, \( \Delta t = 1s \).

Assume vehicle \( i \) is always under control with driving stability. The maximum value of velocity and turning radius should be subject to the road conditions. The motion states of vehicle \( i \) are subject to the following formulas:

\[
\sqrt{F_X^2 + F_Y^2} = \varphi F_Z \tag{27}
\]

\[
F_X = m_i g f + \frac{C_D A v_i^2(t)}{21.15} \tag{28}
\]

\[
F_Y \geq m_i \frac{v_i^2(t)}{R(t)} \tag{29}
\]

where \( F_X \) and \( F_Y \) denote the longitudinal and lateral force of vehicle \( i \), respectively, \( F_Z \) the ground reaction forces, \( \varphi \) the adhesion coefficient, \( f \) the rolling resistance coefficient, \( C_D \) the air resistance coefficient and \( A \) the windward area of vehicle \( i \).
Based on equations (25), (27), (28) and (29), the relationship between steer angle $\delta$ and velocity $v_i$ can be derived as follows:

$$|\delta(t)| \leq \frac{K}{M} + \frac{1}{M \cdot v_i^2(t)} \sqrt{N - 2F_j W \cdot v_i^2(t) - W^2 \cdot v_i^4(t)}$$  

(30)

where

$$W = \frac{C_D A}{21.15}$$  

(31)

$$N = \varphi^2 F_j^2 - m_i^2 \sigma^2 f^2$$  

(32)

$$M = m_i/L$$  

(33)

$$F_j = m_i g f$$  

(34)

With the increase of driving velocity of vehicle $i$, the allowable steer angle $\delta$ decreased according to equation (30). Meanwhile, the steer angle $\delta_i$ is constrained by the mechanical structure of the vehicle. The maximum value is equal to the steer angle limit $\delta_{max}$. Generally, $\delta_{max} \in [-\pi/4, \pi/4]$ for a passenger car.

Next, the probable motion trajectory of vehicle $i$ should have a certain boundary according to the steer angle range, and the motion states of vehicle $i$ are stabilized within this boundary absolutely. As shown in Figure 7, the black dotted curves denote the left and right limit of the prediction trajectory. When vehicle $i$ is driving straight on the road, the driver may take the following action, straight driving, turn to the left lane and turn to the right lane, all of which are controlled through steering wheel. Let the steering angle and the turning-probability be $\delta_k$ and $p_k$, respectively. Therefore, the turning-probability $p_k$ can be defined as follows:

$$\sum_{k=-n}^{n} p_k = 1$$  

(36)

$$\delta_k = k \cdot \Delta \delta, \ k \in [-n, n]$$  

(37)

where $k, n \in \mathbb{Z}$. $\Delta \delta$ indicates the increment of the steering angle. In addition, $\delta_0$ denotes straight driving, $\delta_k$ indicates turning left if $k$ is a positive integer, otherwise, the $\delta_k$ denotes turning right.

However, it is difficult to predict steer angle of the driver and assign it a corresponding value for the turning-probability. To solve this problem, we use real free driving experimental data. The details of the experimental route are shown in Figure 8. This free driving database contains a significant amount of original experiment data of 33 actual experienced drivers, including GPS data and vehicle data. In addition, this database contains about 32.5 h and more than 1,160,000 measuring points of highway experiment data. Therefore, we count all the highway experiment statistics data to analyze the steering angle of the drivers. The probability of steering angles in highway section basically presents the Gauss normal distribution. The details of the result are shown in Figure 9. And the Gauss normal distribution is defined as follows:

$$p_k(\delta_k \mid \mu, \sigma) = \frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{(\delta_k - \mu)^2}{2\sigma^2}}$$  

(38)

### 3.3 The traffic safety field concept

This subsection describes a new concept of the traffic safety field. The traffic risk and its distribution region are described in the above subsections. We use a series of equivalent force to describe the potential impact of the traffic environment which is caused by a road user. Guided by this principle, the road environment will be covered with this kind of force when road users are moving on the road, including vehicles, pedestrians and cyclists. Furthermore, as previously mentioned, the traffic risk is caused by the ACP and the PCP. The active or passive is a relative concept. In the real traffic environment, each road user can display as an ACP or PCP in different time and space. The safety rate of the road environment can be quantized by analyzing the range and the distribution of the equivalent force. Therefore, we named this force range as the traffic safety field. The value of equivalent force decreases with the distance between the predicted point and the road user increases. Similarly, the value of equivalent force decreasing laterally to

![Figure 8 The experimental route](Image 328x65 to 565x235)
both sides, and the weight of the equivalent force is defined as $w_k$ in the pictorial diagram of Figure 10(a). Meanwhile, the weight $w_k$ is defined as follows:

$$w_k = \frac{p_k(\delta_k)}{p_0(\delta_0)}$$  \hfill (39)

where $k \in [-n, n]$ and $k, n \in \mathbb{Z}$. $p_0(\delta_0)$ denotes the probability of vehicle stayed at angle $\delta_0$ by the driver in the next moment, $p_k(\delta_k)$ the probability of vehicle steered to angle $\delta_k$ by the driver in the next moment.

Based on equation (39), the equivalent force in each predictive position can be calculated as follows:

$$F_{ki} = \frac{E}{\Delta x_i} \cdot w_k = \frac{1}{2} w_k m_i v_i^2$$ \hfill (40)

Finally, the traffic risk map of the straight driving vehicle $i$ is described by MATLAB as shown in Figure 10(b) ($m_i = 1500\text{kg}$, $v_i = 20\text{m/s}$). The zero value of the equivalent force is set as the white color for convenient analysis. The boundaries of traffic risk influence area are illustrated obviously by arc curves which separated the white areas and deep blue areas in Figure 8 (b). In addition, the value of equivalent force decreases progressively with the increase of longitudinal and lateral distances. Moreover, the boundaries of the above influence area will change with the velocity of the vehicle and road conditions based on equation (16). It has a time-varying property. Therefore, the traffic risk map is a time-varying map.

### 4. Model verification for traffic risk assessment

To verify the effectiveness of the proposed fusion approach, we use a vehicle platform to collect datasets in real typical dense scenarios on highways and urban roads. It is important to point out that there is no rigorous definition for dense traffic scenario, which is empirically attributed to conditions with more than seven or eight road users such as cars, trucks, pedestrians and cyclists around ego car on average. A detection-level fusion approach with traditional track management incorporating states of new, mature and erased is used as the baseline approach (Chavez-Garcia et al., 2014).

#### 4.1 The vehicle platform

To realize the real-time function of all-around multi-object perception in dense traffic environment, a vehicle platform equipped with the Radars, a camera and a LiDAR is used, as presented in Figure 11. Two mid-range radars are mounted at the frontal bumper and one at the rear bumper for dynamic target detection, with the detection range of 160 m and the...
FOV of 20°. The camera with the detection range of 200 m and the FOV of 80° is located on the windshield for frontal target detection. Two side radars with the detection range of 80 m and the FOV of 150° are installed at the rear corners to enlarge the detection field. Moreover, the Velodyne HDL-64E rotating laser scanner with the detection range of 120 m and the FOV of 360 degrees is mounted on the roof for surrounding object perception.

4.2 The efficiency of multi-object perception
The generic DST-based fusion approach presented in Section 2 is used to enhance the complete perception framework that is to be evaluated by the collected datasets. To understand the surroundings of the ego vehicle better in the datasets, a visualization interface is designed based on the robot operating system, as shown in Figure 12. It realizes the function of online and offline display of the point clouds from the LiDAR, the original detections of all the sensors and bounding boxes forms which are not shown in Figure 12. The fusion results in the form of red bounding boxes and the videos in the left subfigures collected by some other onboard cameras except for the frontal camera are mentioned in Section 2. In this part, we design multiple metrics to evaluate the proposed fusion approach as suggested by detection-level fusion (Chavez-Garcia et al., 2014) and the multiple objects tracking benchmark (Luo et al., 2014). Unlike the number of misclassifications used in Chavez-Garcia et al.’s (2014) study, we extend the metrics to adopt ground truth (GT), false positive (FP) and false negative (FN), as well as false positive rate (FPR) and false negative rate (FNR), which are defined as:

\[
\begin{align*}
FPR &= \frac{\sum_{i=1}^{N} FP_i}{\sum_{i=1}^{N} GT_i}, \\
FNR &= \frac{\sum_{i=1}^{N} FN_i}{\sum_{i=1}^{N} GT_i}.
\end{align*}
\]

(41)

where \( N \) represents the number of the frames, and \( GT_i, i = 1, 2, \ldots, N \), is the number of the targets in \( i \)-th frame. Besides, the basic concepts of \( GT, FP \) and \( FN \) are same as them in W. Luo et al.’s (2014) research. Among them, \( FP \) represents the fusion approaches output a fused track that is actually not from the interested target, while \( FN \) means a real interested target is not tracked using the fusion approaches. Besides, an \( FP \) tracking may lead to an unnecessary brake, and an \( FN \) tracking may even bring about a collision accident, which is undesired to be occurred on the road. Furthermore, the real number of interested object around the ego car is represented as \( GT \).

By using the visualization interface to replay the datasets of two highway scenarios and one urban scenario, the fusion results of DST-based fusion and baseline approach are gathered, as illustrated in Tables I and II. The former shows the aggregate results of two approaches in highway and urban scenarios, which contain the \( GT, FP \) and \( FN \) of all the frames with the range of 0 to 80 m. And the latter table represents the corresponding average results of the former to verify the tracking ability in different ranges. The results in Table I indicate that the \( FP \) of DST-based fusion approach is near half of it of the baseline, while the \( FN \) numbers of two approaches are basically equal. And the results in Table II imply that the \( FP \) and \( FN \) of the proposed approach in the smaller range are less than it in the larger range, which conforms to our experience.

4.3 The result of traffic risk assessment for intelligent driving
The position and motion information of the vehicle platform and the surrounding road users including their types, positions and velocities can be captured accurately by the vehicle platform based on the multisensor-fusion approach. In addition, the sampling rate is 50. Therefore, the equivalent force that is caused by the surrounding road users can be calculated based on the position and motion information. In this paper, the vehicle platform is defined as the ego vehicle. According to the multisensor-fusion data, the traffic risk in a braking process of the intelligent vehicle is illustrated in Figure 13 based on the details of the traffic risk assessment method in Section 3. The equivalent force loaded on the ego

<table>
<thead>
<tr>
<th>Scenario</th>
<th>GT</th>
<th>DST-based fusion</th>
<th>Baseline</th>
</tr>
</thead>
<tbody>
<tr>
<td>Highway 1</td>
<td>290</td>
<td>18</td>
<td>33</td>
</tr>
<tr>
<td>Highway 2</td>
<td>271</td>
<td>18</td>
<td>33</td>
</tr>
<tr>
<td>Urban road</td>
<td>216</td>
<td>22</td>
<td>46</td>
</tr>
</tbody>
</table>
vehicle is calculated according to the collected data. As shown in Figure 13(a), the driver starts to brake at around 40 ms, which means the traffic risk was recognized by the driver at that time. As mentioned in section 3, the traffic risk always generates along the possible trajectory of the moving object (illustrated in Figure 10). Therefore, the equivalent force caused by ego vehicle is illustrated in Figure 13(b), and the maximum value of equivalent force caused by ego vehicle appeared before 40 ms, which means the traffic risk assessment method can perceive the traffic risk in advance.

Generally, drivers are not concerned about traffic risk from behind, even when they fail to see the rear vehicle. Therefore, we are unable to obtain driver response to the traffic risk from behind. However, our traffic risk assessment can recognize the traffic risk caused by the following vehicles. As shown in Figure 13(c) and (d), the equivalent forces caused by rear vehicles and the closest following vehicle show high value,

### Table II  Average results of two approaches with two different ranges in highway and urban scenarios

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Frame no.</th>
<th>GT FPR (&lt;40m)</th>
<th>GT FNR (&lt;40m)</th>
<th>DST-based fusion (%) FPR (&lt;40m)</th>
<th>DST-based fusion (%) FNR (&lt;40m)</th>
<th>Baseline (%) FPR (&lt;40m)</th>
<th>Baseline (%) FNR (&lt;40m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Highway 1</td>
<td>25</td>
<td>163</td>
<td>127</td>
<td>4.91</td>
<td>7.87</td>
<td>1.23</td>
<td>3.15</td>
</tr>
<tr>
<td></td>
<td>40-80m</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>6.75</td>
<td>17.3</td>
</tr>
<tr>
<td>Highway 2</td>
<td>25</td>
<td>156</td>
<td>115</td>
<td>4.49</td>
<td>9.57</td>
<td>1.28</td>
<td>0.87</td>
</tr>
<tr>
<td></td>
<td>40-80m</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>5.77</td>
<td>20.9</td>
</tr>
<tr>
<td>Urban road</td>
<td>20</td>
<td>131</td>
<td>85</td>
<td>6.87</td>
<td>15.3</td>
<td>4.58</td>
<td>5.34</td>
</tr>
<tr>
<td></td>
<td>40-80m</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>13.7</td>
<td>21.4</td>
</tr>
</tbody>
</table>

**Figure 13** The test result

Notes: (a) Acceleration of vehicle; (b) equivalent force caused by ego vehicle; (c) equivalent force caused by rear vehicles; (d) equivalent force caused by the closest following vehicle

---

Dense traffic environment  
Xunjia Zheng, Bin Huang, Dasheng Ni and Qing Xu  
Journal of Intelligent and Connected Vehicles  
Volume 1 · Number 2 · 2018 · 41–54
which means ego vehicle was driving in a relatively high-risk environment during that period. With these questions, we look back at the experimental data and found that the car was continuously bypassed by other cars during this time as shown in Figure 14. In addition, the risk map of this continually overtaking scenario is shown in Figure 14(c). The coordinate of the vehicle platform is (0,0) in the risk map, and the detected four vehicles are shown with the equivalent force displays. The vehicle platform can easily recognize the distribution of traffic risk by this risk map, and this approach will benefit the safety control of intelligent vehicles.

5. Discussion

This study presented the concept of traffic safety field by embedding the equivalent force and established a new traffic risk model. Our result suggested that traffic safety field can capture the state of traffic risk within (dense) traffic environment. To make the model more intuitive and reasonable, the traffic risk is described by assigning an equivalent force vector to each point of a map by using this method. The norm of equivalent force vector represents the state of the risk at that point, which is similar to our previous research (Wang et al., 2014, 2015, 2016). Compared to our presented study, the principle of modeling has changed. The greatest improvement of this model is that it omitted a large number of undetermined parameters, which made it easier to apply. However, a few problems remain unsolved.

5.1 The relationship between the risk index and incident severity

The risk state of a collision process is depended on the kinetic energy of colliding objects in this study, and hence the risk level approximates the real situation. In our previous study, the risk level was assessed by using the relative driving safety index (RDSI). However, RDSI is based on a standard driving safety index (DSI) in a specific dangerous traffic scenario. It means that we must analyze dangerous scenarios as many as possible to calculate this index accurately. In this paper, we only used the equivalent force to describe the traffic risk, which is more intuitive and straightforward. We will keep improving the model structure by considering the risk index and incident severity. Meanwhile, components in the traffic environment will be separated into three categories: the first is normal road users including passenger cars, trucks and buses; the second includes obstacles such as barrier and traffic cones; the third is vulnerable road users such as pedestrians and cyclists. The risk index model will be established according to the equivalent impairment of traffic components which is caused by the equivalent force.

5.2 The real-world data analysis and its application

In this paper, the section of the model verification is described in a relatively simple way. Right now, the model only considers the impacts of moving vehicles without accounting for other factors such as road signs, lane markings and pedestrians. However, the applications in AD and traffic management are complex. The most challenging part of real applications is the surrounding environment information captured for the autonomous vehicle and the human–vehicle–environment interactions for traffic control, which can be solved in future research.

6. Conclusion

This paper described a novel multisensor-fusion-based multi-object tracking approach based on evidence theory and presented a road traffic risk assessment approach by embedding the equivalent force based on the traffic safety field concept. On one hand, we designed a generic DST-based detection-level fusion framework for multi-object perception to meet the perception requirement in dense traffic scenarios for ADAS and AD. Two novel points were put forward. For one thing, four tracking states (N, MI, MU and E) are defined and transformed naturally to each other in track management, which can distinguish interested/uninterested and birth/death

Figure 14 The continuously overtaking scenario

Notes: (a) Scenario case; (b) the fusion results; (c) risk map
tracks in dense scenarios. For another, the information of object type, position and velocity is used to offer evidence to data association module, reducing erroneous association between tracks and detections. We conducted experiments of real dense traffic environment on the highways and urban roads, and the result analysis indicated that the FPR tracking results were lower than those of the baseline approach, while FNR results were almost the same, which supported that our approach based on adapted DST is more robust to be implemented in track management and data association.

On the other hand, the relationship between the road user and the traffic environment or other road users was established by using mathematical derivations. This relationship indicated that the traffic risk is determined by the motion states of road users such as the velocity and steering angle of the vehicle, and it is also related to the environmental conditions such as the adhesion coefficient, the rolling resistance coefficient and the air resistance coefficient. Moreover, an accurate traffic risk in the traffic environment caused by a vehicle was calculated by considering the longitudinal and lateral influence range according to the real highway experiment statistical data. Finally, the road traffic risk was described as a field of equivalent force. Every road user generates their own field of equivalent force in the road traffic risk map, which was time-varying.

As a future plan, the shape of an object detection from LiDAR changes with different angles of view, which will be taken into consideration when using the shape information for data association. Furthermore, the traffic safety field will be described in detail by considering the driver–vehicle–road interaction. The risk level of every road user influenced by the traffic risk equivalent force will also be studied. Some driver assistant algorithms and AD algorithms can also be developed based on this traffic risk assessment approach.

**References**


**Corresponding author**

Qing Xu can be contacted at: qingxu@tsinghua.edu.cn

For instructions on how to order reprints of this article, please visit our website:

www.emeraldgrouppublishing.com/licensing/reprints.htm

Or contact us for further details: permissions@emeraldinsight.com
Stable trajectory planning and energy-efficiency control allocation of lane change maneuver for autonomous electric vehicle

Liwei Xu and Guodong Yin
School of Mechanical Engineering, Southeast University, Nanjing, China
Guangmin Li
Southeast University, Nanjing, China
Athar Hanif
Department of Electrical Engineering, COMSATS Institute of Information Technology, Lahore, Pakistan, and
Chentong Bian
Southeast University, Nanjing, China

Abstract
Purpose – The purpose of this paper is to investigate problems in performing stable lane changes and to find a solution to reduce energy consumption of autonomous electric vehicles.

Design/methodology/approach – An optimization algorithm, model predictive control (MPC) and Karush–Kuhn–Tucker (KKT) conditions are adopted to resolve the problems of obtaining optimal lane time, tracking dynamic reference and energy-efficient allocation. In this paper, the dynamic constraints of vehicles during lane change are first established based on the longitudinal and lateral force coupling characteristics and the nominal reference trajectory. Then, by optimizing the lane change time, the yaw rate and lateral acceleration that connect with the lane change time are limed. Furthermore, to assure the dynamic properties of autonomous vehicles, the real system inputs under the restraints are obtained by using the MPC method. Based on the gained inputs and the efficient map of brushless direct-current in-wheel motors (BLDC IWMs), the nonlinear cost function which combines vehicle dynamic and energy consumption is given and the KKT-based method is adopted.

Findings – The effectiveness of the proposed control system is verified by numerical simulations. Consequently, the proposed control system can successfully achieve stable trajectory planning, which means that the yaw rate and longitudinal and lateral acceleration of vehicle are within stability boundaries, which accomplishes accurate tracking control and decreases obvious energy consumption.

Originality/value – This paper proposes a solution to simultaneously satisfy stable lane change maneuvering and reduction of energy consumption for autonomous electric vehicles. Different from previous path planning researches in which only the geometric constraints are involved, this paper considers vehicle dynamics, and stability boundaries are established in path planning to ensure the feasibility of the generated reference path.

Keywords Autonomous electric vehicle, Energy-efficient control allocation, Lane change maneuver, Stable trajectory planning

Paper type Research paper

1. Introduction

Autonomous vehicles (AV) and electric vehicles (EV), wherein in-wheel motors (IWMs) are adopted to drive wheels, have attracted increasing attention from both industrial and academic communities recently. Autonomous driving technology has tremendous potential in reducing vehicle casualties, and IWM EV can immensely enhance energy efficiency and lead to flexibility actuation which considerably enhances vehicle maneuverability, stability and safety (Li et al., 2013; Jin et al., 2015; Yin et al., 2015). Numerous studies have revealed that A-IWM EV is an effective option that can increase...
traffic safety and decrease emissions and energy crisis (Li et al., 2017; Poduri and Singh, 2015).

Unlike manned vehicles, which follow the driver’s command to accomplish various driving tasks with the result that driver characteristics, vehicle dynamic features and energy management are major concerns (Wang et al., 2013, 2015, 2016; Wu et al., 2013; Dai et al., 2014), the AV is supposed to appropriately perform various maneuvers under rare driver interventions or even without drivers. Therefore, autonomous lane change and the corresponding abilities of trajectory planning and trajectory tracking are most significant for AV. Many research works have been conducted in lane-changing trajectory planning (Soubbaksh et al., 2013; Kim et al., 2014; Chen et al., 2014; You et al., 2015) and lane change control (Bayar, 2013; Berntorp et al., 2014; Naranjo et al., 2008). For example, Soubbaksh et al. (2013) evaluated three different path planning methods – state lattice, predictive constraint-based planning and spline-based search tree. Chen et al. (2014) proposed a feasible trajectory generation algorithm based on quartic Bezier curve to generate local trajectory for AV. You et al. (2015) adopted a polynomial method to describe the trajectory of AV carrying out the lane change maneuver. In comparison to conventional path planning strategies (such as road map, cell decomposition and potential field methods), which are constantly mentioned in the robotics field, the above curve-type path planning methods can greatly reduce calculation and avoid being stuck in the local minima. For lane change control, Bayar (2013) used the PID method to resolve the trajectory tracking control. In Berntorp et al. (2014), an optimal trajectory-based minimization of yaw acceleration was acquired, and the simulation and comparative analysis were done with different speed values. In Naranjo et al. (2008), the fuzzy controllers that mimicked human behavior and reactions were established to conduct AV executing the overtaking maneuver in the scenario of two vehicles overtaking.

Although the above research works on lane change path planning and lane change control have made great contributions, there are still some apparent shortages that need to be settled. To begin with, current researches about lane change trajectory planning only consider geometric constraints and kinematic characteristics (e.g. the road curvature and lateral acceleration); the restrictions associated with vehicle dynamic characteristics are normally neglected. Consequently, the vehicle’s dynamic stability may not be fulfilled if the AV drives along the predesigned trajectory. In addition, the problem of the A-IWM EV’s energy efficiency during lane change is rarely considered. For EV, especially for A-IWM EV, despite the redundant degrees of freedom providing additional control flexibility in maintaining vehicle safety and stability (such as Traction Control and Direct Yaw Control), unreasonable dynamic control laws that ignore the energy consumption may immensely shorten the driving mileage of EV.

Based on the aforementioned discussion, this paper presents a novel lane change control system for A-IWM EV, which consists of a stable trajectory planning level that ensures the feasibility of the generated reference path, a high-level model predictive control (MPC) and a low-level energy-efficient control allocation (EECA) scheme, to enhance the feasibility of lane changing and to reduce energy expenditure. The rest of this paper is organized as follows. In Section 2, the stable lane change trajectory that includes vehicle constraints is developed. A control-oriented model of IWM EV planar motion is described in Section 3. In Section 4, the control system is proposed. In Section 5, simulation results are displayed to verify the control performance and energy savings of the EECA. Conclusion is presented in Section 6.

2. Stable lane change trajectory

In this section, a new lane change trajectory that can guarantee the stability of A-IWM EV and keep the vehicle running smoothly is proposed. To establish this trajectory, the fifth-order polynomial function is first used to realize smooth lane change and the maximal comfortableness of passenger. Then, by founding the rational vehicle stability bounds and introducing those constraints into the trajectory equations, the stable lane change trajectory is created.

It should be noticed that in this paper, only the scenario of active lane change is considered, i.e. there should be no vehicles in the front and target lanes when the A-IWM EV is changing lanes. Therefore, the situation of collection avoidance is not considered in the reference trajectory generation. The corresponding path planning that can guarantee the stability of the vehicle and prevent vehicle collision at the same time can be studied in future research.

2.1 Stability constraints of in-wheel motors and electric vehicles

This section describes the plane dynamics of IMW EV. Hence, the stability constraints on longitudinal movement, lateral movement and yaw movement are constructed. In light of the vehicle dynamics, the lateral acceleration can be expressed as:

$$a_y = \omega v_x + \dot{v}_y$$  \hspace{1cm} (1)

where $\omega$ is the yaw rate, $v_x$ and $v_y$ are the longitudinal and lateral velocities. Denoting $\beta$ the slip angle of Center of Mass (CM), we get $v_x = v_y \tan(\beta)$. The relationship between the lateral acceleration, yaw rate and slip angle can be described as follows:

$$a_y = \omega v_x + \tan \beta \dot{v}_x + \frac{\dot{\beta} v_x}{\sqrt{1 + \tan^2 \beta}}$$  \hspace{1cm} (2)

Note that the lateral acceleration should not exceed the maximal force that the ground can offer. Suppose $\beta$ and $\dot{\beta}$ are small during vehicle lane change, the yaw rate of the vehicle under steady state should meet the following constraints (Rajamani, 2011):

$$|\omega| \leq \frac{\mu g \varepsilon}{v_x}$$  \hspace{1cm} (3)

where $\mu$ is the adhesion coefficient, $g$ is the gravity coefficient and $\varepsilon$ is the scale factor, which is usually approximately equal to 0.85 in practical calculation. In addition, because the linear tire model is used in this paper, the maximum lateral acceleration should not surpass 0.5g to ensure the tire working in the linear area, i.e.
The variations in yaw rate with regard to the different lane change times at 10 s, 8 s, 6 s and 4 s

Thus equation (3) is modified as:

\[ |\omega| \leq \frac{0.5e^{\mu_{g}y}}{v_{x}} \]  

(5)

For longitudinal acceleration \( a_{x} \), according to the adhere-circle restriction, as shown in Figure 1, the longitudinal acceleration should abide by the following inequality:

\[ |a_{x}| \leq \sqrt{\mu^{2}g^{2} - 0.25g^{2}} \]  

(6)

2.2 Reference trajectory generation

Assume \( t_{0} \) is the initial time of lane change, \( t_{f} \) is the window time of lane change and the initial and final lateral states of the vehicle during lane change are \([Y_{0}, v_{x0}, a_{y0}]T\) and \([Y_{f}, v_{xf}, a_{yf}]T\).

Hypothetically, if the vehicle carries out straight line driving before and after lane change, then \( a_{y0} = v_{y0} = a_{yf} = v_{yf} = 0 \).

According to the findings of Hult and Tabar (2013), the reference lateral curves of vehicle, which can guarantee the succession of later acceleration and jerk minimum, can be expressed by using the fifth-order polynomial function. Considering the initial and final lateral states of vehicle, this function can be written as:

\[ Y_{r}(t) = \left( Y_{0} - Y_{f} \right) \left( -6 \frac{t^{5}}{t_{f}^{5}} + 15 \frac{t^{4}}{t_{f}^{4}} + 10 \frac{t^{3}}{t_{f}^{3}} \right) + Y_{0}, t_{0} \leq t \leq t_{f} \]  

(7)

According to equation (7), \( t_{f} \) can be written as:

\[ t_{f} = \sqrt[3]{\frac{10|Y_{0} - Y_{f}|}{3|a_{y,max}|}} \]  

(8)

where \( a_{y,max} \) is the maximum lateral acceleration during lane change.

Moreover, insomuch as the longitudinal reference trajectory is normally longer than the lateral one, the following expression is adopted:

\[ X_{r}(t) = X_{0} + \int_{t_{0}}^{t} v_{x}dt, \quad t_{0} \leq t \leq t_{f} \]  

(9)

Figure 1 The constraints of the longitudinal and lateral forces

Note: The blue rectangle shows the constrained boundaries

It is noteworthy that the longitudinal acceleration is not constant. Considering the fluctuation of the longitudinal velocity in the actual steering process and the constraint of longitudinal jerk variation, the longitudinal acceleration is signified as:

\[ \ddot{a}_{x}(t) = \eta \sin(\kappa t) \]  

(10)

where \( \eta \) is the positive constant and \( \kappa = 2\pi/t_{f} \).

Equations (7)-(11) constitute the original reference trajectory that can maintain the continuity of steering and achieve the jerk minimum. To introduce vehicle dynamic restrictions, the yaw rate in ideal state is given:

\[ \omega_{r}(t) = \frac{v_{x}(t)}{\rho(t)} = \frac{v_{x}(X_{r}Y_{r} - X_{r}Y_{r})}{\sqrt{(X_{r}^{2} + Y_{r}^{2})^{3}}} \]  

(11)

where \( \rho(t) \) is the radius of curvature.

The maximum \( \omega_{r} \) is denoted by \( \omega_{r,max} \). Because the initial and final states of \( X_{r} \) and \( Y_{r} \) are certain, the value of \( \omega_{r,max} \) is only connected to \( t_{f} \). By restricting \( a_{y,max} \) \( \omega_{r,max} \) and \( \ddot{a}_{x} \) not outstripping the boundaries described in equations (4)-(6), the minimal lane change time \( t_{f} \) that can simultaneously fulfill the constraints of dynamics and the succession of later acceleration can be obtained. The new reference curve \((X_{r}^{+}, Y_{r}^{+})\) that can simultaneously pledge the vehicle stability and fulfill the jerk optimization is obtained.

Nevertheless, seeing that the order of \( \omega_{r} \) is generally high, it is difficult to give the explicit expression about \( \omega_{r,max} \). In consequence, the \( t_{f} \) is hard to gain. Actually, by observing the variation in the yaw rate of the vehicle driving along some curves, it can be perceived that the positive and negative maximum values always approximately arise at the quadrat and three-quarter lane time, i.e. \( t_{f}/4 \) and \( 3t_{f}/4 \) in Figure 2. Meanwhile, the maximum yaw rate \( \omega_{r,max} \) is monotone decreasing when the lane time increases. Therefore, can be represented as:

\[ \omega_{r,max} = \partial \min \left\{ \frac{\omega_{r}}{4}, \frac{3\omega_{r}}{4} \right\} \]  

(12)

Figure 2 The variations in yaw rate with regard to the different lane change times \( t_{f} \) at 10 s, 8 s, 6 s and 4 s

Note: The circle, square, pentagram and right-pointing triangle represent the yaw rates at \( t_{f}/4 \) and \( 3t_{f}/4 \)
where $\theta = 1.2$ is the penalty factor that offsets the loss of the authentic maximum of yaw rate. After $\omega_{\theta,\text{max}}$ is gained, the bisection search algorithm is used to seek the suitable $\tau$. The whole search algorithm is shown in Figure 3.

### 3. Vehicle model for A-IWM EV

#### 3.1 Dynamic model

In this section, as shown in Figure 4, the traditional vehicle model is used to describe the plane dynamics of A-EGV and the dynamic model can be written as:

$$\begin{align*}
\dot{X} &= v_x \cos \theta - v_y \sin \theta \\
\dot{Y} &= v_x \sin \theta + v_y \cos \theta \\
\dot{\theta} &= \omega \\
\dot{v}_x &= \omega v_y + (F_X - C_\alpha v_x^2 - C_m g) / m_v \\
\dot{v}_y &= -\omega v_x + F_Y / m_v \\
\dot{\omega} &= M_Z / I_Z
\end{align*}$$

(13)

where $m_v$ is the IWM EV mass, $I_z$ is the moment of inertia, $X$ and $Y$ are the longitudinal and lateral coordinates of vehicle in the inertial frame, $\theta$ is the heading angle of vehicle, $F_X$ and $F_Y$ are, respectively, the longitudinal and lateral forces, $M_z$ is the aerodynamic resistant force, $C_a$ and $C_m$ are the aerodynamic resistance coefficient and the rolling resistance coefficient, respectively.

The forces $F_X$, $F_Y$ and moment $M_z$ that are related to the four tire forces and the front steering angle $\delta$ can be expressed as:

$$\begin{align*}
F_X &= (F_{x1} + F_{x4}) \cos \delta - (F_{x1} + F_{x4}) \sin \delta + F_{x1} + F_{x4} \\
F_Y &= (F_{x1} + F_{x4}) \sin \delta + (F_{x1} + F_{x4}) \cos \delta + F_{x2} + F_{x3} \\
M_z &= \frac{I_y}{2} \left((F_{x1} - F_{x4}) \cos \delta + (F_{x1} - F_{x4}) \sin \delta - F_{x1} + F_{x4} + f_1 \left(F_{x2} + F_{x3} - f_2 \left(F_{x2} + F_{x3} + f_3 \left(F_{x1} \sin \delta + F_{x1} \cos \delta + F_{x4} \sin \delta + F_{x4} \cos \delta\right)\right)\right)
\end{align*}$$

(14)

In equation (14), $F_{x1}$ and $F_{x4}$ are, respectively, the longitudinal and lateral tire forces, where $i = 1, 2, 3, 4$ represents different wheels, $l_1$ is the track, $l_2$ and $l_3$ are the front and rear CM distances.

For the lateral tire force, $F_{x1}$, when the vehicle lateral acceleration is small and the dynamics of the tire is in the linear region, the following linear-lateral-tire-forces model can be used in vehicle control:

$$\begin{align*}
F_{x1} &= F_{x4} = K_f \alpha_f = K_f \left(\delta - (v_y + l_1 \omega) / v_x\right) \\
F_{x2} &= F_{x3} = K_s \alpha_s = K_s \left(l_2 \omega - v_y\right) / v_x
\end{align*}$$

(15)

where $K_f$ and $K_s$ are, respectively, the tire lateral stiffness, $\alpha_f$ and $\alpha_s$ are the front and rear tire slip angles.

Let $x = [x_1, x_2, x_3, x_4]^T = [\chi, \chi_x, \chi_y, \chi_\theta]^T$ and $u = [u_1, u_2, u_3, u_4, u_5]^T = [\delta, F_{x1}, F_{x2}, F_{x3}, F_{x4}]^T$ be the states and control inputs of system in equation (1); $T$ is the interval, then its discrete time form can be written as:

$$\chi(k + 1) = \chi(k) + T f(\chi(k), u(k))$$

(16)

where $f(\cdot)$ represents the nonlinear terms in equation (14).

Note that for the sake of expression, the discrete states and inputs at time $k$ are written as $\chi_k = [x_1^k, x_2^k, x_3^k, x_4^k]^T$ and $u_k = [u_1^k, u_2^k, u_3^k, u_4^k, u_5^k]^T$.

The state trajectory, denoted by $\chi_{k + 1}$, is obtained by applying the input $u_k = u_{k - 1}$ to system (13) at the time $k$ with $\chi_k = \chi_{k - 1}$; i.e.:

$$\chi_{k + 1} = \chi_k + T f(\chi_k, u_k)$$

(17)

In the light of equation (17), the nonlinear system (16) can be transformed into a linear time varying (LTV) system linearized at each time step $k$ around the point $(\chi_k, u_k)$ as follows:

$$\chi_{k + 1} = A_k \chi_k + B_k u_k + d_k$$

(18)

With
where

\[ a_1 = -T(\ddot{\chi}_k \cos \dot{\chi}_k + \dot{\chi}_k \sin \dot{\chi}_k), \]

\[ a_2 = T(\ddot{\chi}_k \cos \dot{\chi}_k - \dot{\chi}_k \sin \dot{\chi}_k), \]

\[ a_3 = 1 - \frac{2T}{m_v(\ddot{\chi}_4)^2} (C_a(\ddot{\chi}_4)^3 + K_f \sin \ddot{u}_1(\ddot{\chi}_5 + \dot{\chi}_6)), \]

\[ a_4 = T\ddot{\chi}_k + \frac{2TK_f \sin \ddot{u}_1}{m_v \ddot{\chi}_4}, \]

\[ a_5 = T\dot{\chi}_5 + \frac{2TK_f \sin \ddot{u}_1}{m_v \ddot{\chi}_4}, \]

\[ a_6 = -\ddot{\chi}_6 T + \frac{2T}{m_v} \left( K_e \ddot{\chi}_6 - \dot{\chi}_k \ddot{\chi}_6 \right) + \frac{K_f \cos \ddot{u}_1(\ddot{\chi}_5 + \dot{\chi}_6)}{(\ddot{\chi}_4)^2}, \]

\[ a_7 = 1 - \frac{2T(K_e + K_f \cos \ddot{u}_1)}{\ddot{\chi}_4}, \]

\[ a_8 = -T\dot{\chi}_4 + \frac{2T}{m_v} \left( \frac{K_e}{\ddot{\chi}_4} - \frac{K_f \cos \ddot{u}_1}{\ddot{\chi}_4} \right), \]

\[ a_9 = \frac{2T}{I_z} \left( \frac{K_e(\ddot{\chi}_5 - \dot{\chi}_k \ddot{\chi}_6)}{(\ddot{\chi}_4)^2} + \frac{K_f \cos \ddot{u}_1(\ddot{\chi}_5 + \dot{\chi}_6)}{(\ddot{\chi}_4)^2} \right), \]

\[ a_{10} = \frac{2T}{I_z} \left( \frac{K_e \ddot{\chi}_5 - K_f \cos \ddot{u}_1}{\ddot{\chi}_4} \right), \]

\[ a_{11} = 1 - \frac{2T(K_e \ddot{\chi}_6 + K_f \cos \ddot{u}_1 + K_f \ddot{u}_1)}{\ddot{\chi}_4}, \]

\[ b_1 = -\frac{T}{m_v} \left( \ddot{u}_2 + \ddot{u}_5 \right) \sin \ddot{u}_1 + 2K_f \left( \sin \ddot{u}_1 \right) \]

\[ + \cos \ddot{u}_1 \left( \ddot{u}_1 - \frac{\dot{\chi}_5 + \dot{\chi}_6}{\ddot{\chi}_4} \right). \]

\[ b_2 = \frac{T}{m_v} \left( \ddot{u}_2 + \ddot{u}_5 \right) \cos \ddot{u}_1 + 2K_f \left( \cos \ddot{u}_1 \right) \]

\[ - \sin \ddot{u}_1 \left( \ddot{u}_1 - \frac{\dot{\chi}_5 + \dot{\chi}_6}{\ddot{\chi}_4} \right). \]

\[ b_3 = \frac{T}{l_i} \left( l_i \sin \ddot{u}_1(\ddot{u}_1 - \ddot{u}_5) + l_i \cos \ddot{u}_1(\ddot{u}_5 + \ddot{u}_6) \right) \]

\[ + 2K_f l_i \left( \cos \ddot{u}_1 - \sin \ddot{u}_1(\ddot{u}_1 - \frac{\dot{\chi}_5 + \ddot{u}_6}{\ddot{\chi}_4}) \right), \]

\[ b_4 = -\frac{T}{l_i} \left( l_i \cos \ddot{u}_1 - l_i \sin \ddot{u}_1 \right), \]

\[ b_5 = \frac{T}{l_i} \left( l_i \cos \ddot{u}_1 + l_i \sin \ddot{u}_1 \right). \]

### 3.2 In-wheel motor model

In this paper, it is assumed that the brushless direct current (BLDC) IWMs are used as the actuators of the EV. The dynamic models of BLDC-IWMs in both driving and braking cases can be described as follows:

\[ J_{\omega_w} \omega_w = M_i - F_i R_{off}, \quad i = 1, 2, 3, 4 \tag{20} \]

where \( J_{\omega_w} \) is the combined rotational inertia of the wheel and IWM, \( \omega_w \) is the yaw rate of wheel, \( M_i \) is the drive torque and \( R_{off} \) is the effective radius of the tire.

The efficiency of adopted DC motors in the driving and braking states are obtained by IWMs EV tests that are conducted on a twin-roll chassis dynamometer (shown in Figure 5). Note that in the IWMs EV tests, the in-wheel motor torque values at different speeds and torque control signals were measured by a torque sensor equipped on the chassis dynamometer, and the motor control signals were changed from 1.5 V to 4.5 V with a 0.15-V step at different motor speeds. A dSPACE MicroAutoBox was used to control and record all the EGV and chassis dynamometer signals in real-time. Given the limited space available, the detailed test process is omitted in this paper, but the similar test method can be seen in (Wang et al., 2011). Based on the test data, the efficiency maps are plotted in Figure 6. In addition, to introduce the motor efficiency into the efficiency management control, similar to (Chen and Wang, 2014), the polynomial fitting
method is used to gain the change in motor efficiency, and the polynomial function can be written as follows:

\[ \eta_D(M_t) = a_0 M_t^5 + a_1 M_t^4 + a_2 M_t^3 + a_3 M_t^2 + a_4 M_t + a_5 \]

\[ \eta_B(M'_t) = b_0 M'_t^3 + b_1 M'_t^2 + b_2 M'_t + b_3 \]  

(21)

where \( \eta_D(M_t) \) and \( \eta_B(M'_t) \) are, respectively, the driving and braking efficiencies of one IWM, and \( M_t \) and \( M'_t \), respectively, represent the driving torque and regenerative braking torque of wheel.

4. The establishment of controller

In light of the obtained reference trajectory and vehicle model, in this section, a novel autonomous lane change control system that can ensure precise dynamic tracking control and optimal energy consumption is proposed. The structure of the control system is shown in Figure 7.

Note that different from the previous control allocation (CA) researches wherein the sliding mode control (SMC) is adopted to gain the virtual control laws (Song et al., 2015), in the dynamic control level of this controller, MPC method is used to acquire the real control signals to resolve the chattering phenomena and the problem of control inputs under restraints in traditional SMC-based CA studies.

4.1 Planning control level

Based on reference trajectory in Equations (7) and (10) and \( t^* \) in Subsection 2.2, the reference states of vehicle in the next \( N_p \) times can be described as

\[
X_{r,N_p} = \left[ \chi_r(t_0 + kT, t^*_j), L, \chi_r(t_0 + (k+N_p)T, t^*_j) \right]
\]

(22)

where \( \chi_r = [X_r, Y_r, \theta_r, v_{x,r}, v_{y,r}, \alpha_r]^T \).

Within equation (22), the reference yaw angle, longitudinal and lateral velocities can be expressed as

\[
v_{x,r}(t_0 + (k + i)T, t^*_j) = v_{x,0} + \int_{t_0}^{t_0 + (k + i)T} \dot{a}_x(t, t^*_j) \, dt
\]

\[
v_{y,r}(t_0 + (k + i)T, t^*_j) = 0^{1 \times N_p}
\]

Figure 7 Structure of the proposed control system
\[ \theta(t_0 + (k + i)T, t_f) = \arctan \left( \frac{v_y(t_0 + (k + i)T, t_f)}{v_x(t_0 + (k + i)T, t_f)} \right) \]  

(23)

where \( \mathbf{0}^{i \times N_p} \) represents the zero vector that includes \( N_p \) elements, and \( i = 1, \ldots, N_p \).

Remark 1: In stability control of vehicle steering, the slip angle of vehicle is expected to be zero. Because vehicle slip angle equals to the specific value of lateral velocity to longitudinal speed, the reference lateral speed \( v_y \) is zero in equation (23).

4.2 Dynamic control level

Based on the reference states generated by planning controller and the LTV system of vehicle (18), the cost function in the finite horizon optimal control problem can be expressed as

\[ J_1 = \sum_{i=1}^{N_c} \| x_{ik} - X_{r_{ik}} \|^2 + \sum_{i=1}^{N_c} \| \delta u_{ik} \|^2 \]  

(24)

where \( Q_1 \in \mathbb{R}^{6 \times 6} \) and \( Q_2 \in \mathbb{R}^{5 \times 5} \) are definite positive matrices and \( Z_p \) is control horizons. At each time step \( T \), the following finite horizon optimal control problem is solved:

\[ \min_{\tilde{x}} J_1 \]  

s.t.  
\[ \tilde{x}_{t+1} = A_k \tilde{x}_k + B_k \tilde{u}_k + d_k \]  
\[ d_k = \tilde{x}_k - A_k \tilde{x}_k - B_k \tilde{u}_k \]  
\[ \tilde{u}_k = u_{k-1} - \tilde{x}_k = \lambda_k \]  
\[ u_{k|k} = \hat{u}(k - 1) + \delta \tilde{u}_{ik}, s = 0, \ldots, N_c - 1 \]  
\[ \delta \tilde{u}_{ik} = 0, s = t + Z_p, \ldots, t + N_p \]  
\[ u_{\min} \leq u_{ik} \leq u_{\max} \]  
\[ \delta u_{\min} \leq \delta \tilde{u}_{ik} \leq \delta u_{\max} \]  
\[ \alpha_{\min} \leq \alpha_{ik} \leq \alpha_{\max} \]  

(26)

where \( \Xi = [\delta \tilde{u}_{ik}, \ldots, \delta \tilde{u}_{i+N_c-1|k}]^T \).

The optimization problem (25) can be modified into a quadratic program (QP). The sequence of optimal input deviations computed at time \( k \) by solving (25) for the current states \( \chi(k) \) is denoted by \( \Xi^* \). Then, the first sample of \( \Xi^* \) is used to compute the optimal control action and the resulting state feedback control law is

\[ u_k = u_{k-1} + \delta \tilde{u}_{k|k} \]  

(27)

At the next time step \( k + 1 \), the optimization problem (28) is solved over a shifted horizon based on the new measurements of the state \( \chi(k + 1) \) and based on an updated linear model in equations (18)-(19) computed by linearizing the nonlinear vehicle model.

\[ V_{\text{inp}}(k) = \begin{bmatrix} V_1(k) \\ V_2(k) \\ V_3(k) \end{bmatrix} = \begin{bmatrix} \frac{1}{2} (F_{k_1} + F_{k_2}) \cos \delta_k + F_{x_3} + F_{x_3} \\ \frac{1}{2} (F_{k_1} + F_{k_2}) \sin \delta_k \\ \frac{1}{2} (F_{k_1} + F_{k_2}) \cos \delta_k - F_{x_2} + F_{x_3} + \frac{1}{2} (F_{k_1} + F_{k_2}) \sin \delta_k \end{bmatrix} \]  

(28)

Let \( u_k = [\tilde{\delta}_k, \tilde{F}_{x_1}, \tilde{F}_{x_2}, \tilde{F}_{x_3}, \tilde{F}_{x_4}] \), by substituting it into above equation, then the ideal virtual control forces \( V_{\text{inp}}(k) \) can be gained. Furthermore, based on the IWM model (20), \( V_{\text{inp}}(k) \) can be re-expressed as

\[ V_{\text{inp}}(k) = B_E \Lambda + B_I w \]  

(29)

where

\[ B_E = \begin{bmatrix} \cos \delta_k & \cos \delta_k \\ \sin \delta_k & \sin \delta_k \end{bmatrix} \]  

\[ B_I = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \]  

\[ \Lambda = [M_{t_1}, M_{t_2}, M_{t_3}, M_{t_4}]^T, \]  

\[ w = [\omega_{\omega_1}, \omega_{\omega_2}, \omega_{\omega_3}, \omega_{\omega_4}]^T. \]  

where the wheel angular acceleration \( \omega_{\omega_i} \) can be estimated through Kalman filters, just as was done in (Chen and Wang, 2012).

Based on the virtual control expression (29), the EECA design is described to solve the following nonlinear optimization problem:

\[ \min J_2 = \| B_P [\Lambda^T \Lambda^T]^T + B_I w - V_{\text{inp}}(k) \|_2 + \sigma P_c \]  

s.t.  
\[ \Lambda_{\min} \leq \Lambda \leq \Lambda_{\max} \]  
\[ \Lambda_{\min} \leq \Lambda \leq \Lambda_{\max} \]  
\[ \Lambda_{ij} = 0, i = 1, 2, 3 \]  

(30)

where \( \Lambda' = [M_{t_1}, M_{t_2}, M_{t_3}, M_{t_4}]^T \) stand for the regenerative brake torque signals, \( B_P = [B_E, B_E], Q_3 \) and \( \sigma \) are the positive weighting factors.
Within equation (30), $P_c$ is the total power consumption of in-wheel motors for the driving and regenerative braking modes and can be formulated as

$$P_c = \sum_{i=1}^{4} P_{oh}(M_0) - \sum_{i=1}^{4} P_{hi}(M_0) \eta_{Dh}(M_0)$$

where $P_{oh}$ and $P_{hi}$ is the energy consumption of IWMs in the driving model and regenerative braking mode, respectively. The corresponding electric efficiencies are indicated by $\eta_{Dh}$ and $\eta_{Dh}$, which can be obtained by using equation (21).

What is noteworthy is that $f_2$ is nonlinear and nonconvex optimization problem. To resolve this problem, the KKT-based method, which can transfer the nonlinear/nonconvex optimization problem into an algebraic eigenvalue problem and improve the computational speed, is applied in this paper. Because the focus of this paper is not on the optimization solution, the relevant resolving approach, which can be found in (Chen and Wang, 2012), is omitted.

5. Simulation and results

To verify the capability of the proposed control system, simulation analyses is carried out. The simulations are implemented based on the CarSim-Simulink platform with a high-fidelity and full-vehicle model. The simulation parameters are listed in Table I.

The simulation results are shown in Figures 8-17. In those figures, “Dynamic” means that only dynamic tracking control is involved, “D-EECA” signifies the controller proposed in Section 4, “$E_{sv}$”, “$E_{sy}$”, “$E_{sa}$”, “$E_{sw}$” are the absolute tracking errors of actual output of Carsim to the references. And the root-mean-square-errors (RMSE) of the vehicle states are collected in Table II. From Figures 8-11 and Table II, one can see that both “Dynamic” and “D-EECA” controllers can track the references accurately. Meanwhile, when searching the optimal lane time $t_0$ by introducing the vehicle dynamic stability boundaries into the path planning, the variations of yaw rate, lateral and longitudinal accelerations are limited (Figures 11, 16 and 17), and the homoluggous optimal lane time $t_{0}^{*}$, maximal lateral acceleration $|a_{y,\text{max}}|$ and maximal yaw rate $|\omega_{\text{max}}|$ are equal to 2.9 s, 0.2256 g and 4.36 deg/s, respectively.

To control energy efficiency by redistributing the torques of the four wheels (Figures 14 and 15), the power consumption in “D-EECA” should be obviously less than that in “Dynamic” as shown in Figure 12. The total energy consumption in “D-EECA” and “Dynamic” are 1.1646e + 3 kJ and 1.217e + 3 kJ during simulation (Table III). The energy is reduced by 4.3 per cent in “D-EECA”, compared with “Dynamic”. In some cases, “D-EECA” controller can realize accurate dynamic tracking control and reduce energy consumption, the proposed control method is valid.

It also should be noticed that the torques of the wheel in “D-EECA” and “Dynamic” are approximated in the first half of lane change time. This phenomenon is caused because of the small weight $\sigma$. Because the dynamics performance is the first thing that must be satisfied for an autonomous vehicle, a small $\sigma$ can realize the fact that the energy consumption can be reduced effectively in the case of high tracking accuracy. To further decrease energy consumption, the new EECA method and a more complete and accurate model of energy loss may be available and will be researched in the future.

### Table I Simulation parameters

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Value</th>
<th>Symbol</th>
<th>Value</th>
<th>Symbol</th>
<th>Value</th>
<th>Symbol</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$m_v$</td>
<td>1795 kg</td>
<td>$K_r$</td>
<td>55000 N/rad</td>
<td>$a_0$</td>
<td>-3.77e-9</td>
<td>$b_0$</td>
<td>2.49e-6</td>
</tr>
<tr>
<td>$I_f$</td>
<td>4175 kg.m$^2$</td>
<td>$C_o$</td>
<td>0.7446</td>
<td>$a_1$</td>
<td>7.09e-7</td>
<td>$b_1$</td>
<td>-4.41e-4</td>
</tr>
<tr>
<td>$I_t$</td>
<td>1.5 m</td>
<td>$C_r$</td>
<td>0.034</td>
<td>$a_2$</td>
<td>-3.75e-5</td>
<td>$b_2$</td>
<td>2.67e-2</td>
</tr>
<tr>
<td>$I_y$</td>
<td>1.4 m</td>
<td>$\eta$</td>
<td>2 m/s$^2$</td>
<td>$a_3$</td>
<td>-1.69e-4</td>
<td>$b_3$</td>
<td>1.42e-2</td>
</tr>
<tr>
<td>$I_e$</td>
<td>1.6 m</td>
<td>$R_{eff}$</td>
<td>0.353 m</td>
<td>$a_4$</td>
<td>5.22e-2</td>
<td>$l_{\mu}$</td>
<td>0.9 kg.m$^2$</td>
</tr>
<tr>
<td>$k_f$</td>
<td>55,000 N/rad</td>
<td>$\sigma$</td>
<td>0</td>
<td>$a_5$</td>
<td>-3.35e-2</td>
<td>$N_0$</td>
<td>6</td>
</tr>
<tr>
<td>$M_{h,\text{min}}$</td>
<td>-45 N.m</td>
<td>$M_{\mu,\text{min}}$</td>
<td>-45 N.m</td>
<td>$\delta$</td>
<td>1.2</td>
<td>$\delta \mu_{1,\text{min}}$</td>
<td>-1 deg</td>
</tr>
<tr>
<td>$M_{h,\text{max}}$</td>
<td>45 N.m</td>
<td>$M_{\mu,\text{max}}$</td>
<td>45 N.m</td>
<td>$\delta_{\mu,\text{min}}$</td>
<td>-4 deg</td>
<td>$\delta \mu_{1,\text{max}}$</td>
<td>1 deg</td>
</tr>
<tr>
<td>$\delta h_{\text{min}}$</td>
<td>-150 N</td>
<td>$\delta \mu_{1,\text{max}}$</td>
<td>150 N</td>
<td>$\delta_{\mu,\text{max}}$</td>
<td>4 deg</td>
<td>$N_c$</td>
<td>4</td>
</tr>
<tr>
<td>$T$</td>
<td>0.05 s</td>
<td>$Y_0$</td>
<td>0 m</td>
<td>$Y_1$</td>
<td>3.5 m</td>
<td>$X_0$</td>
<td>0 m</td>
</tr>
<tr>
<td>$t_0$</td>
<td>2 s</td>
<td>$v_{x_0}$</td>
<td>120 km/h</td>
<td>$\mu$</td>
<td>0.85</td>
<td>$\varepsilon$</td>
<td>0.85</td>
</tr>
</tbody>
</table>
6. Conclusion

In this paper, a novel lane change control system for A-IWM EV that can enhance vehicle stability and reduce energy expenditure is proposed. The whole control system consists of stable trajectory planning level, high dynamic control level and low EECA level. In the planning level, to ensure the feasibility...
of the generated reference path, vehicle dynamics is considered and stability boundaries are established. The MPC and KKT-based algorithm are adopted to guarantee the precision of dynamic tracking and resolve the nonlinear optimization problem in the high and low levels, respectively. Simulation results on an autonomous vehicle with in-wheel motors based on a full-vehicle model in CarSim show the effectiveness of the proposed control system.

References


Hult, R. and Tabar, R.S. (2013), Path Planning for Highly Automated Vehicles, Chalmers University of Technology, Gothenburg.


Corresponding author
Guodong Yin can be contacted at: ygd@seu.edu.cn

For instructions on how to order reprints of this article, please visit our website: www.emeraldgrouppublishing.com/licensing/reprints.htm
Or contact us for further details: permissions@emeraldinsight.com
Applications of intelligent computing in vehicular networks

Daxin Tian
Beihang University, Beijing, China
Weiqiang Gong
Nanjing LES Information Technology Co., Ltd, Nanjing, China
Wenhao Liu, Xuting Duan, Yukai Zhu and Chao Liu
Beihang University, Beijing, China, and
Xin Li
Department of Computer Science and Technology, Tsinghua University, Beijing, China

Abstract
Purpose – This paper aims to introduce vehicular network platform, routing and broadcasting methods and vehicular positioning enhancement technology, which are three aspects of the applications of intelligent computing in vehicular networks. From this paper, the role of intelligent algorithm in the field of transportation and the vehicular networks can be understood.

Design/methodology/approach – In this paper, the authors introduce three different methods in three layers of vehicle networking, which are data cleaning based on machine learning, routing algorithm based on epidemic model and cooperative localization algorithm based on the connect vehicles.

Findings – In Section 2, a novel classification-based framework is proposed to efficiently assess the data quality and screen out the abnormal vehicles in database. In Section 3, the authors can find when traffic conditions varied from free flow to congestion, the number of message copies increased dramatically and the reachability also improved. The error of vehicle positioning is reduced by 35.39% based on the CV-IMM-EKF in Section 4. Finally, it can be concluded that the intelligent computing in the vehicle network system is effective, and it will improve the development of the car networking system.

Originality/value – This paper reviews the research of intelligent algorithms in three related areas of vehicle networking. In the field of vehicle networking, these research results are conducive to promoting data processing and algorithm optimization, and it may lay the foundation for the new methods.

Keywords Intelligent computing, Vehicular ad hoc networks

Paper type Research paper

1. Introduction

The advancement of computer technology in recent years has allowed researchers to develop efficient optimization techniques for solving large-scale problems in various fields and helped practitioners to incorporate some of these techniques into their planning activities through specific integrated decision support systems (Jin and Meng, 2010; Chiong and Weise, 2011). Despite this, many real-world problems still cannot be solved within reasonable computational time using exact algorithms because of the complexities associated with the large number of decision variables involved and the extent of the constraints imposed (Chetty et al., 2010). As a consequence, non-exact methods such as custom designed heuristics, search algorithms and meta-heuristic approaches have become increasingly popular due to algorithmic efficiency, ease of implementation, flexibility for adaption and quality of solutions (Chiong et al., 2012). Among them, intelligent computing approaches inspired by principles of nature such as evolutionary algorithms, swarm intelligence algorithms, artificial neural networks and fuzzy logic have emerged as a rapidly growing research area. These approaches have been applied to various problems in different fields (Çatay et al., 2013; Jin and Hammer, 2014).

© Daxin Tian, Weiqiang Gong, Wenhao Liu, Xuting Duan, Yukai Zhu, Chao Liu and Xin Li. Published in Journal of Intelligent and Connected Vehicles. Published by Emerald Publishing Limited. This article is published under the Creative Commons Attribution (CC BY 4.0) licence. Anyone may reproduce, distribute, translate and create derivative works of this article (for both commercial and non-commercial purposes), subject to full attribution to the original publication and authors. The full terms of this licence may be seen at http://creativecommons.org/licenses/by/4.0/legalcode

This research was supported in part by the National Natural Science Foundation of China under Grant Nos. 61672082, U1564212.

Received 11 January 2018
Revised 9 April 2018
12 April 2018
Accepted 13 April 2018

The current issue and full text archive of this journal is available on Emerald Insight at: www.emeraldinsight.com/2399-9802.htm
With the rapid development of the key technology of wireless networks, intelligent computing and other intelligent technologies, networking industry makes the vehicular networks enter a new stage of research, research on cross car networking involves many subjects, including the automotive industry, transportation, information and communication technology (Liu et al., 2010). The promising applications of vehicular ad hoc networks (VANET) are collision warning, location tracking, emergency brake notification, signal violation notification, parking lot availability, weather information and major information services on the road (Saravanan and Arunkumar, 2014; Saravanan and Arunkumar, 2016). At the same time, some of applications such as routing algorithms (Ratnasamy et al., 2002; Golden et al., 2010), data mining (Kargupta et al., 2004; Kargupta, 2010), positioning enhancement (Golestan et al., 2015; Du et al., 2015) and others have been widely used in the vehicular networks on the basis of intelligent computing.

In Section 2, this paper discusses the methods vehicular network platform and data quality assessment, and it also compares performance metrics. In Section 3, a routing and broadcasting methods in vehicular networks is drawn into this paper, and the effect has been improved obviously than before. We propose a method of vehicle positioning enhancement named CV-IMM-EKF in Section 4. Finally, this paper will sum up the three sections in Section 5.

2. Methods vehicular network platform and data quality assessment

Focus on our topic on internet of vehicle, vehicular network data need to be of high quality to reflect the operation pattern and running status of the vehicle, including vehicle fuel consumptions and traveling route (Tian et al., 2015). However, due to abnormalities of sensors and the unreliability of wireless communication environment, some data of vehicular sensors are inaccurate or incomplete. Hence, the quality of vehicular data is difficult to be guaranteed, which has strong relation to the validity and accuracy of data mining results (Hou et al., 2016; Chen et al., 2017). Therefore, quality control and analysis on these data is indispensable, to assessment the quality level of the data or to detect the sensor error, filter out the vehicle with inferior data quality.

In this section, we give a brief introduction of vehicular network big data platform and its data quality problem, including the database and outlier situation. Then we introduced a classification framework based on data fusion and machine learning approach to assess the quality of vehicular data.

2.1 Vehicular network data platform and data quality problem

A brief system framework of integrated vehicular network contains cellular networks and vehicle-to-vehicle (V2V) communication (Biswas and Giaffreda, 2014). The topic and data set introduced in this paper is from a vehicular network platform, which contain the vehicle running state data from thousands of vehicles. These data consist of global positioning system (GPS) coordinates, speeds, driving direction angle, fuel level value and so on. Figure 1 shows the real-time big data platform of our vehicular network and the data types of database. However, there are several types of abnormal situation during the process of fuel level data acquisition.

It is well-known that there should be positive correlation between the correct fuel consumption and the speed data, which is the fuel level should decline when the vehicle is running (Chen and Zhao, 2014). Contrary to the common sense, the fuel level of some vehicles remains unchanged while the vehicle is moving over an extended period. In some other conditions, same values appear frequently, while the uploaded fuel-level data are float numbers which should always fluctuate in a lesser range. We suppose that the possible causes of these abnormal situations are in the following items:

- The sensors of vehicles may have too much noise or their errors are too large.
- The terminal equipment may upload or read wrong values or error codes in communication with the sensors.
- The information processing system on the server side may read and update wrong or null values into the database.

2.2 Data quality analysis for vehicular network data

Data quality problem is a further topic of anomaly detection, which is generally statistical approach (Chen et al., 2017). It could make use of the historical distribution of data. As a result, the classification approach is much stronger connected to their application area; this is often achieved by building rules. Multi-sensor data fusion generally provides significant advantages in data mining procession. In addition to the statistical advantage gained by obtaining an improved estimate of a physical phenomenon through redundant observations. The use of multiple types of sensors may increase the accuracy with which a phenomenon can be observed.

In this study, we tried to find the vehicles with large amounts of outliers which are defined as typical samples with abnormal data through observing the fuel level-time curve. The typical fuel level-time curves of two sample sets are as Figure 2. From the comparison of these figures, Figure 2(a) shows a typical fuel level curve of the sample with relatively credible data, while Figure 2(b) and (c) shows the opposite situation.

Figure 1 Vehicular network big data platform

![Figure 1](image-url)
2.2.1 Features extraction based on data fusion and discrete wavelet transform

Data fusion with vehicular sensor data. To assess the data quality of the speed data, we compute the travel speed according to the GPS coordinates and compare it with the speed which is collected from running cars on real time (Smith et al., 2011). The travel speeds could be computed by latitudes $\lambda$ and longitudes $\lambda$, and we compute the relative error ratio of the uploaded speeds $v_u$ with the calculated travel speeds $v_t$. We use these calculated travel speeds to evaluate the accuracy of the uploaded speeds and the GPS coordinates, and found that these data are credible to some extent. In equation (1), $t$ is the timestamp and $i$ means the $i$th data point in the series:

$$v_i = 111.199 \left[ (\varphi_i - \varphi_{i+1})^2 + (\lambda_i - \lambda_{i+1})^2 \cos \left( \frac{\varphi_i + \varphi_{i+1}}{2} \right) \right]$$

(1)

Classification algorithm needs the input of the feature attributes before training. There is a relationship between fuel level and speed value; hence, we use standard deviation (STD) of these data and correlation coefficient between these multi-dimension data as features for classifier.

In the preprocessing procedure of many types of sensor data, the original series data may contain variety of noise and anomaly pattern. With the decomposition with discrete wavelet transform (Rogers and Nicewander, 1988), the set of approximation coefficients $A_k$ retained the main profile of the original data, which is a smooth serious with less noise. On the other hand, the set of detail coefficients $D_k$ represented the noise and slope which contains anomaly pattern in different scaling function. The signal is decomposed simultaneously using a high-pass filter $H$. The outputs give the detail coefficients from the high-pass filter and approximation coefficients from the low-pass. However, as half the frequencies of the signal have now been removed, half the samples can be discarded:

$$y_H[n] = [n](x * G) = \sum_{k=-\infty}^{\infty} x[k]H[2n - k]$$

(2)

$$y_L[n] = [n](x * G) = \sum_{k=-\infty}^{\infty} x[k]G[2n - k]$$

(3)

2.2.2 Data quality assessment with machine learning algorithm

Bayesian network is a useful and powerful algorithm for uncertainty reasoning. For a data set $S$ with attribute set $X = \{X_1, \ldots, X_n\}$ and class label $Y = \{y_1, \ldots, y_n\}$, $t = \{x_1, \ldots, x_n\}$ is a sample in $S$, the classifier predicts the class label of $t$ by calculating probability. To overcome the limitation of conditional independent assumption, functional dependency is an important part of relational database, which represents the constraint relations among different attributes. Some existing research studies have found some similarities between relational database and probabilistic inference. Then we have:

$$\arg\sum_{\max} P(y|x_1, \ldots, x_n)$$

$$= \arg\sum_{\max} P(y) \cdots P(x_n|y)$$

(4)

In semi-supervised learning, as the joint likelihood of the labeled and unlabeled data is not in closed form, usual solutions to this would be to use expectation maximization (Chen and Zhao, 2014; Bishop, 2006). The algorithm initials model parameters from limited amount of labeled data and uses these to get probabilistic labels for each unlabeled sample in E-step. M-step gets the parameters using these labels for unlabeled instances. A regulating variable $\lambda$ is used in the proposed method. This parameter moderates the unlabeled data by reduce the learning rate $\eta$ and the weight of the unlabeled samples in step $M$. $\theta_i$ is current estimator of the model parameters. $f_i^2$ gives the count of feature in instance $x$. Conditional estimates of $P(y|x)$ from labeled data improve the accuracy of Bayesian approach:

$$\text{Initial} : \theta_L = \arg\max_{x \in B_0} \log p_h(x, y)$$

$$\text{Estep} : \forall x \in D_u \cup D_c \text{compute } \theta_i(y|x)$$

$$\text{Mstep} : \theta_{t+1} = \arg\max_{x \in D_u \cup D_c} \log p_h(x, y)$$

$$N_{y|x} = \sum_{x \in D_u} f_i^2 P_{\theta_i}(y|x)$$

(5a)
In the vehicular network data quality problem, the quality states of this deployment are adopted from the flagging scheme. There are flags associated with particular processing tasks. The quality flags are designed to indicate the level of uncertainty involved in the sensor reading. Sometimes, sensor readings associated with abnormal events may be incorrectly labeled. These labels belong to four different classes of the classification problem. The data that we used in the experiments were labeled into four classes:
1. good data;
2. probably good data;
3. bad data which are potentially correctable; and
4. bad data.

Figure 3 shows a labeled data series after separated into many pieces of length; thus, a long series of data can be evaluated for each period.

The blue lines denoted the probably good or bad curves, and the yellow one is series with bad flag.

This work presents our experiences for big data analytics based on a vehicular network big data testbed, in terms of sensors data management, multi-dimension data fusion and data quality assessment for the vehicular sensor data. In this section, we have investigated the problem of multidimensional analysis of vehicular network testbed data. Some statistical indicators are introduced and applied in data quality evaluation, and a novel classification-based framework is proposed to efficiently assess the data quality and screen out the abnormal vehicles in database. Our experiments on large real data sets show the feasibility and utility of proposed methods.

3. Routing and broadcasting methods in vehicular networks

With the development of wireless communication technologies and increasing demand for high-data-rate communication, the integration of the cellular network with other access networks has become a popular trend. Device-to-device (D2D) communication is an example of an integrated network technology that appears to be a vital component in next generation communication technologies (5G). The D2D technology allows for direct message delivery between terminals that are near each other to lighten the load of node base stations, improve spectral reuse and enhance system capacity. Therefore, mobile ad hoc networks (MANETs) are expected to be popularized in the future based on D2D communication (Mumtaz et al., 2015). VANETs, as a special application of MANETs in transportation, have attracted increasing academic and industry attention in recent years. These networks can help to realize a variety of vehicular safety and intelligent applications, such as cooperative collision avoidance, road obstacle warning, self-adapted cruise control system, multimedia streaming and other telematics services (Panichpapiboon and Pattara-Atikom, 2012). Nevertheless, VANETs have a highly dynamic characteristic due to the fast movement of vehicles, which results in a number of great challenges such as dynamic network topology, frequent disconnections, changing node density, and a lack of centralized control. For example, the urban road net is a complex network system. The vehicle’s movement appears as a frequent entrance and departure to the area networks, which results in a lack of instantaneous end-to-end paths; hence, the VANET is considered as a type of delay and disruption-tolerant networks (DTNs). As a common phenomenon in large cities, traffic congestion causes the vehicular network to become very dense, which has an influence on the subsequent data dissemination. In addition, traffic scenes or phenomena such as freeways, platoons or cluster driving and tide traffic may lead to many specific network changing modes. The integrated vehicular network is shown in Figure 4, including internet, cellular network and D2D networks.

As the high mobility of vehicles leads to increased uncertainty in the opportunistic transmission between vehicles, many of the traditional MANET routing protocols and message dissemination models that depend on the maintenance of an end-to-end transmission path are no longer suitable for VANETs. Therefore, the development of a feasible and efficient broadcasting mechanism for message dissemination is necessary. The fundamental ability of any kind of communication network is the transportation or routing of data from a source to a destination. Communication paths determine the communication quality and transmission delay to a great extent. Routing is the process to select the best communication paths in a network. There has been much research concerning the optimal routing methods in ad hoc networks. The simplest approach may be to let the source or a moving relay node carry or retransmit the message to the destination. A suitable method to perform routing in delay tolerant networks is epidemic routing through use of the “store-carry-forward” mechanism. This mechanism is referred to as an epidemic broadcast, as an analogy of the transmitting behavior of viruses.

In epidemic broadcasts, messages are spread among nodes. When a node with a message labeled as an infected node contacts other nodes without a message, the infected node sends a copy of the message to those susceptible nodes. For more detail, a mobile node stores a message in its local memory after receiving this message from a source and carries the message along the movement path until the next hop moves into the communication range. Once it encounters another node, this infected node duplicates and forwards the copy to the new node. This “store-carry-forward” process will be repeated until the message is forwarded to the destination. In
VANETs, every vehicle can be such a mobile infected node, and thus the information packet will be able to propagate from the source to the other distant vehicles. In a dense road scene, a continuous communication path without cellular network can be found, this message dissemination process is shown in Figure 5.

3.1 Epidemic broadcasting simulation based on a realistic traffic scene
Simulation is the predominant tool used in research related to VANET (Liu et al., 2009). We will discuss a simulation that was based on real-world road traffic conditions on the East Fourth Ring Road in Beijing (Figure 6). We performed a field survey and collected historical data from the East Fourth Ring Road to obtain reliable simulation parameters such as speed, density and congestion distribution. In Figure 6(a), the real-time traffic is colored in green, indicating that free-flow traffic conditions existed, where vehicles could travel at high speeds and at low densities. Under these conditions, drivers could arrive at their destinations within the prediction time. In Figure 6(b), the selected segment was mainly yellow, which indicated that traffic was moving at a relatively slower speed (approximately 45 km/h) and at densities on the middle level. In Figure 6(c), traffic congestion conditions were apparent, where both yellow and red sections were present on the selected segment. The red and yellow colors were distributed in two different directions on the road, where one direction was mainly yellow and the other direction was mainly red, indicating that vehicles were at low speeds and high densities.

3.2 Simulation setting
We simulated the data packages broadcasting in three scenarios (i.e. the free-flow traffic, heavy traffic and congestion scenarios), according to the traffic state of a road section of the East Fourth Ring Road of Beijing. To simplify the simulation, we simulated the traffic in a single lane for both directions. The length of the section was 5 kilometers. Two devices were located at each end of the section: the source node, which could generate data packages and the destination node, which would receive the message. In each scenario, the simulation interval was 1 s and the time was 150 s. The source node generated data packages every 5 s. Two vehicles could communicate only

---

**Notes:**
(a) Free-flow; (b) heavy traffic; (c) congestion
when their distance was less than 150 meters. When two vehicles were within communication range, one vehicle copied its particular data packages and sent them to the other. If two vehicles had the same data packages, there were no packages transmitted between them. The lifetime of each copy was 60 s. In the free-flow, heavy traffic and congestion scenarios, we set the average speed at 70, 50 and 25 kilometers per hour, respectively, and the number of vehicles in a lane was set to be 50, 250 and 400, respectively.

3.3 Results and analysis
We used the aforementioned simulation setting to simulate the process of data package broadcasting. In each scenario, we performed the simulation ten times within the same simulation setting. Figure 7 shows the results of the number of copies versus the time variation in the three scenarios. Based on the results, the destination node in the congestion traffic condition could receive more copies than in the free-flow condition. This result was as expected because the vehicles density is high in the congestion condition; thus, the copies can be more easily relayed than in the non-congestion condition.

We also observed the number of relays for a copy before it arrived at the destination node. Figure 8 shows the results of the number of copies versus the number of relays in the three scenarios. Intuitively, more relays were required to send the message when the vehicle density was higher. The average number of relays in the three scenarios (free-flow, heavy and congestion) were 39.8, 47.5 and 49.7, respectively, which was considered to be reasonable. In addition, we observed the time taken by a copy to arrive at the destination node, as shown in Figure 6. The average times obtained from the three scenarios (free-flow, heavy and congestion) were 9.3, 11.3 and 9.5 s, respectively. The reasons why the average times in the free-flow and congestion scenarios were less than that in the heavy traffic scenarios were attributed to the high vehicle speed in the free-flow scenario and the high vehicle density in the congestion scenario, both of which benefited the message transmission. In addition, we observed that many copies could arrive at the destination once they were generated. In the free-flow and heavy traffic scenarios, approximately 70 per cent of the copies could arrive at the destination node within 10 s, whereas approximately 80 per cent could arrived in the congestion scenario. Nevertheless, some copied packets were unable to reach the destinations when the density of vehicles in the free-flow scenario was too low.

In the case study, we performed a broadcasting simulation based on real-road traffic conditions from the East Fourth Ring Road in Beijing, and observed that different vehicle densities can dramatically affect message transmission. Moreover, experiment results showed that when traffic conditions varied from free-flow to congestion, the number of message copies increased dramatically and the reachability was also improved. Nevertheless, the high network density may introduce other issues, such as the broadcast storm problem.

4. Vehicular positioning enhancement with connected vehicle
The availability of high-accuracy location-awareness is essential for a diverse set of vehicular applications including intelligent transportation systems, location-based services, navigation and a couple of emerging cooperative vehicle-infrastructure systems (Alam and Dempster, 2013). However, it still faces a big challenge in the areas with inconsistent availability of satellite networks, especially in dense urban areas where the standalone global navigation satellite systems (e.g. GPS).

To tackle the aforementioned problems, a new class of cooperative positioning (CP) methods that relies on V2V communications and data fusion filtering (Sayed et al., 2005; Win et al., 2011; Alam et al., 2012) has been presented in recent years, which can further improve the accuracy of positioning. Actually, such concern raised in CP is the reliability of the localization approaches in heavy multipath and NLOS scenarios, which is similar to that in indoor environment (Sayed et al., 2005; Gu et al., 2009). An exploration into CP from theoretical foundations to specific applications, including advanced technologies and cooperative algorithms was presented in Win et al. (2011). Several CP techniques have
been proposed to achieve lane-level vehicular positioning using DSRC signals broadcasted by onboard both units (OBUs) and RSUs (Alam et al., 2012; Dao et al., 2007).

Because of the low speed of vehicles, the DFS is too difficult to be extracted from noise, and thus for DFS vehicular positioning methods, the STD of positioning error increases as the relative speed between the target vehicle (TV) and the other vehicles decreases. So we will investigate the method to overcome this problem. In this section, we will focus on the scenario that the neighbor vehicles travel in the opposite direction of the TV, for this case can provide obviously detectable Doppler effect.

4.1 Problem statement

The problem to be solved is to estimate the position of a TV moving on a 2D road, where there are many other neighbors around the TV. All of the vehicles are able to know their own state information, including position and velocity, provided by coarse GPS receiver, and they can know the neighbors’ state information via vehicular communications as well. Consequently, this case can be treated as a simple but practical CV scenario. A TV is considered as a research object for positioning enhancement based on CV, and a neighbor is considered as the vehicle who is within a certain communication range to the TV and travel in the opposite direction of the TV. Each vehicle is assumed to be with an offshore banking unit providing both the DSRC and the DFS measurements (Duan et al., 2015).

Considering the rth moving vehicle at time instant $k$ with a state vector $\theta_k^r = [p_{x,k}^r, p_{y,k}^r, v_{x,k}^r, v_{y,k}^r]^T$, $i = 1, \ldots, n_p$, where the $(p_{x,k}^r, p_{y,k}^r)$ and $(v_{x,k}^r, v_{y,k}^r)$ denote the rth vehicle’s position and velocity, respectively, $n_p$ is the total number of the vehicles driving on the road and $T$ is a transpose operator. The dynamic state can be modeled by the following system:

$$\theta_k^r = \Phi_{k-1}^r \theta_{k-1}^r + G_{k-1}^r (u_{k-1}^r + w_{k-1}^r) \quad (6)$$

where $\Phi_{k-1}^r$ is the state transition matrix, and $G_{k-1}^r$ is the noise distribution matrix. $u_{k-1}^r$ is the control vector and $w_{k-1}^r$ is zero-mean white Gaussian noise with covariance matrix $C_{w_{k-1}^r}$.

For the dynamic model presented by equation (6), the following observation model can be defined:

$$\psi_k = h(\theta_k) + \gamma_k(r_c) \quad (7)$$

where $h = [p_{x,k}^r, p_{y,k}^r, v_{x,k}^r, v_{y,k}^r, \omega_{x,k}^r, \ldots, \omega_{y,k}^r]^T$ is a nonlinear observation vector in terms of $\theta_k$. $\omega_{x,k}^r$ is the DFS of the received signal from the $j$th neighbor, $j = 1, \ldots, n_k$, $n_k < n_p$, and $n_k$ is the total number of the neighbors on the road. $\gamma_k(r_c)$ is the observation noise that can be used to describe the M types of observation errors by assuming a set of another covariance matrices. The transition among M types of the errors is generally modeled as a first-order M-state homogeneous Markov chain $r_c$, $c = 1, 2, \ldots, M$.

Specifically, assuming that the DFS measurements from the OBU can be modeled in a derivative form of the DSRC carrier frequency, $f_c$, as follows:

$$\omega_{x,k}^r = -\frac{f_c}{c} \nabla_i (d_{k}^i + \psi_{i}^r(r_c))$$

$$d_{k}^i = \sqrt{(p_{x,k} - p_{x,k}^i)^2 + (p_{y,k} - p_{y,k}^i)^2} \quad (8)$$

Notes: (a) Free-flow; (b) heavy traffic; (c) congestion
where $c$ is the speed of light, $d^i_k$ is the relative distance between
the TV and its neighbor $j$ and $\theta^i_k$ is the DFS observation noise
of neighbor $j$, where $(p^i_{x,k}, p^i_{y,k})$ and $(v^i_{x,k}, v^i_{y,k})$ is the position
and velocity vector of the neighbor $j$. To solve this nonlinear
observation function, with the first-order Taylor expansion of
(8) around an arbitrary state vector, $k$ can be transformed to a
fixed form of matrix, in which all of the components are
supposed to obtain from both the GPS and OBU. As a result,
the observation model of the TV can be reformulated as a linear
one:

$$
Z_k = H_k \theta_k + \gamma_k(r_c)
$$

(9)

with the observation transition matrix $H_k$.

Based on the aforementioned models from equation (6) and (9),
it is reasonable to assume that $\theta^i_k$ and $G_k$ are invariable at both
each time instant and vehicle. Therefore, the position estimation of the TV can be formulated as the problem of
linear filtering for M-state jump Markov systems and the model
can be simplified as:

$$
\theta^i_k = \phi \theta^i_k + G(u_{k-1} + \omega^i_{k-1})
$$

$$
Z_k = H_k \theta_k + \gamma_k(r_c)
$$

(10)

Because $H_k$ can be estimated by data fusion from both the GPS
and OBU at each time instant $k$, a CV-enhanced interacting
multiple model extended Kalman filter (IMM-EKF) can be deployed.

4.2 Connected vehicles-enhanced interacting multiple
model extended Kalman filtering (IMM-EKF)

4.2.1 Mixing probabilities and state estimates:

$$
\mu_{k+1|s,t} = \sigma_{ui} \mu_{k|s} c_t
$$

(11)

where $\mu_{k+1|s,t}$ is known as the mixing probability in the IMM
estimator, $\mu_{k|s}$ is the probability of the event that the $t$th motion
model is in effect at time step $k$, $s, t = 1, 2, \ldots, M$, correspond
to the $s, t$th mode of the Markov chain $r_c$ and $c_t$ is a
normalization constant:

$$
\theta^0_{k|k,t} = \sum_{i=1}^{M} \mu_{k+1|s,t} \theta^0_{k|k,t}
$$

$$
P^0_{k|k,t} = \sum_{i=1}^{M} \mu_{k+1|s,t} \left[ P^0_{k|k,t} + \left( \theta^0_{k|k,t} - \theta^0_{k|k,t} \right) \right]
$$

$$
\times \left( \theta^0_{k|k,t} - \theta^0_{k|k,t} \right)^T
$$

(12)

4.2.2 Mode update and prediction steps:

The CV-IMM-EKF advanced prediction is given by:

$$
\theta_{k|k-1,t} \approx \phi \theta_{k-1|k-1,t} = \phi \theta_{k-1|k-1,t}
$$

(13)

The State prediction error covariance matrix is as follows:

$$
P_{k|k-1,t} \approx \phi P_{k-1|k-1,t} \phi^T + C_{k-1}
$$

$$
= \phi P_{k-1|k-1,t} \phi^T + C_{k-1}
$$

(14)

From the previous data, the CV-IMM-EKF gain is given by:

$$
T_k = P_{k|k-1,t} \left( H_k \left( \theta_k(N) \right) P_{k|k-1,t} \left( H_k(N) \right) + R_k(r_c(N)) \right)^{-1}
$$

(15)

where $\theta_k(N)$ and $r_c(N)$ are functions of $N$ and can change the
dimension of the observation transition matrix $H_k$ and the
covariance matrix $R_k$, respectively.

The CV-IMM-EKF update steps are given by:

$$
\theta^0_{k+1|k,t} = \theta^0_{k|k-1,t} + T_k \left( Z_k - H_k \left( \theta_k(N) \right) \theta^0_{k|k-1,t} \right)
$$

$$
P^0_{k+1|k,t} = P^0_{k|k-1,t} - T_k \left( Z_k - H_k \left( \theta_k(N) \right) \right) \theta^0_{k|k-1,t}
$$

$$
+ R_k(r_c(N)) \right) T_k^T
$$

(16)

The CV-IMM-EKF prediction steps are given by:

$$
\theta^0_{k+1|k,t} = \phi \theta^0_{k|k,t} + G \mu_{k|s}
$$

$$
P^0_{k+1|k,t} = \phi \theta^0_{k|k,t} \phi^T + GCG^T
$$

(17)

4.2.3 Estimates combination:

$$
\theta_{k|k} = \sum_{i=1}^{M} \mu_{k|s,t} \theta_{k|k,t}
$$

$$
P_{k|k} = \sum_{i=1}^{M} \mu_{k|s,t} \left[ P_{k|k,t} + \left( \theta_{k|k,t} - \theta_{k|k} \right) \right]
$$

$$
\times \left( \theta_{k|k,t} - \theta_{k|k} \right)^T
$$

(18)

4.3 Numerical study

A basic set with $N$ neighbors for the TV can be formed through
Algorithm CV-IMM-EKF. Considering a section of urban roads, which is with a width of four lanes (each one is
3.5 m wide) and a length of 1 kilometer. It is assumed that the
traffic density of the road section is 20 vehicles/km and the
average speed of traffic is generated stochastically in
duration from 50 km/h to 60 km/h following a uniform
distribution.

The noise vector $\omega_{k-1} = [\sigma_{ax,k-1}, \sigma_{ay,k-1}]^T \sim N(0, C)$, with
covariance matrix $C = diag(\sigma_{ax}^2, \sigma_{ay}^2)$, where the elements
$\sigma_{ax,k-1} = \sqrt{0.99/2}$ and $\sigma_{ay,k-1} = \sqrt{0.01/2}$ are the
acceleration noises along the $x$ and $y$-axis, respectively, with
STD in $m/s^2$. The covariance matrix $R(r_c)$ of observation noise
$Y_{k}(r_c) \sim N(0, R(r_c))$ is described as a first-order Markov chain
switching between two models $R(r_1) = \text{diag} \left[ \alpha^2_{px}, \alpha^2_{py}, \alpha^2_{ux}, \alpha^2_{uy}, \alpha^2_{u1}(r_1), \ldots, \alpha^2_{uN}(r_1) \right]$ and $R(r_2) = \text{diag} \left[ \alpha^2_{px}, \alpha^2_{py}, \alpha^2_{ux}, \alpha^2_{uy}, \alpha^2_{u1}(r_2), \ldots, \alpha^2_{uN}(r_2) \right]$, of which the elements are with STDs in units of $m, m/s^2$ and $Hz$. The transition probability for this Markov chain is $\pi_{r_1} = \left[ 0.9 \ 0.1 \ 0.1 \ 0.9 \right]$, and their initial probability is $\mu_0 = [0.5 \ 0.5]$. According to the achievable performance discussed in Alam and Dempster (2013) and Tian et al. (2015), as the number of the neighbors is increasing, the performance enhancement can be less obvious and lead to more additionally computational burden.

Therefore, we set $N = 4$, which is a relatively conservative number of the neighbors for the basic set, which is mentioned in algorithm CV-IMM-EKF.

In the simulations, the sampling period and length are taken to be 0.2 and 100, respectively, and the communication range of the DSRC is 300 $m$. As the DFS measurements presented in Tian et al. (2015) and Schmidl and Cox (1997), the probability density function (PDF) of the DFS is approximately zero-mean asymmetric Gaussian with the left and right STDs of 100 and 120 $Hz$, when the vehicles travel at the speed of 60 $km/h$, broadcasting the DSRC packets with a frequency of 5.89 $GHz$ and a rate of 100 packets. It is worth noting that the PDF of the DFS remains a fairly consistent estimation from LOS to NLOS. Considering the noise of the DFS measurements as zero-mean Gaussian with two states of STDs: $\sigma_{aN}(r_1) = 100 Hz$ and $\sigma_{aN}(r_2) = 120 Hz$. Specifically, the state of the observation noise remains unchanged in $r_1$ between 0 and 6 $s$, and changes in the following 10 $s$ to $r_2$. Finally, the state changes back to $r_1$ for another 4 $s$. The position and velocity measured by GPS are assumed to be added noise with the variance $(\sigma_{px} = \sqrt{200/2m}, \sigma_{py} = \sqrt{200/2m})$ and $(\sigma_{ux} = \sqrt{15/2m}, \sigma_{uy} = \sqrt{15/2m})$, respectively.

To quantify the performance of the proposed approach, the root mean square error (RMSE) of vehicular positioning is calculated to assess the closeness of the estimated trajectory $(\hat{p}_{x,k}, \hat{p}_{y,k})$ to the true trajectory $(p_x, p_y, k)$ at each time instant over $N_m = 500$ Monte Carlo simulations. In equation (28), $(\hat{p}_{x,k}(m), \hat{p}_{y,k}(m))$ denotes the estimated position vector in the $m$th Monte Carlo run at the $k$th step:

$$\text{RMSE} = \sqrt{\frac{1}{N_m} \sum_{m=1}^{N_m} \left[ (\hat{p}_{x,k}(m) - p_{x,k})^2 + (\hat{p}_{y,k}(m) - p_{y,k})^2 \right]}$$

(19)

The performance comparison between the proposed CV-IMM-EKF and the GPS-only approach is shown in Figure 10 with respect to the RMSE in distance. It is obvious that the proposed CV-IMM-EKF method outperforms the GPS alone localization. To indicate the enhancement of vehicular positioning of the proposed approach, the enhancement indicator $\mu$ is calculated as follows:

$$\mu = \frac{A_{RMSE}}{B_{RMSE}} \times 100\%$$

(20)

The enhancement of vehicular positioning is shown in Table I. Compared to the GPS-based localization, the proposed CV-IMM-EKF approach achieves the enhancement of $\mu = 35.59$ per cent. If $A_{RMSE}$ is better than $B_{RMSE}$, $\mu$ will be greater than 0. The increase of $\mu$ is link to the good performance of $A_{RMSE}$.

By describing the transition of the measurement noise as a first-order $M$-state jump Markov chain, the proposed CV-IMM-EKF approach has been proved to achieve better performance in a scenario that is similar to a practical one.

5. Conclusions

In this paper, we discuss three applications of intelligent computing in vehicular networks and their performance and effect is better. In vehicular networks, based on the intelligent computing, multidimensional analysis of vehicular network testbed data, the number of message copies increased dramatically, and the reachability and vehicular positioning enhancement have been improved. So the application of intelligent computing in the vehicle network system will be more extensive, and the computing intelligence will be an

Figure 10 CV-IMM-EKF and GPS performance in positioning error
important technical means of car networking and improve the development of the car networking system.

References


**Corresponding author**

Xuting Duan can be contacted at: duanxuting@buaa.edu.cn

For instructions on how to order reprints of this article, please visit our website:

www.emeraldgrouppublishing.com/licensing/reprints.htm

Or contact us for further details: permissions@emeraldinsight.com
Markov probabilistic decision making of self-driving cars in highway with random traffic flow: a simulation study

Yang Guan
Tsinghua University, Beijing, China

Shengbo Eben Li
Department of Automotive Engineering, Tsinghua University, Beijing, China, and Jingliang Duan, Wenjun Wang and Bo Cheng
Tsinghua University, Beijing, China

Abstract
Purpose – Decision-making is one of the key technologies for self-driving cars. The high dependency of previously existing methods on human driving data or rules makes it difficult to model policies for different driving situations.

Design/methodology/approach – In this research, a probabilistic decision-making method based on the Markov decision process (MDP) is proposed to deduce the optimal maneuver automatically in a two-lane highway scenario without using any human data. The decision-making issues in a traffic environment are formulated as the MDP by defining basic elements including states, actions and basic models. Transition and reward models are defined by using a complete prediction model of the surrounding cars. An optimal policy was deduced using a dynamic programming method and evaluated under a two-dimensional simulation environment.

Findings – Results show that, at the given scenario, the self-driving car maintained safety and efficiency with the proposed policy.

Originality/value – This paper presents a framework used to derive a driving policy for self-driving cars without relying on any human driving data or rules modeled by hand.

Keywords Markov decision process, Decision-making, Dynamic programming, Self-driving cars

Paper type Research paper

1. Introduction

Recently, there has been a steady increase in interest of many researchers toward developing technologies for self-driving cars. This technology has the potential to reshape mobility by enhancing the safety, accessibility, efficiency and convenience of transportation. A major milestone in self-driving cars was the DARPA Urban Challenge. In 2007, six teams finished the event, which demonstrated that fully autonomous urban driving is possible (Buehler et al., 2009). The Google self-driving car and Tesla’s Autopilot system have been two popular examples that receive considerable attention since then.

The autonomous driving system can be divided into three layers: environment perception, decision-making and dynamic control. Environment perception detects surroundings in real time via radar, laser light, GPS, odometry and computer vision. The decision-making further understands the environment and predicts the movement of different participants. It, then, conducts maneuver selection and path planning. Finally, dynamic control instructs throttle, brake and steering of self-driving cars. In environment perception, Janai et al. (2017) reviewed the current state-of-the-art on several specific aspects in computer vision for autonomous vehicles, including recognition, reconstruction, motion estimation, tracking, scene understanding and end-to-end learning. Patole et al. (2017) summarized various aspects of automotive radar signal processing techniques as well as unique problems associated with automotive radars such as pedestrian detection. For the perception of multiple vehicles, Liu et al. (2018) presented a novel distributed Bayesian filtering method using measurement dissemination for multiple unmanned ground vehicles with dynamically changing interaction topologies. In decision-making, an existing method of motion prediction and risk assessment for intelligent vehicles based on the semantics used to define motion and risk was reviewed by Lefèvre et al. (2014).
Paden et al. (2016) proposed a structured decision-making in the contemporary autonomous driving system into route planning, behavioral decision-making, local motion planning and feedback control. He also summarized the state-of-the-art currently available methods on planning and control algorithms for urban environments. In dynamic control, Carvalho et al. (2015) presented an overview of control design methods that systematically handle uncertain forecasts for autonomous and semi-autonomous vehicles. For the control of multiple vehicles, this paper (Li et al., 2017) introduces a decomposition framework to model, analyze and design the platoon system from the perspective of multiagent consensus control. For the control of multiple vehicles, Zheng et al. (2017) presented a distributed model predictive control algorithm for longitudinal automation of connected vehicles with unidirectional topologies. Li et al. (2017b) presented a robust distributed control method for multiple vehicles with bounded parameter uncertainty and a broad spectrum of interaction topologies.

This paper focuses on the decision-making, which is a tricky problem to tackle owing to the highly dynamic, stochastic and uncertain nature of the traffic environment. The finite state machine (FSM) has become the most common approach for decision-making of self-driving cars since the Urban Challenge in 2007. It models finite situations of the traffic environment manually and obtains policies based on rules. These situations and rules make the decision system explicit, but it needs an experienced expert to design them to obtain good performance. The champion of 2008 Urban Challenge by Boss (Urmson et al., 2008) used this approach in its behavioral layer. It defined three high-level behaviors in advance, including lane driving, intersection handing and goal selection. Each of them corresponds to several low-level behaviors. The decision system chose these behaviors by rules prescribed in advance when the car ran at different situations. The runner-up Junior (Montemerlo et al., 2008) used FSM for switching among 13 driving situations and invoking exception behaviors to overcome stuckness. Besides, Odin (Bacha et al., 2008) ran with appropriate behaviors under the current situation using a system of the hierarchical FSM. The system is capable to distinguish among intersection, parking lot and normal road scenarios. Similarly, Furda et al. (Furda and Vlacic, 2011) presented a multiple-criteria decision-making approach to divide the decision-making task into two stages: the first one determined feasible maneuvers based on Petri nets, whereas the second used a multivariate utility function to select the most appropriate one. The FSM is simple and effective in given situations. However, it does not explicitly consider environment uncertainties and thus cannot be applied in a dynamic traffic scenario. Besides, it requires classifying situations and modeling policies by hand, which fails to make a decision under unusual situations.

Compared with explicit decomposition of the problem, Chen et al. (2015) trained a convolutional neural network (CNN) model on 484,815 images collected and labeled when playing a car racing video game TORSO for 12 h. The model mapped an input image to a small number of key perception indicators that is then sent in the designed controller. Bojarski et al. (2016) trained a CNN model, mapping raw pixels from a single front-facing camera directly to steering commands. The training data were collected by driving on a wide variety of roads and in a diverse set of lighting and weather conditions. About 72 h of driving data were collected and sampled at ten frames per second, which is used for training after being augmented further. The system can learn how to drive on local roads or highways with lane marking or not. The paper (Bojarski et al., 2017) explained how the neural-network-based system called PilotNet learns and makes decisions. They developed a method for determining which elements in the road image influence the steering decision the most. The paper also showed that PilotNet can learn more subtle features that are hard to program and anticipate by engineers. These end-to-end approaches simultaneously optimize all processing steps instead of human-selected intermediate criteria. It will perform better, especially for smaller systems, but requires huge amounts of data to get good performance.

This paper proposes a probabilistic decision-making method, which can be applied in a dynamic traffic environment while considering safety, efficiency and comfort simultaneously. The driving task was first formulated as the Markov decision process (MDP) by defining the environment state space, agent action space. Then, it built a state transition model and a reward model by using a prediction model of surrounding cars. The optimal policy was then automatically deduced using the value iteration method of dynamic programming (DP). The simulation results show the preset goal can be achieved. The framework of the proposed method is shown in Figure 1. This paper provides a probability decision-making approach that is neither dependent on human driving data nor limited to only a variety of rules

![Figure 1: Framework of the method](image-url)
modeled by hand. In addition, the formulation of the driving task under a two-lane highway scenario is presented, including state discretization and transition model estimation.

The remainder of the paper is structured as follows: In Section 2, a basic knowledge of the MDP is presented. Section 3 presents a MDP formulation for the driving task. Section 4 analyzes the result policy and provides explanations and intuitions about the MDP method. Finally, Section 5 concludes with remarks on the main work of this paper and potential areas for future research.

2. Basic knowledge of MDP

This section will give an overview about the used terminology. There are five basic elements of MDP, that is the tuple \( \langle S, A, r(s, a, s'), p(s, a, s'), \gamma \rangle \), whereas \( S \) denotes the set of all states, \( A \) denotes the set of all actions, \( r(s, a, s') \in \mathbb{R} \) denotes the expected immediate reward on transition from state \( s \) to \( s' \) under action \( a \), where \( s, s' \in S \), \( a \in A \), and \( \mathbb{R} \) denotes set of all possible rewards. And \( p(s, a, s') \) denotes the probability of transition to state \( s' \) from state \( s \) taking action \( a \).

A policy \( \pi \) is a stochastic rule by which the agent selects actions as a function of states. The agent’s objective is to maximize the amount of reward it receives over time. That is, finding an optimal policy \( \pi^* \) satisfying formula (1), whereas \( v_{\pi}(s) \) denotes the expected return from state \( s \) using policy \( \pi \).

\[
\pi_* = \arg \max_{\pi} v_{\pi}(s) \tag{1}
\]

For all \( s \in S \).

The Bellman optimality equation (2) is a special consistency condition that the optimal value functions must satisfy and that can, in principle, be solved for the optimal value functions, from which an optimal policy can be determined with the value iteration method.

\[
\pi_* = \arg \max_{a} \sum_{s', r} p(s', r | s, a) [r + \gamma v_{\pi_*}(s')] \tag{2}
\]

3. MDP formulation for driving task

MDP is a mathematically formulation for decision and control problems with uncertain system behavior. To derive the optimal policy using it, a tuple including state, action, transition model, reward model and discounting is first required.

3.1 Environment state space

Variables selected in states are supposed to contain the complete environmental information, such as properties of road, own car and other cars, which is required by the agent decision. However, with the number of variables included in a state rising, the number of states increases exponentially, which causes computation and memory problems. As a result, it is necessary to do simplification work to minimize the number of these variables. Consider there is only one car of interest and assume that cars are always parallel to the lane line. In addition, distance is more worthy of concern than absolute position in the longitudinal direction. Consequently, we formulate a tuple as a state.

\[
(\Delta x_{\text{lon}}, x_{\text{lat,ego}}, x_{\text{lat,veh}}, v_{\text{lon,ego}}, v_{\text{lon,veh}}) \in S_e \tag{3}
\]

where \( \Delta x_{\text{lon}} \) represents the longitudinal distance between the ego car and the other car, and it is positive when the ego car is behind the other car. \( x_{\text{lat,ego}}, x_{\text{lat,veh}} \) denote lateral positions of the ego car and the other car, and \( v_{\text{lon,ego}}, v_{\text{lon,veh}} \) denote the longitudinal velocity, respectively.

MDP needs a discrete description. For variables related to distance, we discrete it by dividing the road into no overlapping areas with equal distance along longitudinal and lateral directions. Besides, the speed is discreted with a fixed interval. The state space is discretized as shown in Figure 2.

3.2 Agent action space

Action space has all decisions that we can make. There are two main layers of decision-making architecture for intelligent cars, that is driving behavior layer and trajectory planning layer. Therefore, decisions can be deduced from all trajectories planned for all driving behaviors. As a result, the action space should contain all these trajectories. First, we define our behavior set. Li et al. (2017a) categorized driving behavior in highway traffic into 12 maneuver states. Here we define behavior set \( B \) as:

\[
B = W \times D \tag{4}
\]

where \( W \) denotes the basic behavior set and \( D \) denotes the degree set of basic behavior radicalism. They are defined as:

\[
W = \{ \text{go straight, turn left, turn right} \} \tag{5}
\]

\[
D = \{ \text{very radical, medium radical, normal, medium cautious, very cautious} \} \tag{6}
\]

Then, we use a tuple to represent a trajectory in the action space,

\[
(\bar{a}_{\text{lon}}, v_{\text{lat}}, t) \in A_e \tag{7}
\]

where \( \bar{a}_{\text{lon}} \) denotes the longitudinal acceleration, \( v_{\text{lat}} \) denotes the lateral speed and \( t \) denotes the time that actions go through. Every \( b \in B \) corresponds a subset of \( A_e \). We can discretize all these subsets the same way as state discretization. \( a_{\text{lon}} \) and \( v_{\text{lat}} \)
can be discretized within their scope of experience. However, a reasonable range should be determined for the discretization of t. It cannot be too long or too short, so 1-3 s is appropriate.

3.3 Environment transition model

3.3.1 Prediction model

Transition model \( p(s, a, s') \) describes the probability of transition from state \( s \) to \( s' \) under action \( a \). Thus, a prediction model for the other car is needed to obtain the transferred state. Suppose the scenario has two lanes and the driving behavior set includes lane keeping and lane change. First, the probability of driving behavior is predicted using statistic data, and then, the motion prediction is carried out according to the corresponding trajectory of that driving behavior.

3.3.2 Transition model

Continuous state space \( S_\Gamma \subseteq \mathbb{R}^n \) and discrete state space \( \mathcal{S} \) can be linked by a random variable \( I_\Gamma : S_\Gamma \rightarrow \mathcal{S} \), which is defined on a probability space \( (S_\Gamma, \mathcal{F}, P) \). \( \mathcal{F} \) is the Borel \( \sigma \)-algebra on \( S_\Gamma \). (Royden and Fitzpatrick, 1968) Thus, for every discrete state \( s \in \mathcal{S} \), its corresponding continuous state set is \( I_\Gamma^{-1}(s) = \{s : I_\Gamma(s) \in s\} \). Then, we can acquire the probability measure in probability space \( (S, \mathcal{F}', P') \), where \( \mathcal{F}' \) is the power set of \( S \). For an event \( D \in \mathcal{F}' \), its probability of occurrence can be calculated by formula (8).

\[
P_c(D) = P(I_\Gamma^{-1}(D)) = \int_{s \in I_\Gamma^{-1}(D)} p(s) \tag{8}
\]

In the same way, action sets can also be associated by random variable \( I_a : A_c \rightarrow A \). Therefore, the probability of transition from state \( s \in S \) to \( s' \in S \) under action \( a \in A \) can be solved by formula (9).

\[
p(s'|s, a) = P(I_\Gamma^{-1}(s')|I_\Gamma^{-1}(s'), I_a^{-1}(a)) = \int_{s' \in I_\Gamma^{-1}(s')} \int_{s \in I_\Gamma^{-1}(s)} \int_{a \in I_a^{-1}(a)} p(s'|s, a) \tag{9}
\]

It is hard to solve the transition probability using formula (9) directly. However, it is feasible to approximate it with the Monte Carlo method. We first sample Q state points with weight on the continuous state space \( I_\Gamma^{-1}(s) \) corresponding to the discrete state \( s \). Using action \( a \) and our prediction model, we get the Dirac distribution of \( s' \) for every sample point. Then, the distribution of \( s' \) given \( s \) and \( a \) is obtained by weighted average on all the samples. It is also a Dirac distribution. Finally, we can get \( p(s'|s, a) \) by summing the probability on all \( s' \in I_\Gamma^{-1}(s') \). Figure 3 shows a sampling instance of turning left.

3.4 Environment reward model

The reward model \( r(s, a, s') \) can be obtained just the same way as \( p(s, a, s') \). But how many rewards the agent is able to get after adopting action \( a \) need to be determined first. This reward has a relationship with the driving goals, which usually include safety, efficiency, comfort, economy and compliance with traffic rules and daily driving habits. We set the reward a large negative number when the ego car goes into a bad terminal state to ensure safety. Bad states include colliding with the other car, driving out of the road and being dumped by the other car. For efficiency, a large positive number is assigned to the reward when the ego car dumps the other car, which is the good terminal state. And we achieve other goals by designing rewards as formula (10) when the ego car does not transfer to a terminal state.

\[
r = \sum_{t=1}^{T} \Delta t (f_{\text{com}} a_{\text{com}} + f_{\text{fnd}} v_{\text{fnd}} + f_{\text{obs}} (I_{\text{right}})) \tag{10}
\]

To achieve the purpose of comfort and economy, a negative factor \( f_{\text{con}} \) is added in front of the acceleration to punish the large value of it. To encourage overtaking moves, a positive factor \( f_{\text{fnd}} \) is added in front of the lateral velocity. Because we need the whole action execution process to infer the environment model, an action is decomposed into several steps to implement while a step time is \( \Delta t \). To keep the ego car in the right lane as far as possible, we give it a negative value \( f_{\text{obs}} \) when it is in the left lane in a step but nothing when it is not. By multiplying all the three terms mentioned above by the step time and summing over them on all the steps during the action implementation, the reward can be obtained.

4. Policy evaluation

For the evaluations, we use a two-lane highway scenario. The self-driving agent has to cope with the other vehicle and
manage tasks like overtaking and avoiding collision. A detailed analysis of the results and tests in a two-dimensional simulation environment has been done.

4.1 Analysis of the results
After building the MDP model, we applied the value iteration method to get the optimal policy. When the ego car transfers to a bad terminal state, it is given a reward of \(-1000\). On the contrary, a reward of 1000 is given when it goes into a good terminal state. Otherwise, we assign the reward using formula (10) while setting \(f_{\text{com}} = 0, f_{\text{con}} = 0, f_{\text{del}} = -30\). Besides, we set discounting factor \(\gamma\) to 0.9. There are 64 samples of the ego car and the other car. Hence, the number of samples is \(64 \times 64\). The results are as shown in Figure 4 and Figure 5. Figure 4 shows the optimal policy when the other car is in the right lane and its speed is between 0 and 4 m/s. Figure 4a, 4b and 4c, respectively, represent the state value, the optimal acceleration and the optimal lateral speed when the ego car is also in the right lane, whereas Figure 4d represents the optimal lateral speed when the ego car is in the left lane. Each sub-figure corresponds to values under 165 states, which consists of 33 discrete values of \(\Delta x_{\text{lon}}\) horizontally and five discrete values of \(\Delta x_{\text{lon}}\) vertically.

As can be seen in Figure 4a, when \(\Delta v_{\text{lon}} > 10\) m, the state value increases with the vehicle speed increasing. This is because when the ego car is beyond the other car, it can reach the good terminal state to get the 1000 reward. Because future rewards decay exponentially with respect to steps taken from the current state, the ego car arrives faster, the greater return it gets, and that is why states with a greater speed have a higher value. By contrast, when \(\Delta x_{\text{lon}} > -30\) m and \(<0\), the state value decreases with the increase of the vehicle speed. This is because when the ego car is close to its front car, the greater its speed, the more likely it collides and gets a reward of \(-1000\), the lower the state value.

The optimal policy is the policy that leads to highest values for all states. The Bellman equation shows that the value of a state consists of instantaneous reward and the value of the following state. Therefore, policy selection is a trade-off among three factors including efficiency, safety and avoiding staying in the left lane too long. As shown in Figure 4b, when the ego car is behind and its speed is small, efficiency is more worthy of attention than safety. Thus, a large acceleration is adopted. However, with the speed increasing, the agent is more concerned about preventing collision than being faster. As a result, a small acceleration or even a negative one is taken in this case. On the other hand, when \(\Delta x_{\text{lon}} < 0\), optimal accelerations decrease with \(|\Delta x_{\text{lon}}|\) decreasing also owing to safety considerations. When the ego car goes to the front, there is almost no collision avoidance problem. In addition, there is no comfort or economy consideration in the reward function. The goal is to reach the destination as soon as possible, so the maximum acceleration is chosen, namely, 4 m/s².

For the lateral speed selection, it is still the trade-off among the three factors. As can be seen in Figure 4c, when the ego car runs behind, lateral speed changes from 0 to 2 m/s with the increasing of the vehicle speed. That is because the larger the speed, the more unsafe the agent feels, so it switches to the left lane to avoid collision. For the same reason, the optimal lateral speed changes from 0 to 2 m/s with the distance decreasing.

Figure 4 Optimal policy when the other car is at the right lane with speed 0-4 m/s.

Notes: (a) Maximal value; (b) acceleration; (c) lateral speed (ego on the right lane); (d) lateral speed (ego on the left lane)
However, it changes back to 0 when the distance continues to decrease. This is because in emergency, owing to physical constraints of the tire, the lateral force is turned to 0 to make the longitudinal force the maximal value. When the ego car runs at the front, there are no collision worries, so the lateral speed always equals 0.

In Figure 4d, we can see when the ego car runs behind, the greater the distance and the lower the speed, the more the tendency of changing lanes. In these cases, it is the main consideration to avoid staying in the overtaking lane. Otherwise, we care more about the efficiency and safety. Then, when it runs at the front, with the distance increasing, the focus of our attention changes from safety to rule compliance, and then efficiency. And that is why the optimal lateral speed changes from 0 to 2 m/s, and then 0.

Figure 5a shows the importance of a more sophisticated prediction model. The situation is basically the same as the above example, except the other car drives on the right lane with a speed between 12 and 16 m/s. Obviously, the safety distance shrinks compared with the case where the other car is in low speed, because there is more time to brake. The difference between 5a and 5b is 5a has a complete driving behavior set in its prediction model, whereas 5b only has lane keeping behavior. The missing uncertainty leads to an extra lane changing behavior when the rear car is approaching, because the ego car predicts the rear car will go straight so that it does this to avoid collision. However, this is extremely dangerous when it is really driving because the rear car is very likely to overtake in that situation. The complete prediction model is able to forecast this, and consequently, the ego car in Figure 5a holds lane.

4.2 Simulation test

The decisions of the agent with the other car were practically evaluated in a two-dimensional simulation environment. The speed limit is 30 m/s for the ego car. Several tests were done when the other car was being controlled manually with acceleration and the steering wheel. One of the simulation processes is shown in Figure 6. In this process, the other car is being controlled to behave at random, as the blue lines shown in Figure 6a and 6b. We can see the agent showed reasonable
behavior. Sometimes the controlled car fast approached the ego car in the right lane and overtook it, just as what happened around 10 s. In this situation, the ego car would keep a constant speed and stay in the right lane rather than turning left to avoid collision. Sometimes the ahead controlled car suddenly decelerated, as what happened between 20 s and 30 s. Under this circumstance, it can be seen that the ego car chose to decelerate too and when it slowed down, it chose to turn left and overtook the front car. Besides, when the high-speed controlled car overtook the ego car from the left lane and turned to the right lane suddenly, the ego car would decelerate right away, just as shown around 20 s, 50 s and 62 s.

It can be seen that the results obtained here are consistent with previous analyses. The deduced policy can adapt to any movement of the surrounding car and make the ego car drive safely and efficiently.

4.3 Comparison to existing methods
From the above results, it can be seen that the MDP defines per step reward and uses the DP method to choose behavior in each state to get the maximal expected total reward in the future. To implement the DP, it first needs to estimate the transition model through the Monte Carlo method. Compared with the FSM method, which chooses behavior by rules, the MDP does not need too much experience to design the whole decision system but a little knowledge to design the per step reward, which is related to the driving goals. Compared with the CNN method, which chooses action by a CNN trained on a large number of labeled driving images, the whole process of the MDP does not need any driving data but an environment model used in equation (2) to get the policy by iteration. As a result, the performance of the MDP is restricted by the precision of the estimated model. Besides, a large state space will lead to expensive computing cost when using the DP. In conclusion, the MDP is more suitable for problems where the precise environment model is available and the surrounding traffic environment is not too complex.

5. Conclusion
This paper presents a method that can automatically deduce the optimal behavior for autonomous driving. This method formulates driving tasks as the MDP and integrates a sophisticated motion prediction model of the surrounding car, in which predictions in continuous space and MDP planning in discrete space are combined by means of the probability method. The deduced optimal policy in the given two-lane highway scenario is evaluated analytically and used in a two-dimensional simulation environment. The results show that it behaves reasonably to achieve safe and efficient driving, such as braking when the front car slows down, overtaking the front car when it drives slow and keeping its lane rather than changing lanes when the rear car is fast approaching. This method could be applied in standard scenarios, such as highway and park driving, so that there is no need to model policies by hand comparing with rule-based methods.

But there are several issues that need to be improved. First of all, the time needed rises sharply as the dimension of the state space increases owing to high computational complexity. Because lots of states are virtually impossible to meet in practice, it is reasonable to compute on-line when a new state is encountered rather than doing all the computation off-line. In addition, the result shows that fixed discretization is too fine for states whose values are close and discrete interval is too large for states whose values have a significant difference. A more efficient discretization method is required. Finally, to use the decision-making model in reality, a more detailed vehicle dynamic model and a traffic model need to be included; besides, the approximation function should be used to represent the value function in the continuous state space to enhance generalization of the policy and reduce the computation cost in large-scale decision problems.

References
Li, S.E., Qin, X., Li, K., Wang, J. and Xie, B. (2017b), “Robustness analysis and controller synthesis of homogeneous vehicular platoons with bounded parameter


Corresponding author
Shengbo Eben Li can be contacted at: lishbo@tsinghua.edu.cn

For instructions on how to order reprints of this article, please visit our website: www.emeraldgrouppublishing.com/licensing/reprints.htm
Or contact us for further details: permissions@emeraldinsight.com
41  A novel intelligent vehicle risk assessment method combined with multi-sensor fusion in dense traffic environment
Xunjia Zheng, Bin Huang, Daiheng Ni and Qing Xu

55  Stable trajectory planning and energy-efficiency control allocation of lane change maneuver for autonomous electric vehicle
Liwei Xu, Guodong Yin, Guangmin Li, Athar Hanif and Chentong Bian

66  Applications of intelligent computing in vehicular networks
Daxin Tian, Weiqiang Gong, Wenhao Liu, Xuting Duan, Yukai Zhu, Chao Liu and Xin Li

77  Markov probabilistic decision making of self-driving cars in highway with random traffic flow: a simulation study
Yang Guan, Shengbo Eben Li, Jingliang Duan, Wenjun Wang and Bo Cheng