Reachability-Based Confidence-Aware Probabilistic Collision Detection in Highway Driving

Wang, Xinwei; Li, Zirui; Alonso-Mora, Javier; Wang, Meng

DOI
10.1016/j.eng.2023.10.010

Publication date
2024

Document Version
Final published version

Published in
Engineering

Citation (APA)
Wang, X., Li, Z., Alonso-Mora, J., & Wang, M. (2024). Reachability-Based Confidence-Aware Probabilistic Collision Detection in Highway Driving. *Engineering*, *33*, 90-107. https://doi.org/10.1016/j.eng.2023.10.010

Engineering 33 (2024) 90–107

Contents lists available at ScienceDirect

Engineering

journal homepage: www.elsevier.com/locate/eng

Research
Safety for Intelligent and Connected Vehicles—Article

Reachability-Based Confidence-Aware Probabilistic Collision Detection in Highway Driving

Xinwei Wang [a,b,*], Zirui Li [c,d], Javier Alonso-Mora [b], Meng Wang [d]

[a] School of Engineering and Materials Science, Queen Mary University of London, London E1 4NS, UK
[b] Department of Cognitive Robotics, Delft University of Technology, Delft 2628 CD, Netherlands
[c] School of Mechanical Engineering, Beijing Institute of Technology, Beijing 100081, China
[d] Chair of Traffic Process Automation, "Friedrich List" Faculty of Transport and Traffic Sciences, TU Dresden, Dresden 01069, Germany

ARTICLE INFO

Article history:
Received 27 October 2022
Revised 5 May 2023
Accepted 11 October 2023
Available online 28 December 2023

Keywords:
Probabilistic collision detection
Confidence awareness
Probabilistic acceleration prediction
Reachability analysis
Risk assessment

ABSTRACT

Risk assessment is a crucial component of collision warning and avoidance systems for intelligent vehicles. Reachability-based formal approaches have been developed to ensure driving safety to accurately detect potential vehicle collisions. However, they suffer from over-conservatism, potentially resulting in false–positive risk events in complicated real-world applications. In this paper, we combine two reachability analysis techniques, a backward reachable set (BRS) and a stochastic forward reachable set (FRS), and propose an integrated probabilistic collision–detection framework for highway driving. Within this framework, we can first use a BRS to formally check whether a two-vehicle interaction is safe; otherwise, a prediction-based stochastic FRS is employed to estimate the collision probability at each future time step. Thus, the framework can not only identify non-risky events with guaranteed safety but also provide accurate collision risk estimation in safety-critical events. To construct the stochastic FRS, we develop a neural network-based acceleration model for surrounding vehicles and further incorporate a confidence-aware dynamic belief to improve the prediction accuracy. Extensive experiments were conducted to validate the performance of the acceleration prediction model based on naturalistic highway driving data. The efficiency and effectiveness of the framework with infused confidence beliefs were tested in both naturalistic and simulated highway scenarios. The proposed risk assessment framework is promising for real-world applications.

1. Introduction

Autonomous vehicles (AVs) are expected to significantly benefit future mobility, and one of the prerequisites for enabling the public availability of AVs is to ensure autonomous driving safety [1]. Highways are structured environments designed for vehicles to drive at a consistently high speed for efficient road trips and are the first applications of Levels 1 and 2 automated vehicles. During the transition from human-driven and lower-level automated vehicles to high-level AVs, driving safety on highways must be addressed for both conventional vehicles and AVs. Extensive research has been conducted on risk assessment and collision detection has been conducted to identify driving risk and potential vehicle crashes [2,3]. To accurately detect potential vehicle collisions, researchers have developed reachability-based formal approaches [4], because they can mathematically check whether the behavior of a system satisfies the given safety requirements.

Reachability analysis (RA) has been widely employed to verify driving safety [5,6]. RA computes a complete set of states in which an agent (e.g., a vehicle) can reach an initial condition within a certain time interval [7]. Based on RA, safety verification can be performed by propagating all possible reachable spaces for the AVs and other traffic participants on the road. Thus, safety is ensured if such a forward reachable set (FRS) of the automated vehicle does not intersect that of other traffic participants during the propagation period. Corresponding with this definition, the FRS can formally verify safety between road users, but it easily results in over-conservative results because state propagation is feedforward-based and ignores traffic participant interactions (i.e., vehicles react to the surrounding environment and adjust the control output) [7].

[*] Corresponding author.
E-mail address: xinwei.wang@qmul.ac.uk (X. Wang).

https://doi.org/10.1016/j.eng.2023.10.010

Alternatively, RA can be conducted in a closed-loop manner [8]. Given a target set representing a set of undesirable states (e.g., collision states between two vehicles) and worst-case disturbances, we define the backward reachable set (BRS) as the set of states that can result in being in the target set during a certain time horizon. Specifically, a BRS is the state set in which a control strategy does not exist to prevent an AV from entering the target set under worst-case disturbances. Thus, an unsafe area can be directly identified by the BRS using the initial vehicle states. Note that the BRS can be computed offline in advance and the cached BRS used in real time. Although the BRS considers control reactions from the AV and is less conservative than the FRS, it still suffers from over-conservatism owing to the worst-disturbance closed-loop reactions.

We aim to use RA to evaluate driving risk and detect potential collisions. However, both of those RA approaches suffer from over-conservatism. The over-conservative of forward reachability is typically reduced by keeping the time horizon for an FRS small and frequently recomputing it. Although the BRS incorporates closed-loop feedback to consider the worst disturbance from the surrounding vehicle, general interactions between vehicles are not pursuit–evasion interactions [9]. It is reasonable to consider a more realistic situation: The interactions are not adversarial but may result in crashes.

1.1. Related studies

Driving risk assessments are crucial for identifying potential collisions and quantifying risk levels. Various surrogate measures of safety (SMoS) have been constructed to measure driving risk. Typically, SMoS can be calculated in a time series, including time-to-collision [10,11], time-headway [12], and time-to-lane crossing [13]. However, these developed SMoS are mostly deterministic, which means that uncertainties in vehicle motion and environment are not considered. Although several probabilistic approaches [14–17] have been integrated to improve the performance of SMoS, these methods can suffer from an additional computational load (particularly for long-term prediction) and cannot formally ensure driving risk.

Driving risk can also be assessed by estimating the current and future collision probabilities, given the surrounding road participants. Collision detection can generally be divided into three methodologies: neural network-based, probabilistic, and formal verification approaches. Neural networks can provide accurate vehicle collision detection by classifying safety–critical scenarios. For instance, a collision–detection model using a neural-network-based classifier was developed in Ref. [18]. The proposed model uses onboard sensor data, including acceleration, velocity, and separation distance as inputs into a neural network-based classifier and outputs whether alerts are activated for a possible collision. A specific animal-detection approach was proposed in Ref. [19], where a deep semantic segmentation convolutional neural network was trained to recognize and detect animals in dynamic environments. Although neural-network-based approaches are effective in identifying potential collisions, the trained classifier generally cannot include clear decision rules and is difficult to interpret.

Prediction-based approaches have also been widely adopted to address the uncertainties of surrounding road participants and better estimate collision probabilities. A conceptual framework for analyzing and interpreting dynamic traffic scenes for collision estimation was designed in Ref. [20]. Collision risks were estimated as stochastic variables and predicted by relying on driver behavior evaluation with hidden Markov models [20]. To calculate collision probability, Annell et al. [21] combined intention estimation and a long-term trajectory prediction module to construct a probability field for future vehicle positions. For a set of local path candidates, a collision risk assessment considering the lane-based probabilistic motion prediction of the surrounding vehicles was proposed in Ref. [22]. However, these methods typically require predefined position distribution parameters, which can affect the adaptability of probabilistic collision detection.

Formal verification approaches that can ensure system safety given a specific control input range and safety requirements have been employed to address collision detection [23,24]. As a formal approach, RA computes a complete set of states in which an agent (e.g., a vehicle) can reach an initial condition within a certain time interval [7]. Based on RA, safety verification can be performed by propagating all possible reachable spaces of the AVs and other traffic participants forward in time and checking the overlaps. To reduce the over-conservative nature of forward reachability, Althoff et al. [25] developed a stochastic FRS discretizing the reachable space into grids with probability distributions. At each time step, the collision probability is provided by summing the probabilities of the states that vehicles intersect. However, this approach is based on Markov chains, which assume that the vehicle state and its control input evolve only according to the current state. In addition, it cannot explicitly address two-dimensional motion because lane-change maneuvers are not considered.

RA can also be conducted in a closed-loop manner with worst-case disturbances, namely the BRS [8]. Although the BRS can be constructed offline using advanced Hamilton–Jacobi–Isaacs partial differential equation (HJI PDE) solvers [26], it suffers from over-conservatism owing to worst-disturbance assumptions, which are not realistic. To fill this research gap, we aim to combine the BRS and stochastic FRS techniques into an integrated collision–detection framework, which can theoretically ensure safety in non-risky interactions and provide an accurate collision estimation in safety–critical scenarios.

1.2. Objectives and contribution

A stochastic FRS can accurately estimate collision when integrated with a prediction model. However, it cannot provide a safety guarantee, as the employed probabilistic distribution predictor can fail in corner cases, resulting in false negative/positive cases (e.g., in a safe cut-in scenario, the BRS can ensure safety, whereas the stochastic FRS estimates a biased collision probability, resulting in a false positive alarm). This motivated us to propose an integrated collision–detection framework that includes both the BRS and stochastic FRS to evaluate highway driving risk (Fig. 1). The BRS is first computed using the HJI PDE [8]. If the relative positions of vehicles are identified as unsafe by the BRS, a stochastic FRS considering the surrounding vehicle maneuvering modes is further established to calculate the collision probability at each future time step. The stochastic FRS shares the same reachable states as the FRS. In addition, each state of the stochastic FRS has an estimated probability. Ideally, a stochasti BRS should be used directly for collision detection, but the computation of a stochastic BRS is not readily viable owing to its closed-loop form.

Based on the stochastic FRS, the collision probability between two vehicles can be calculated by summing the state probabilities where the two vehicles spatially overlap. If the collision probability

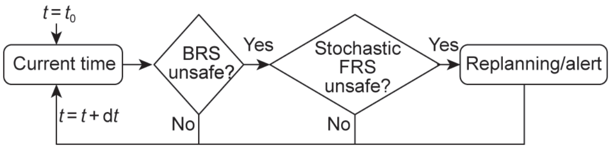


Fig. 1. Brief diagram of the integrated collision–detection framework. t: current time step; t_0: initial time step.

obtained is above a predefined threshold, the ego vehicle must execute an emergency brake or swerve to avoid collisions with other vehicles. The proposed framework benefits from both the BRS and FRS: Driving safety can be theoretically ensured when the relative vehicle positions are outside the unsafe area identified by the BRS; otherwise, the framework provides a collision probability based on a developed stochastic FRS.

To construct a stochastic FRS, we develop a long short-term memory (LSTM) model for multi-maneuver acceleration prediction on highways. The proposed model has two stages for maneuver prediction (i.e., lane-keeping, turning left/right on highways) and acceleration prediction, and the model input features are selected differently at each stage. We further incorporate a confidence-aware belief vector to generate a group of predicted acceleration distributions that can dynamically adjust the degree of confidence inferred from the current prediction accuracy [27]. The confidence-aware belief vector can result in a concentrated stochastic FRS when the LSTM model has a higher prediction accuracy and results in a more spread stochastic FRS when vehicles move unexpectedly.

The main contribution in this work is summarized as follows: ① We propose a multi-modal acceleration prediction model for surrounding vehicles and establish a stochastic FRS for each surrounding vehicle by leveraging the proposed acceleration predictor. Furthermore, we incorporate confidence awareness to generate a group of predicted acceleration distributions and dynamically update the degree of confidence, resulting in more accurate stochastic FRS and more agile collision detection results. ② An integrated probabilistic collision–detection framework that includes both the BRS and stochastic FRS is proposed to evaluate highway driving risk. Within the framework, an offline-computed and cached BRS is used online to check whether the car–car interaction safety can be theoretically ensured; if not, a stochastic FRS is computed online to provide an accurate collision probability at each future time step. We presented the results of stochastic FRS using the LSTM prediction model in our previous study [28]. In this paper, we significantly extend our previous study [28] by including an integrated collision–detection framework and infusing a confidence-aware belief vector for a more accurate stochastic FRS. Extensive and comprehensive experiments, which are different from those in Ref. [28] were conducted to validate the proposed framework. The remainder of the paper is organized as follows: Section 2 provides preliminaries on the BRS, FRS, and Markov-based stochastic FRS. In Section 3, a specific prediction-based confidence-aware stochastic FRS is developed. An integrated driving risk framework, including the BRS and stochastic FRS, is established in Section 4. Extensive experiments are presented in Section 5, and we conclude our paper in Section 6.

2. Preliminaries

2.1. BRS

Backward RA is considered to be an optimal control problem; thus, computing the reachable set is equivalent to solving an HJI PDE [8]. The system dynamics are defined by $\dot{x} = f(x, u, d)$, where $x \in \mathbf{R}^{n_1}$ and $u \in \mathcal{U} \subset \mathbf{R}^{n_2}$ are the state and control input ($\mathcal{U}$ is the admissible control input set), $d \subset \mathbf{R}^{n_3}$ is the disturbance, and $n_i \in \mathbf{Z}^{+}(i = 1, 2, 3)$ are the dimensions. The system dynamics are assumed to be uniformly continuous and bounded.

In the context of two-vehicle interactions, u corresponds to the control of the ego vehicle, and d corresponds to the control of the surrounding vehicle, as its actions are treated as disturbance inputs. Specifically, let $(x_{\text{ego}}, u_{\text{ego}})/(x_{\text{s}}, u_{\text{s}})$ represent the state and control of the ego/surrounding (s) vehicle, and x_{rel} be the system states between vehicles, for instance, the relative two-dimensional distances $(y_{1,\text{ rel}}, y_{2,\text{ rel}})$ and velocities $(v_{1,\text{ rel}}, v_{2,\text{ rel}})$. Therefore, the system dynamics are given by $\dot{x}_{\text{rel}} = f(x_{\text{ref}}, u_{\text{ego}}, u_{\text{s}})$. The formal definition of the BRS, denoted by $\text{BR}(t)$, for the related system is,

$$\text{BR}(t) := \{\bar{x}_{\text{rel}} : \exists u_{\text{s}}(\cdot), \forall u_{\text{ego}}(\cdot), \exists t^{\star} \in [t, 0], \{x_{\text{rel}}(t) = \bar{x}_{\text{rel}} \wedge \dot{x}_{\text{rel}} = f(x_{\text{rel}}, u_{\text{ego}}, u_{\text{s}}) \wedge x_{\text{rel}}(t^{\star}) \in \mathcal{T}\} \tag{1}$$

Here, $\text{BR}(t)$ represents the set of unsafe states $\bar{x}_{\text{rel}}$ at time t, from which if the surrounding vehicle followed an adversarial policy u_{s}, any policy u_{ego} would result in the state set $\mathcal{T}$ at $t^{\star} \in [t, 0]$ where two vehicles collide within a time horizon.

Assuming optimal (i.e., adversarial) surrounding vehicle actions, $\text{BR}(t)$ can be computed by defining a value function $V(t, x_{\text{rel}})$ that obeys the HJI PDE, where the solution $V(t, x_{\text{rel}})$ provides the BRS as its zero-sublevel set:

$$\text{BR}(t) = \{x_{\text{rel}} : V(t, x_{\text{rel}}) \le 0\} \tag{2}$$

The HJI PDE is solved starting from the boundary condition $V(t, x_{\text{rel}})$, which indicates whether the state x_{rel} belongs to the collision set $\mathcal{T}$. We cache the solution $V(t, x_{\text{rel}})$ for online use as a look-up table.

2.2. FRS and Markov-based stochastic FRS

An FRS is computed by considering all possible control inputs $u \in \mathcal{U}$ of a system $\dot{x} = f(x, u)$ given an initial set of states $\mathcal{X}_0$. The FRS of a system is formally defined as:

$$\text{FR}(t) := \{\bar{x} : \exists u(\cdot), \exists t^{\star} \in [0, t], x(0) \in \mathcal{X}_0 \wedge \dot{x} = f(x, u, d) \wedge x(t^{\star}) = \bar{x}\} \tag{3}$$

where $\text{FR}(t)$ is an FRS of states $\bar{x}$ at time $t^{\star} \in [0, t]$ from an initial state $x(0) \in \mathcal{X}_0$ at the current time 0 and subject to any input u belonging to the admissible control input set $\mathcal{U}$.

Based on the definition of the FRS, we can mathematically formulate a stochastic version of the FRS as $\mathcal{S}\text{FR}(t)$, where each state $\bar{x}$ within the FRS is associated with the state probability $p\left(\bar{x}\right)$.

$$\mathcal{S}\text{FR}(t) := \left\{\left(\bar{x}, \boldsymbol{p}\left(\bar{x}\right)\right) : \exists u(\cdot), \exists t^{\star} \in [0, t], x(0) \in \mathcal{X}_0 \wedge \dot{x} = f(x, u, d) \wedge x(t^{\star}) = \bar{x}\right\} \tag{4}$$

One of the most frequently used techniques is the approximation of stochastic processes using Markov chains, which represent a stochastic dynamic system with discrete states [25]. The discretized time step series are denoted as $\{0, 1, \ldots, e\}$, where e is the future final time step, and the duration of each time step is dt. Owing to stochastic characteristics, the system state at the predicted time step is not exactly known, and the probability $\boldsymbol{p}_i(k)$ is assigned to each state i at time step k (the discretized system state is denoted as i, j for simplicity). Subsequently, the probability vector $\boldsymbol{p}(k+1)$ composed of probabilities $\boldsymbol{p}_i(k)$ over all states is updated as

$$\boldsymbol{p}(k+1) = \boldsymbol{\Phi}\boldsymbol{p}(k) \tag{5}$$

where $\boldsymbol{\Phi}$ is the state transition matrix. Here, $\boldsymbol{\Phi}$ is time-invariant as the model is assumed to be Markovian.

To implement a Markov chain model, we need to first discretize the system state if the original system is continuous. The vehicle dynamic system is represented as a tuple with four discretized elements, including the two-dimensional vehicle positions and velocities. Additionally, the control input is to be discretized. The detailed discretization parameters are described in Section 5.1.

Each element $\boldsymbol{\Phi}_{ji}$ in the matrix, $\boldsymbol{\Phi}$ represents the state transition probability from state i to j. Note that the transition probabilities depend on the discrete input u as well, for example, each discrete input u generates a conditional transition probability matrix $\boldsymbol{\Phi}^u$. Specifically, each element $\boldsymbol{\Phi}^u_{ji}$ in the conditional matrix $\boldsymbol{\Phi}^u$ is the possibility starting from the initial state i to j under control $u \in \mathcal{U}$, where u represents the corresponding control input of $\boldsymbol{\Phi}^u_{ji}$. Therefore, the conditional probability $\boldsymbol{\Phi}^u_{ji}$ is expressed as

$$\boldsymbol{\Phi}^u_{ji} = \begin{cases} \boldsymbol{p}^u_i, \text{ if state } i \text{ reaches state } j \text{ with input } u \\ 0, \text{ otherwise} \end{cases} \tag{6}$$

where $\boldsymbol{p}^u_i$ is the control input probability for state i. The time index does not appear here because it is a Markov process. Thus, the overall state transition matrix is constructed as

$$\boldsymbol{\Phi}_{ji} = \sum_{u \in \mathcal{U}} \boldsymbol{\Phi}^u_{ji} \tag{7}$$

The probability distribution of the control input $\boldsymbol{p}^u_i$ is dynamically changed by another Markov chain with a transition matrix $\boldsymbol{\Gamma}_i$, depending on the system state i. This enables more accurate modeling of driver behavior by considering the frequency and intensity of changes in the control input. Consequently, the transition matrices $\boldsymbol{\Gamma}$ are to be learned by observation or set by a combination of simulations and heuristics. By incorporating the two transition matrices $\boldsymbol{\Phi}$ and $\boldsymbol{\Gamma}$, a Markov-based stochastic FRS with probabilities $\boldsymbol{p}(k)$ over discretized states can be obtained at each predicted time step k.

In Althoff et al. [25], the control transition probability matrices $\boldsymbol{\Gamma}$ only depend on the current control input and the state at the current time. Although computational efficiency is ensured by using such a simplified Markovian setting, the future control input and trajectories of a vehicle can be influenced by historical information and interactions with the surrounding environment [29]. Therefore, in this paper, we do not assume that the vehicle system state is Markovian. Instead, to address historical information and vehicle interactions, we aim to use a vehicle control predictor with multi-maneuvering modes to generate and dynamically update the transition matrices at each time step k as

$$\boldsymbol{\Phi}_{ji}(k) = \sum_{u \in \mathcal{U}} \boldsymbol{\Phi}^u_{ji}(k) \tag{8}$$

3. Prediction-based confidence-aware stochastic FRS

In this section, to provide a more accurate prediction of the surrounding vehicles, we first introduce a two-stage multi-modal acceleration prediction model consisting of lane-change maneuver prediction and acceleration prediction modules. We then establish the stochastic FRS by incorporating the proposed acceleration prediction model and infusing a confidence belief vector. Sections 3.1 and 3.2 have been presented in Wang et al. [28] and are included here for completeness. Note that the prediction model is replaceable if the accelerations can be predicted using bivariate normal distributions.

3.1. Acceleration prediction of a surrounding vehicle

Typically, vehicle trajectories and accelerations are predicted using current and historical information [29]. Thus, the prediction accuracy can be improved compared with using only the current states as inputs. This motivated us to establish an LSTM-based network to probabilistically predict future vehicle accelerations using both current and historical vehicle information. An overview of the developed two-stage acceleration prediction model is shown in Fig. 2 [30].

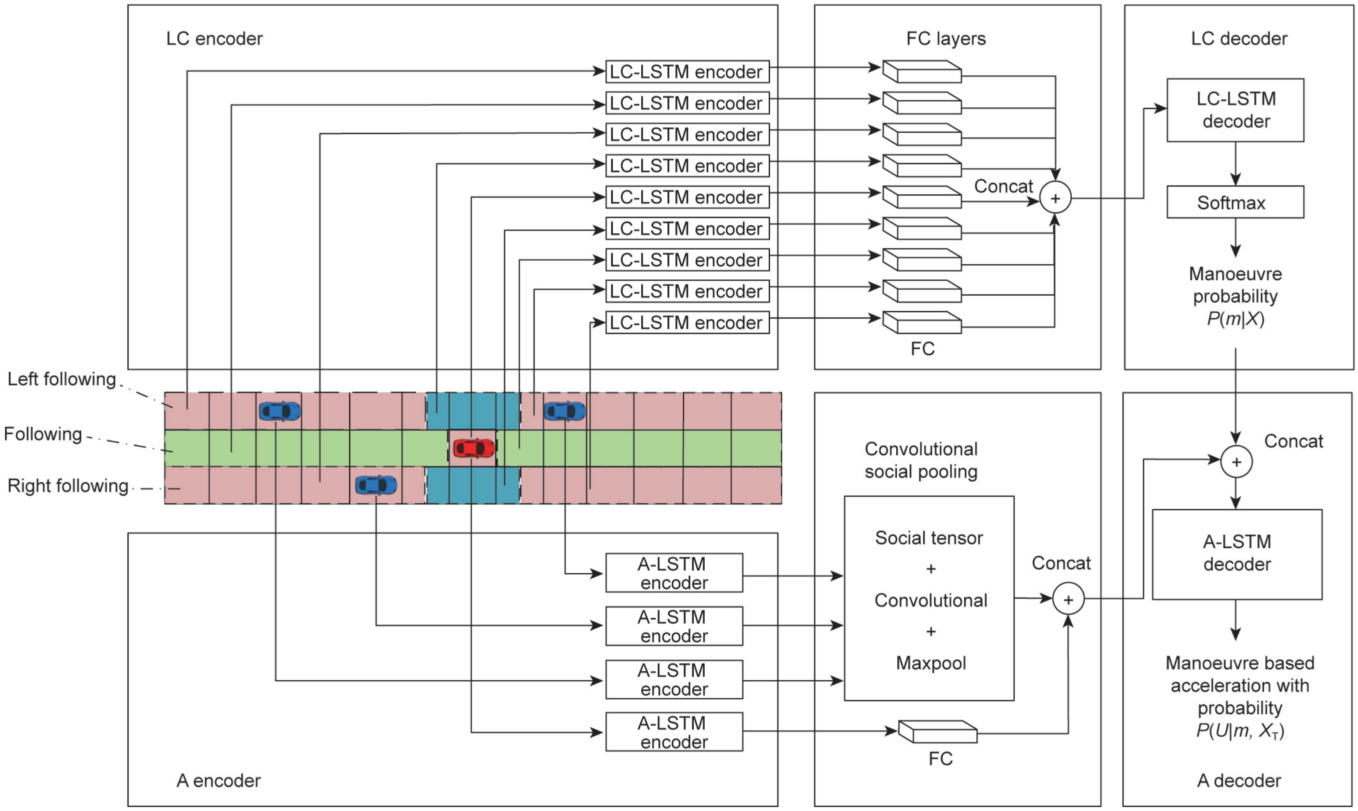


Fig. 2. Overview of the acceleration prediction model, consisting of lane-change maneuver prediction (LC) and acceleration prediction modules (A). The two modules both have an encoder–decoder structure but adopt and process historical information as input in different ways. Concat: concatenation operation; FC: fully connect layer; T-LSTM: the trajectory prediction model. A variant of the model for trajectory prediction was developed in Ref. [30]. X: all input features from the previous time step −h to the current time step 0; m: lane-change maneuver mode; U: predicted vehicle acceleration from time step 1 to time step e; X_T: position information for the vehicle being predicted and its surrounding vehicles from the previous time step −h to the current time step 0.

3.1.1. Two-stage vehicle acceleration prediction

A two-stage multi-modal trajectory prediction model was developed in Ref. [30]. In this paper, we maintain the same lane-change maneuver prediction model in the first stage but develop a new acceleration prediction model in the second stage. This is necessary because the acceleration prediction is employed to enable the dynamic update of the conditional probability $\boldsymbol{\Phi}^{u}_{ji}(k)$ in Eq. (8).

First, we briefly introduce the adopted lane-change maneuver prediction module from Wang et al. [30]. The input of the module is expressed as

$$\boldsymbol{X} = \left[x^{(-h)}, \ldots, x^{(-1)}, x^{(0)}\right] \tag{9}$$

where X represents all input features from the previous time step $-h$ to the current time step 0. At each historic time step, the collected input is composed of three parts: $x^{(\cdot)} = \left\{ {x_{\mathrm{T}}}^{(\cdot)}, b^{(\cdot)}, d^{(\cdot)} \right\}$, where ${x_{\mathrm{T}}}^{(\cdot)}$ is the position information for the vehicle being predicted and its surrounding vehicles, $b^{(\cdot)}$ contains two binary values to check whether the predicted vehicle can turn left and right, and $d^{(\cdot)} \in [-1, 1]$ is the normalized deviation value from the current lane center.

As shown at the top of Fig. 2, LSTM models are used to encode and decode the lane-change maneuver prediction model, in which the encoding information is passed to fully connected layers before decoding. The output of the model is a probability distribution $\mathrm{P}(m|X)$ for each lane-change maneuver mode m (i.e., change to the left, change to the right, and maintain the same lane) in time steps 1 to e.

For the acceleration prediction in the second stage, the input includes the historic positions of the vehicle being predicted and surrounding vehicles in addition to the historic accelerations ${x_{\mathrm{A}}}^{(-h:0)}$ of the vehicle being predicted:

$$\boldsymbol{X}_{\mathrm{T}} = \left[{x_{\mathrm{T}}}^{(-h)}, \cdots, {x_{\mathrm{T}}}^{(-1)}, {x_{\mathrm{T}}}^{(0)}, {x_{\mathrm{A}}}^{(-h:0)}\right] \tag{10}$$

Because we use additional acceleration information for the vehicle being predicted, we modify the input size of the LSTM encoder in Wang et al. [30] for the vehicle being predicted while maintaining the overall network structure unchanged. Detailed information on the second-stage model is provided in Refs. [29,30].

Given the input X_{T} and corresponding maneuver mode probability distribution $P(m|X)$, the output $P(\boldsymbol{U}|m, \boldsymbol{X}_{\mathrm{T}})$ of the second-stage acceleration prediction model is a conditional acceleration distribution over

$$\boldsymbol{U} = \left[u^{(1)}, \cdots, u^{(e)}\right] \tag{11}$$

where $u^{(\cdot)}$ is the predicted vehicle acceleration at each time step within the prediction horizon. Note that the prediction horizon and time increment are the same as those for the reachable set computation.

Given the three defined maneuvers m, the probabilistic multi-modal distributions are calculated as

$$P(U|X) = \sum\nolimits_m P_{\Theta}(\boldsymbol{U}|m, \boldsymbol{X}_{\mathrm{T}}) P(m|\boldsymbol{X}) \tag{12}$$

where outputs $\Theta = \left[\Theta^{(1)}, \ldots, \Theta^{(e)}\right]$ are time-series bivariate normal distributions. $\Theta^{(k)} = \cup_m \left\{ \mu^{k}_{1m}, \mu^{k}_{2m}, \sigma^{k}_{1m}, \sigma^{k}_{2m}, \rho^{k}_{m} \right\}$ corresponds to the predicted acceleration means and standard deviations along two dimensions and the correlation at future time step k under each maneuver mode m, respectively.

Under acceleration distributions Θ, the future vehicle trajectories are propagated as

$$\begin{cases} \boldsymbol{v}^{k+1}_{1m} = \boldsymbol{v}^{k}_{1m} + \mu^{k+1}_{1m}\mathrm{d}t \\ \boldsymbol{v}^{k+1}_{2m} = \boldsymbol{v}^{k}_{2m} + \mu^{k+1}_{2m}\mathrm{d}t \\ \boldsymbol{y}^{k+1}_{1m} = \boldsymbol{y}^{k}_{1m} + \left(\boldsymbol{v}^{k+1}_{1m} + \boldsymbol{v}^{k}_{1m}\right)\mathrm{d}t/2 \\ \boldsymbol{y}^{k+1}_{2m} = \boldsymbol{y}^{k}_{2m} + \left(\boldsymbol{v}^{k+1}_{2m} + \boldsymbol{v}^{k}_{2m}\right)\mathrm{d}t/2 \end{cases}. \tag{13}$$

where $\mathrm{d}t$ is the time increment, $\boldsymbol{v}^{k}_{1m}$, $\boldsymbol{v}^{k}_{2m}$, $\boldsymbol{y}^{k}_{1m}$, and $\boldsymbol{y}^{k}_{2m}$ are the propagated two-dimensional velocities and positions at future time step k for each maneuver mode m, respectively. $\left(\boldsymbol{v}^{0}_{1m}, \boldsymbol{v}^{0}_{2m}, \boldsymbol{y}^{0}_{1m}, \boldsymbol{y}^{0}_{2m}\right)$ denote the system states at time t.

To properly propagate probabilistic distributions from accelerations to trajectories, we assume that the trajectory variance and correlation propagation depend only on the acceleration at the current time rather than on the previous time steps. Otherwise, the trajectory variance/correlation at the future k step is determined by the acceleration variances/correlations from the current time step to the future k step, which complicates and makes the probabilistic distribution propagation intractable. Thus, the propagated trajectory standard deviations can be obtained as $\widetilde{\sigma^{k}_{1m}} = \sigma^{k}_{1m}(\mathrm{d}t)^2/2$ and $\widetilde{\sigma^{k}_{2m}} = \sigma^{k}_{2m}(\mathrm{d}t)^2/2$, and the correlation as $\rho^{k}_{m}(\mathrm{d}t)^2/2$. Therefore, the propagated probabilistic distributions of the vehicle position are expressed as $\widetilde{\Theta^{(k)}} = \left\{ \boldsymbol{y}^{k}_{1m}, \boldsymbol{y}^{k}_{2m}, \widetilde{\sigma^{k}_{1m}}, \widetilde{\sigma^{k}_{2m}}, \rho^{k}_{m} \right\}_{m=\{1,2,3\}}$.

3.1.2. Model training

Typically, a multi-modal prediction model is trained to minimize the negative log-likelihood (NLL) of its conditional distributions as

$$-\log\left(\sum\nolimits_m P_{\Theta}(\boldsymbol{U}|m, \boldsymbol{X}_{\mathrm{T}}) P(m|\boldsymbol{X})\right) \tag{14}$$

For a more accurate collision probability estimation, we focus on the potential collision when two vehicles intersect along their trajectories. Therefore, we directly minimize the trajectory prediction errors propagated from the acceleration prediction in line with Zhou et al. [31] as

$$-\log\left(\sum\nolimits_m P_{\widetilde{\Theta}}(\boldsymbol{Y}|m, \boldsymbol{X}_{\mathrm{T}}) P(m|\boldsymbol{X})\right) \tag{15}$$

where $\boldsymbol{Y} = \left[\boldsymbol{y}^{(1)}, \cdots, \boldsymbol{y}^{(e)}\right]$ is the propagated trajectories with distributions $\widetilde{\Theta}$, and $\boldsymbol{y}^{(k)} = \left\{\boldsymbol{y}^{k}_{1m}, \boldsymbol{y}^{k}_{2m}\right\}$ are the predicted positions of the vehicle at time step k under maneuver mode m.

To further improve the prediction performance, we separately train the lane-change maneuver and vehicle acceleration prediction models. This is because the proposed approach has a two-stage structure: the maneuver probabilities are first predicted and then predicted for the corresponding conditional vehicle acceleration distributions. For the maneuver prediction model, trained to minimize the NLL of the maneuver probabilities $-\log\left(\sum\nolimits_m P(m|\boldsymbol{X})\right)$ or the vehicle acceleration prediction, the adopted model is used to minimize $-\log\left(\sum\nolimits_m P_{\widetilde{\Theta}}(\boldsymbol{Y}|m, \boldsymbol{X}_{\mathrm{T}})\right)$.

3.2. Prediction-based stochastic FRS of a surrounding vehicle

When predicting the future state of a surrounding vehicle, both the current state and historical information need to be considered [29]. In this paper, we use the acceleration prediction results from Section 3.1 to dynamically update the state transition probability matrix at each time step.

The system state i of the surrounding vehicle is represented as a tuple with four discretized elements, including two-dimensional vehicle positions and velocities. The system input is expressed as a two-dimensional acceleration (a_1, a_2). The probability of the current state is known in advance. Typically, we have an initial state i with probability $\boldsymbol{p}_{\boldsymbol{i}}(0) = 1$, or an initial probability distribution is provided to address the state uncertainties. In practice, from the current time, we need to calculate multiple stochastic FRS at multiple forwarded time steps and check the corresponding FRS at each future time step $k \in \{1, 2, \cdots, e\}$.

At each predicted time step k, the acceleration prediction model provides a bivariate normal distribution function $f_m^k(a_1, a_2)$ for each maneuver mode m as follows:

$$f_m^k(a_1, a_2) = \frac{1}{2\pi\sigma_1\sigma_2\sqrt{1-\rho^2}}\exp\left(-\frac{1}{2\left(1-\rho^2\right)}\left[\left(\frac{a_1-\mu_1}{\sigma_1}\right)^2 + \left(\frac{a_2-\mu_2}{\sigma_2}\right)^2 - 2\rho\frac{\left(a_1-\mu_1\right)\left(a_2-\mu_2\right)}{\sigma_1\sigma_2}\right]\right) \tag{16}$$

where $\mu_1,\ \mu_2,\ \sigma_1,$ and σ_2, provided by the prediction model, denote predicted means and standard deviations along two directions, and the correlation at future time instant k for each maneuver mode m, respectively. For brevity, the time and maneuver indices of the five parameters are omitted.

To propagate the system states, we calculate the conditional probability $\boldsymbol{p}_i^u(k)$ at time step k under state i and accelerations $u = \left(a_1^u, a_2^u\right)$ as follows:

$$\boldsymbol{p}_i^u(k) = \frac{\widetilde{\boldsymbol{p}_i^u}(k)}{\sum_{u\in\mathcal{U}}\widetilde{\boldsymbol{p}_i^u}(k)} \tag{17}$$

$$\widetilde{\boldsymbol{p}_i^u}(k) = \sum\nolimits_m \lambda_m^k \cdot \int_{\bar{a}_2^u}^{\bar{a}_2^u}\int_{\bar{a}_1^u}^{\bar{a}_1^u} f_m^k(a_1, a_2)\mathrm{d}a_1\mathrm{d}a_2 \tag{18}$$

where λ_k^m is the probability for maneuver mode m at time step k, and $\bar{a}_1^u, \bar{a}_1^u, \bar{a}_2^u, \bar{a}_2^u$ are the integral boundaries of u.

The conditional state probability $\boldsymbol{p}_i^u(k)$ is implicitly relevant to the current and historical states. This is because current and historical information has been considered when providing the predicted acceleration results. This implies that the state transition matrix now has to be computed online.

Substituting Eqs. (17) and (6) into Eq. (7), we can obtain the overall state transition matrix $\boldsymbol{\Phi}$. To distinguish the Markov-based approach which can compute the transition matrix offline, we denote the state transition matrix obtained with the prediction model at the predicted time step k as $\boldsymbol{\Phi}(k)$. Subsequently, at each predicted time step, the state probability vector is iteratively computed as

$$\boldsymbol{p}(k+1) = \boldsymbol{\Phi}(k) \times \boldsymbol{p}(k) \tag{19}$$

To measure driving risk, the collision probability at the current time P_{col} is expressed as the product of the collision probability at each predicted time step.

$$\boldsymbol{P}_{\text{col}} = 1 - \prod_k\left(1 - \sum_{i\in\mathcal{H}(k)}\boldsymbol{p}_i(k)\right) \tag{20}$$

where $\mathcal{H}(k)$ is the set of states occupied by the ego vehicle at time step k. The vehicle dimensions are considered when calculating the collision probability.

3.3. Infusing confidence belief

A prediction model applied to compute the stochastic FRS is not always accurate because vehicles can move unexpectedly. To address this problem, we adopt a confidence-aware belief vector to modify probabilistic motion predictions that exploit the modeled structure when the structure successfully explains the vehicle motion and degrades gracefully whenever the vehicle moves unexpectedly [27].

If prediction confidence is not considered, the probabilistic acceleration is expressed by Eq. (16). To address the prediction confidence level, similar to the study in Fridovich-Keil et al. [27], we infuse an additional coefficient β into the acceleration probability density function:

$$f_m^k(a_1, a_2, \beta) = \frac{1}{2\pi(\sigma_1\beta)(\sigma_2\beta)\sqrt{1-\rho^2}}\exp\left(-\frac{1}{2(1-\rho^2)}\left[\left(\frac{a_1-\mu_1}{\sigma_1\beta}\right)^2 + \left(\frac{a_2-\mu_2}{\sigma_2\beta}\right)^2 - 2\rho\frac{(a_1-\mu_1)(a_2-\mu_2)}{\sigma_1\beta\sigma_2\beta}\right]\right) \tag{21}$$

where a positive coefficient β controls the confidence level of the prediction model. For instance, when setting β to infinity, the acceleration probability would be uniformly distributed and ignore the prediction model. When β is close to 0, the discretized input (a_1, a_2), which is closest to the predicted acceleration mean values μ_1, μ_2 is assigned with probability one.

Consequently, the input conditional probability computation in Eq. (18) can be extended as

$$\widetilde{\boldsymbol{p}^u}(k,\beta) = \sum_m\sum_\beta \lambda_m^k \boldsymbol{b}^k(\beta)\int_{\underline{a}_2^u}^{\bar{a}_2^u}\int_{\underline{a}_1^u}^{\bar{a}_1^u} f_m^k(a_1, a_2, \beta)\mathrm{d}a_1\mathrm{d}a_2 \tag{22}$$

where $b^k(\beta) \in (0, 1)$ is the belief value for a specific β at time step k.

Therefore, the performance of the prediction model may change over time. For instance, the model may have a relatively worse prediction accuracy when the surrounding vehicle begins to make a lane change, as the lane-change maneuver may not be recognized promptly. To reflect the dynamic properties of the confidence level, we should update the belief of each confidence level frequently. Thus, a Bayesian belief vector regarding possible values of β is introduced.

Initially, each β is assigned with a uniform probability $b^0(\beta)$; subsequently, the belief vector evolves given posterior probabilities of the state and input under each β at time step k:

$$b^{k+1}(\beta) = \frac{P\left(u^{(k-k'+1:k)}|x^{(k-k'+1:k)};\Theta^{(k-k'+1:k)},\beta\right)b^k(\beta)}{\sum_{\widetilde{\beta}}P\left(u^{(k-k'+1:k)}|x^{(k-k'+1:k)};\Theta^{(k-k'+1:k)},\widetilde{\beta}\right)b^k\left(\widetilde{\beta}\right)} \tag{23}$$

where

$$P\left(u^{(\text{ss}:k)}|x^{(\text{ss}:k)};\Theta^{(\text{ss}:k)},\beta\right) = \prod_{i\,=\,k-k'+1}^{k}\widetilde{\boldsymbol{p}^{u_i}}(k-i,\beta) \tag{24}$$

is the posterior probability of the observed actual accelerations, $u^{(\text{ss}:k)}$ with $t_{\text{ss}} = t_{\max\left(k-k'+1,1\right)}$ from the previous k' time steps.

In practice, the Bayesian belief vector with a relatively small set and small previous time steps, for instance, five discrete values of β and $k' = 2$, can achieve significant improvement [27].

4. Integrated collision–detection framework

In this section, we propose an integrated collision–detection framework that combines the BRS and stochastic FRS described in the previous section. Hence, we also specify the system dynamics models for the BRS and stochastic FRS and prove the equivalence of different models in the BRS and stochastic FRS.

4.1. Framework

We propose an integrated collision–detection framework for highways by combining the BRS and stochastic FRS. Before employing this framework, a BRS is computed using the HJI PDE [32]. Although computing a BRS is time-consuming, this can be done offline, and the results of the BRS are cached in a look-up table for subsequent real-time risk assessment. Note that we are required to interpolate the BRS look-up table in practice as the cached states are discretized. A flowchart of the framework, including the three iterative steps, is shown in Fig. 3.

Step 1: We check whether the surrounding vehicles are inside the unsafe area identified by the BRS. If so, we proceed to the next step; otherwise, the safety of vehicle interactions is theoretically ensured at the current time.

Step 2: A stochastic FRS is generated online to calculate the collision probability at each predicted time step. Note that the stochastic FRS can be obtained using either heuristic rules [25] or the prediction-based approach introduced in Section 3.

Step 3: If the obtained collision probability is above a predefined threshold, the ego vehicle must replan its motion trajectories or the ego driver receives an alert to avoid potential crashes with the surrounding vehicle, which is not within the scope of this paper.

4.2. Vehicle dynamics and equivalent transformation between BRS and stochastic FRS

The computation of the BRS and (stochastic) FRS depends on the selection of the vehicle dynamic models. Ideally, we can use the same system dynamics for the ego to construct both the BRS and FRS, and the equivalent transformation, which is used to ensure a consistent control input range between different vehicle dynamics, is no longer necessary. However, in this paper, we use different system dynamics for BRS and FRS for two reasons. First, we construct the BRS using a bicycle model for the ego vehicle and a unicycle model for the surrounding vehicle following Li et al. [32] because these models are relatively more realistic than a point mass model. Second, we construct the FRS using a point mass model primarily to accommodate the control input predictors, which typically output two-dimensional acceleration values [31,33]. The details of the vehicle dynamics employed in this paper are as follows:

To calculate the BRS, we define the vehicle dynamics of the ego and surrounding vehicles according to Li et al. [32]. A more complex ego vehicle model increases the dimensions of the relative dynamics model, resulting in an overly complicated calculation of the BRS. In the study of Leung et al. [8], computing the BRS using a seven dimensional relative system and discretization required approximately 70 h on a 3.0 GHz octa-core advanced micro devices (AMD) Ryzen 1700 central processing unit (CPU). In this paper, collision detection is guaranteed by a combination of BRS and stochastic FRS. The vehicle models used in the two modules (BRS and stochastic FRS) must be consistent. A high-dimensional vehicle model in the BRS significantly increases the computational load of the stochastic FRS, even with a parallel computation assumption. Therefore, as shown in Fig. 4, we apply a bicycle model for the ego vehicle, where the state includes longitudinal/lateral positions $\boldsymbol{y}_{1,\text{ego}}/\boldsymbol{y}_{2,\text{ego}}$, the heading angle ψ_{ego}, and the ego vehicle velocity $\boldsymbol{v}_{\text{ego}}$. Its control input is the acceleration aego and steering angle $\boldsymbol{\delta}_f$. O is the center of vehicle rotation. $\boldsymbol{\delta}_r = 0$ is the rear wheel steering angle and $\boldsymbol{\beta}_{\text{ego}} = \tan^{-1}\left(\frac{\boldsymbol{l}_r}{\boldsymbol{l}_f + \boldsymbol{l}_r}\tan\boldsymbol{\delta}_f\right)$ is the slip angle of the ego vehicle, where $\boldsymbol{l}_f/\boldsymbol{l}_r$ is the distance from the front/rear to the vehicle reference point. For the surrounding vehicle, we model its state with two-dimensional positions $\frac{\boldsymbol{y}_{1,\text{s}}}{\boldsymbol{y}_{2,\text{s}}}$, the heading angle ψ_{s}, and the surrounding vehicle velocity $\boldsymbol{v}_{\text{s}}$. The control input is acceleration a_{s} and angular velocity $\boldsymbol{\omega}_{\text{s}}$. Subsequently, the relative dynamics can be represented as

$$\begin{cases} \dot{\boldsymbol{y}}_{1,\text{rel}} = \frac{\boldsymbol{v}_{\text{ego}}}{l_r}\sin\beta_{\text{ego}}\boldsymbol{y}_{2,\text{rel}} + \boldsymbol{v}_{\text{s}}\cos\psi_{\text{rel}} - \boldsymbol{v}_{\text{ego}}\cos\boldsymbol{\beta}_{\text{ego}} \\ \dot{\boldsymbol{y}}_{2,\text{rel}} = -\frac{\boldsymbol{v}_{\text{ego}}}{l_r}\sin\boldsymbol{\beta}_{\text{ego}}\boldsymbol{y}_{1,\text{rel}} + \boldsymbol{v}_{\text{s}}\sin\psi_{\text{rel}} - \boldsymbol{v}_{\text{ego}}\sin\boldsymbol{\beta}_{\text{ego}} \\ \dot{\psi}_{\text{rel}} = \boldsymbol{\omega}_{\text{s}} - \frac{\boldsymbol{v}_{\text{ego}}}{l_r}\sin\boldsymbol{\beta}_{\text{ego}} \\ \dot{\mathbf{s}}_{\text{ego}} = \boldsymbol{a}_{\text{ego}} \\ \dot{\boldsymbol{v}}_{\text{s}} = \boldsymbol{a}_{\text{s}} \end{cases} \tag{25}$$

where $\boldsymbol{y}_{1,\text{ego}}/\boldsymbol{y}_{2,\text{ego}}$ and ψ_{rel} are relative two-dimensional coordinates and heading angles, respectively.

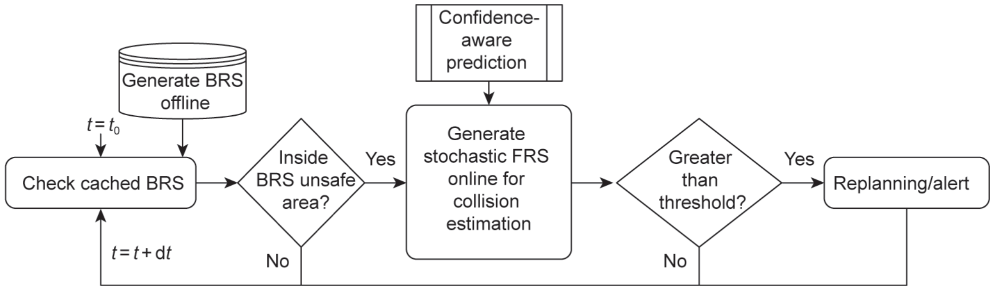


Fig. 3. Flowchart of the integrated collision–detection framework.

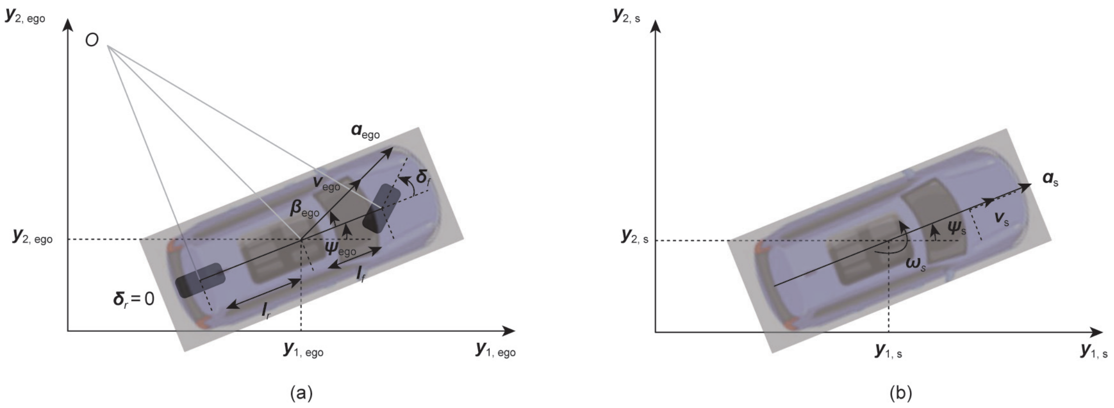


Fig. 4. Vehicle dynamic systems in the BRS. (a) Ego vehicle dynamic model with inputs: acceleration ($\boldsymbol{a}_{\text{ego}}$) and steering angle ($\delta_f$). (b) Surrounding vehicle dynamic model with inputs: acceleration ($\boldsymbol{a}_{\text{s}}$) and angular acceleration (ω_{s}).

To calculate the stochastic FRS and accommodate acceleration prediction models that typically represent probabilistic accelerations with bivariate distributions, we employ a point-mass model to compute the stochastic FRS of the surrounding vehicle. The control input is simplified with two-dimensional accelerations, and future vehicle positions can then be directly propagated with the predicted accelerations. We assume that the planned trajectories of the ego vehicle are deterministic and are known in advance. Nevertheless, the motion uncertainties of an ego vehicle can be represented by extending its future position with additional adjacent states, which is left for future research.

A summary of the system dynamics used for BRS and FRS is provided in Table 1. Given the different vehicle dynamic models and control inputs employed for the BRS and stochastic FRS, it is essential to match the control input ranges among the three dynamic models. We term this process the equivalent vehicle dynamics transformation between the BRS and FRS. We provide a detailed equivalent transformation procedure in the Supplementary text in Appendix. A. The BRS and stochastic FRS can now be integrated into the same framework.

5. Experiments

In this section, we first introduce the naturalistic highway driving dataset highD employed for the prediction model training/testing as well as the experimental setup for the prediction model and BRS/FRS computation. Several experiments were designed and conducted with respect to three aspects: ① We tested the proposed acceleration prediction model by comparing it with three existing predictors, as the acceleration prediction model plays a vital role in the establishment of the stochastic FRS. ② We further tested the performance of the prediction-based confidence-aware FRS using naturalistic driving data from highD and simulated cut-in events. Risky cut-in events were simulated to test the collision estimation performance based on stochastic FRS. ③ To validate the integrated collision–detection framework, we first compared the identified unsafe areas between the BRS and FRS and then tested the framework for both risky and non-risky cut-in events.

5.1. Dataset and setup

The highD dataset [34], which contains bird–view naturalistic driving data on German highways, was used to train and test the acceleration prediction model. We randomly selected equal numbers of samples for the three lane-change maneuver modes, resulting in 135 531 (45 177 for each maneuver mode) and 19 482 (6494 for each mode) samples for training and testing, respectively. The original dataset sampling rate was 25 Hz, and we down sampled it by a factor of five to reduce the model complexity. We considered 2 s historic information as input and predicted within a 2 s horizon.

The prediction model was trained using Adam with a learning rate of 0.001, and the sizes of the encoder and decoder are 64 and 128, respectively. The size of the fully connected layeris 32. The convolutional social pooling layers consist of a 3 × 3 convolutional layer with 64 filters, a 3 × 1 convolutional layer with 16 filters, and a 2 × 1 max pooling layer, which reconsistent with the settings in Ref. [29].

Corresponding with the vehicle dynamics in Section 4.2, the BRS was computed offline in five dimensions: relative two-dimensional positions, heading angle, and velocities for the ego and surrounding vehicles, respectively. The relative longitudinal position was discretized from −10 to 40 m in 0.5 m increments, the relative lateral position from −4 to 4 m in 0.4 m increments, the heading angle from −45° to 45° in 9° increments, and ego/surrounding vehicle velocity from 20 to 40 m·s^{-1} in 1 m·s^{-1} increments, resulting in over five million states. The longitudinal position range setting was sufficient to identify an enclosed unsafe area for all simulations in this study, as shown in Fig. 14. The range of the control input is given in the Supplementary text to ensure an equivalent model transformation between BRS and FRS.

The FRS states were expressed in four dimensions, including two dimensions for position and velocities, respectively. The vehicle longitudinal position was discretized from −4 to 80 m in 2 m increments and the relative lateral position from −4 to 4 m in 1 m increments. The longitudinal velocity was discretized from 20 to 40 m·s^{-1} in 0.4 m·s^{-1} increments, and the lateral velocity from −2.5 to 2.5 m·s^{-1} in 0.2 m·s^{-1} increments, resulting in approximately half a million states. Here, the longitudinal position range of the FRS was from −4 to 80 m because we assumed that a vehicle could not move backward, and the marginal position to collide with a surrounding vehicle was −4 m, given the vehicle length at 4 m. It reached a maximum of 80 m with the highest longitudinal velocity (40 m·s^{-1}) in the prediction horizon (2 s). For the control input, we discretized the longitudinal (lateral) accelerations from −5 to 3 (−1.5 to 1.5) m·s^{-2} in increments of 1 (0.5) m·s^{-2}, resulting in 63 acceleration combinations. We also added several constraints to limit acceleration selection, including maximal acceleration, strict forward motion, and maximal steering angle [17]. Ultimately, 37 million possible state transfers were generated. To alleviate the computational load, we assumed that an advanced graphics processing unit (GPU) [35] that enables 2048 × 28 parallel computations is available. The stochastic FRS with state probability distributions $\boldsymbol{p}(k)$ was calculated at each predicted future time step within 2 s in 0.4 s increments, for instance {0.4, 0.8, 1.2, 1.6, 2.0}.

5.2. Acceleration/trajectory prediction models

To validate the proposed two-stage acceleration prediction model (denoted as T-LSTMa), we compared T-LSTMa with three state-of-the-art probabilistic multi-modal predictors: T-LSTM [30], social convolutional trajectory predictor (S-LSTM) using convolutional neural networks to represent surrounding vehicles [29], and its variation S-LSTMa for fair comparisons, where the prediction output was modified to probabilistic accelerations. Utilizing the testing dataset from highD, we report the comparative results in Table 2, including five evaluation indicators, for instance, root mean square error (RMSE), average displacement error (ADE), final displacement error (FDE), NLL (the lower, the better), and average lane-change prediction F1 score LC-F1 (a metric calculated as the

Table 1
System dynamics to construct the BRS and stochastic FRS.

Type of reachable set	Ego	Surrounding
BRS	Bicycle model with input: acceleration and heading angle	Unicycle model with input: acceleration and steering angle
FRS	—	Point mass model with input: two-dimensional accelerations

Table 2
Prediction model performance on the highD testing dataset.

Lane-change maneuver mode	Indicators	T-LSTMa	S-LSTMa	T-LSTM	S-LSTM
Lane-keeping	RMSE (m)	0.17	0.42	0.46	0.59
	ADE (m)	0.10	0.31	0.26	0.27
	FDE (m)	0.26	0.81	0.61	0.72
	NLL (m)	−3.47	0.91	−1.59	−1.38
	LC-F1 (%)	98.70	0.00	98.70	96.94
Turning left	RMSE (m)	0.24	0.56	0.52	0.62
	ADE (m)	0.14	0.38	0.35	0.36
	FDE (m)	0.37	0.96	0.83	0.89
	NLL (m)	−2.80	−0.33	−0.63	−0.66
	LC-F1 (%)	99.64	87.04	99.64	97.85
Turning right	RMSE (m)	0.23	0.29	0.49	0.58
	ADE (m)	0.14	0.17	0.33	0.37
	FDE (m)	0.37	0.45	0.74	0.86
	NLL (m)	−2.97	−2.25	−0.74	−0.46
	LC-F1 (%)	99.80	99.95	99.80	99.34
Overall	RMSE (m)	0.22	0.44	0.49	0.60
	ADE (m)	0.13	0.29	0.32	0.86
	FDE (m)	0.34	0.74	0.73	0.86
	NLL (m)	−3.07	−0.58	−0.97	−0.82
	LC-F1 (%)	98.79	859.20	98.79	96.18

harmonic mean of the precision and recall [36]). We show the predictor performance on the overall testing dataset and compare the prediction results in terms of the three lane-change maneuver modes. Note that we did not directly evaluate the acceleration prediction accuracy. Instead, we compared the vehicle position prediction accuracy, which directly affects collision estimation.

Examining the overall comparison results, we observed that using acceleration prediction and then propagating the future vehicle position can significantly improve the prediction accuracy (see comparisons between S-/T-LSTMa and S-/T-LSTM). This could be owing to the additional physical information when using the predicted acceleration to propagate future positions. Meanwhile, S-/T-LSTMa and S-/T-LSTM had the same LC-F1 because they share the same lane-change prediction submodel. S-LSTMa exhibited the worst performance in terms of NLL and LC-F1. Specifically, the LC-F1 for lane-keeping trajectories was 0. This indicated that jointly predicting the acceleration and lane-change maneuver modes in one neural network results in undesirable results, and it is reasonable to consider decoupling the acceleration and lane-change maneuver mode predictions using two neural networks, which have been employed in T-LSTMa. In summary, our proposed acceleration predictor T-LSTMa achieved the best performance in terms of all indicators, with ADE < 0.15 m and FDE < 0.40 m.

We further tested the model's prediction performance over different prediction horizons (one and three seconds). As shown in Table 3, the proposed predictor, T-LSTMa, achieved superior performance over all prediction horizons. In addition, both models obtained more accurate prediction results with shorter prediction horizons. For the remainder of this study, we used T-LSTMa with default settings to predict accelerations.

5.3. Confidence-aware position prediction and collision estimation

In this section, we compare different approaches for generating a stochastic FRS. Three different groups of coefficients β were selected: Only one distribution with β = [1]; three normal distributions with β = [1/2, 1, 2]; five normal distributions with β = [1/3, 1/2, 1, 2, 3]. This led to three different approaches to generating prediction-based stochastic FRS (denoted as PSRS, PSRS-3β, and PSRS-5β). A further increase in the number of distribution groups was not considered because we were required to ensure real-time computation of the stochastic FRS. The heuristic method in the study of Althoff et al. [25] was also adopted as a baseline to generate the stochastic FRS (denoted as HSRS).

To test the performance of the prediction-based confidence-aware FRS, we used naturalistic driving data from highD and simulated cut-in events. This is because highD does not contain safety–critical events; thus, we created simulated risky events to test the collision estimation performance based on the stochastic FRS.

5.3.1. highD trajectories

We assumed that the vehicle occupied the exact space state and the four adjacent states along the longitudinal and lateral directions. Naturalistic driving trajectories were randomly selected from highD, including lane-keeping and lane-change trajectories.

Table 3
Prediction model performance over prediction horizon 1 and 3 s.

Time (s)	Indicators	T-LSTMa	S-LSTMa	T-LSTM	S-LSTM
1	RMSE (m)	0.06	0.13	0.15	0.19
	ADE (m)	0.04	0.09	0.14	0.15
	FDE (m)	0.08	0.20	0.25	0.28
	NLL (m)	−4.79	−2.46	−2.31	−2.17
	LC-F1 (%)	98.90	65.98	98.90	97.86
3	RMSE (m)	0.47	0.86	0.79	0.85
	ADE (m)	0.26	0.56	0.48	0.53
	FDE (m)	0.77	1.52	1.32	1.48
	NLL (m)	−1.90	1.11	−0.26	−0.02
	LC-F1 (%)	94.71	52.18	94.71	91.00

Examples of stochastic FRS predicted using different approaches are presented in Figs. 5 and 6, respectively. For the lane-keeping trajectory, the position prediction accuracies at time t = 4.4 s for the four approaches were 44.33%, 45.56%, 46.84%, and 52.74%. For the lane-change trajectory, the position prediction accuracies at time t = 6.4 s for the four approaches were 24.54%, 29.38%, 32.58%, and 37.15%. Infusing confidence awareness can improve vehicle position prediction, particularly for lane-changing trajectories. The belief vector changes are shown in Fig. 7 for both lane-keeping and lane-change trajectories. For the lane-keeping trajectory, Figs. 7(a) and (b) show corresponding results from PSRS-3β and PSRS-5β; both had a stable acceleration prediction accuracy, as the surrounding vehicle moved as expected. Moreover, the belief value with the lowest β converged to one for both two approaches. This is because the confidence-infused prediction model provided a more accurate mean value of accelerations, resulting in a corresponding higher belief value with lower β in line with Eq. (24). For the lane-change trajectory, a similar convergence trend of the belief value changes can be observed in Figs. 7(c) and (d). However, the belief value of the lowest β (Fig. 7(d), PSRS-5β) required about 4 s to converge and less than 2 s in Fig. 7(c). This indicates that infusing a higher-dimensional belief vector can adjust more complicated prediction scenarios (e.g., lane-change trajectories) and provide more accurate prediction results.

We tested the four approaches using randomly selected 100 trajectories from highD for both lane-keeping and lane-change scenarios. The average position prediction performance is summarized in Table 4, which shows that when the prediction horizon is short (e.g., 0.4 s), all approaches could generate a stochastic FRS with higher prediction accuracy. However, when

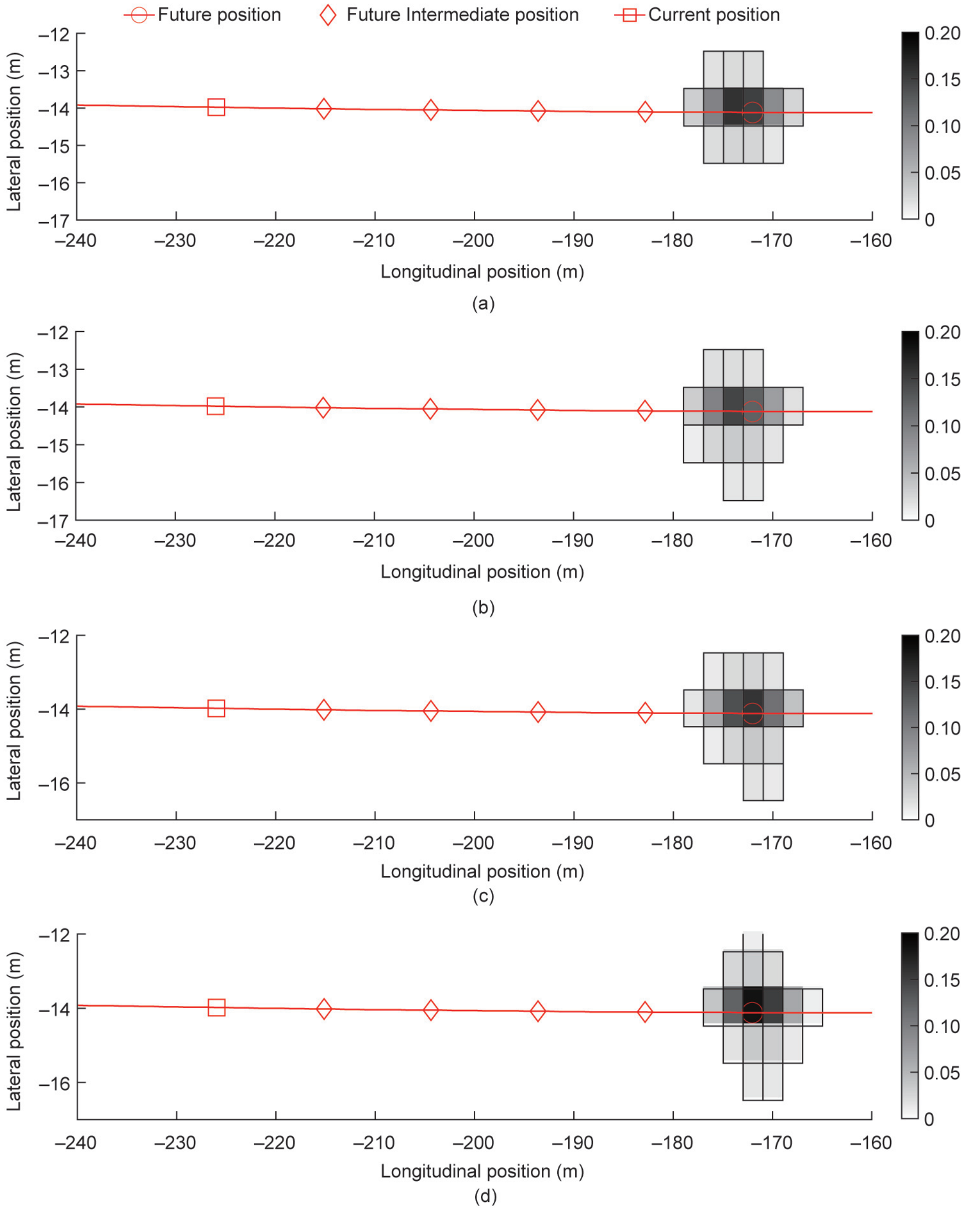


Fig. 5. State probability distributions of the stochastic FRS using different approaches for a lane-keeping trajectory (current time t = 4.4 s). Only position states with > 0.01 probability are displayed. (a) HSRS, (b) PSRS, (c) PSRS-3β, (d) PSRS-5β.

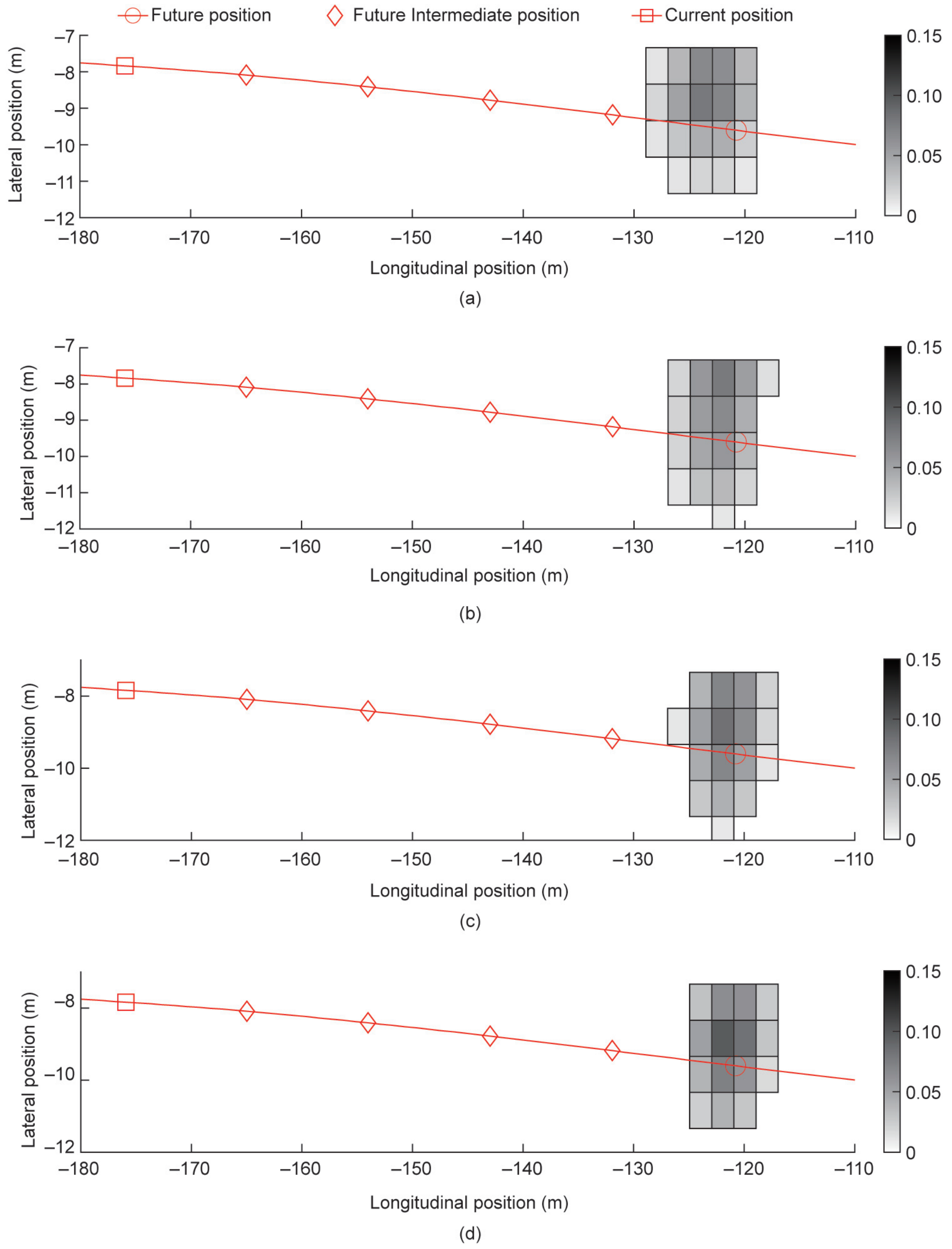


Fig. 6. State probability distributions of stochastic FRS using different approaches during a lane change (current time t = 6.4 s). Only position states with > 0.01 probability are displayed. (a) HSRS, (b) PSRS, (c) PSRS-3β, (d) PSRS-5β.

the prediction horizon increased, HSRS exhibited the worst performance, followed by PSRS. Again, infusing confidence belief effectively improves prediction accuracy; such an improvement is more significant in lane-change scenarios.

5.3.2. Simulated safe-critical cut-in trajectories

Safety-critical cut-in events were simulated to test the collision–detection performance of different stochastic FRS. In the simulated cut-in scenario, the ego vehicle in the middle lane and the surrounding vehicle in the right lane traveled with constant initial longitudinal speeds of $\boldsymbol{v}_{\text{ego}}$ = 30 m·s^{-1} and $\boldsymbol{v}_{\text{s}}$ = 25 m·s^{-1}, respectively. The surrounding vehicle was 15 m ahead of the ego vehicle at t = 1.0 s and began turning left with a lane-change duration of 7.5 s. The vehicle length and width were 4 and 2 m, respectively. The driving behaviors of the two vehicles were simulated using the classic intelligent driving model (IDM) for car-following and a lateral control model [37].

The IDM equations are expressed as

$$\begin{cases} \mathrm{dg}^*(t) = \mathrm{dg}_0 + \max(0, \boldsymbol{v}(t)\mathrm{TH} + \boldsymbol{v}(t)\mathrm{d}\boldsymbol{v}(t)/2\sqrt{a_a a_b}) \\ a(t) = \max\left(a_a\left(1 - \left(\frac{\boldsymbol{v}(t)}{\boldsymbol{v}_0}\right)^4 - \left(\frac{\mathrm{dg}^*(t)}{\mathrm{dg}(t)}\right)^2\right), a_{\min}\right) \end{cases} \tag{26}$$

where $\mathrm{dg}^*(t)$ and $\mathrm{dg}(t)$ are the desired and current longitudinal distance gaps, respecitvely; $\boldsymbol{v}(t)$ is the longitudinal speed; $\mathrm{d}v(t)$ is the longitudinal speed difference from the lead; $a(t)$ the current accelerration. If there is no leading vehicle, $\mathrm{d}\boldsymbol{v}(t)$ and $\frac{1}{\mathrm{dg}(t)}$ are set to zero. The IDM parameters are the longitudinal desired speed $(\boldsymbol{v}_0)$, time headway (TH), minimum gap (dg_0), acceleration coefficients (a_a), and deceleration (a_b), respectively.

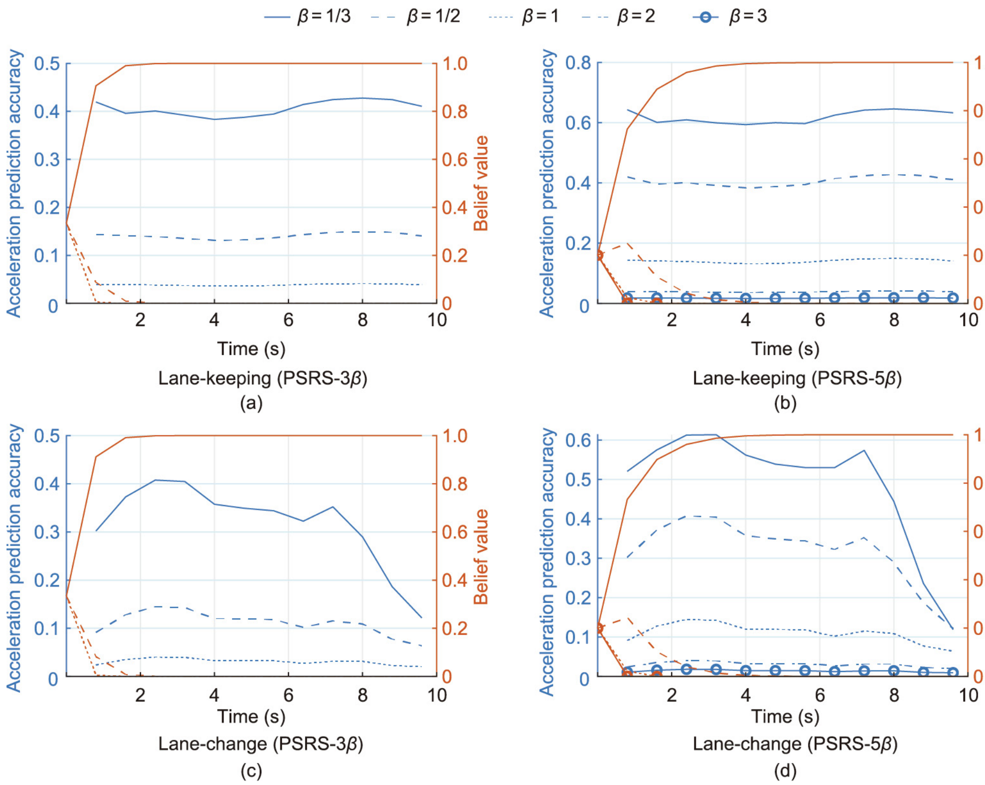


Fig. 7. Belief dynamic changes using naturalistic driving data under different coefficient settings. (a) and (b) are for a lane-keeping trajectory, whereas (c) and (d) are for a lane-change trajectory. The blue lines represent the change in the acceleration prediction accuracy with different coefficient settings, and the orange lines indicate the corresponding dynamic belief values.

Table 4
Average position prediction accuracy (%) using four stochastic FRS with different prediction times.

Time (s)	Lane-change maneuver mode	HSRS	PSRS	PSRS-3β	PSRS-5β
2.0	lane-keeping	43.08	45.44	48.64	52.81
	lane-change	26.81	30.47	34.9	38.58
1.2	lane-keeping	79.45	78.38	80.23	84.01
	lane-change	50.01	50.48	51.69	55.63
0.4	lane-keeping	100	100	100	100
	lane-change	87.08	86.22	86.95	88.39

These IDM parameters were adopted from Kurtc [38] based on the car-following trajectories in highD. The ego vehicle had the desired speed as its initial longitudinal speed $\boldsymbol{v}_{\text{ego}}$, and the desired speed of the surroundings was set to 36.1 m·s^{-1} (130 km·h^{-1}). The two vehicles shared the same values for the remaining IDM parameters: TH = 0.8 s, $\mathrm{dg}_0 = 6.0$ m, a_a = 1.0 m·s^{-2}, a_{ab} = 1.0 m·s^{-2}. For the lateral cut-in behaviors, we adopted the polynomial curves according to Mullakkal-Babu et al. [37], which can provide lateral accelerations with smooth trajectories.

Given these cut-in scenario settings, a safety–critical cut-in event was created and a crash occurred at approximately t = 5.0 s. As shown in Fig. 8, the space state probabilities of the stochastic FRS were more concentrated using PSRS-3β and PSRS-5β. The future position prediction accuracies at this time were 31.81%, 33.4%, 36.73%, and 37.39% using HSRS, PSRS, PSRS-3β, and PSRS-5β, respectively. The belief dynamic updates for the simulated cut-in event are shown in Fig. 9. By employing PSRS-3β in Fig. 9(a), the belief value of the lowest β shortly converged to 1 in 2 s. As shown in Fig. 9(b), the acceleration prediction accuracy based on confidence levels β = 1/3 and β = 1/2 decreased to the same low value at t = 2.4 s. The difference in acceleration prediction accuracy became evident between β = 1/3 and β = 1/2, resulting in a belief value convergence for β = 1/3 later. Different confidence level settings and changes in the dynamic belief value eventually resulted in a better performance of PSRS-5β compared with PSRS-3β.

The collision detection results are shown in Fig. 10. In the beginning (from 0 to 1.2 s), the four FRS-based approaches all predicted a small collision probability because the surrounding vehicle does not start a lane change until t = 1 s. Subsequently, significant differences in collision estimation were observed for the four approaches. For instance, when the estimated collision probability using PSRS-5β was close to 0.20 at time t = 2.4 s, PSRS without infusing confidence awareness predicted the collision probability as 0.10, and HSRS had a predicted collision probability of approximately 0.05. This indicates that the stochastic FRS using confidence-aware prediction is agile and effective in identifying potential collisions in a risky cut-in event.

We also simulated a safety–critical cut-in event that did not result in a crash by changing the initial longitudinal speed of the

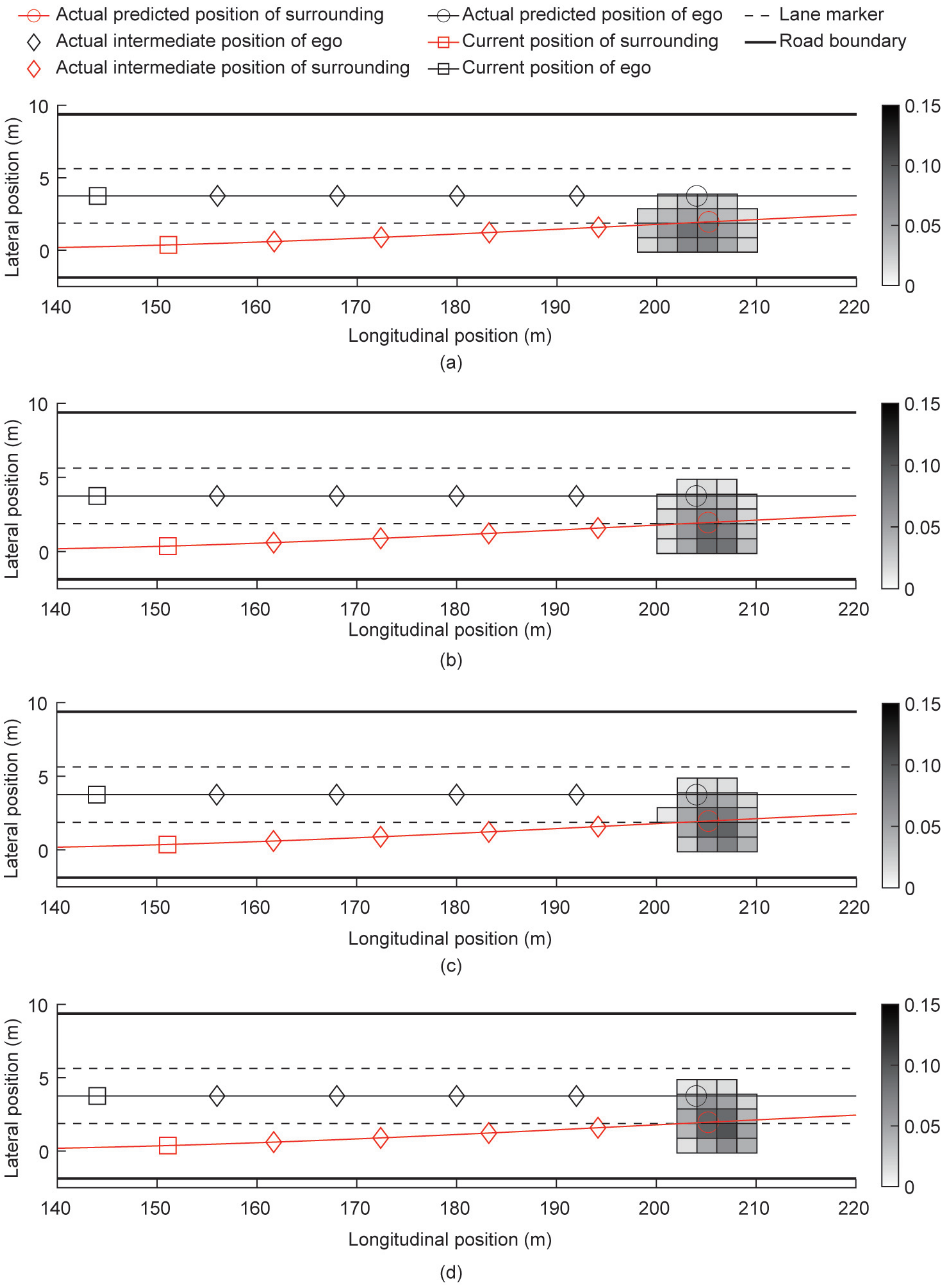
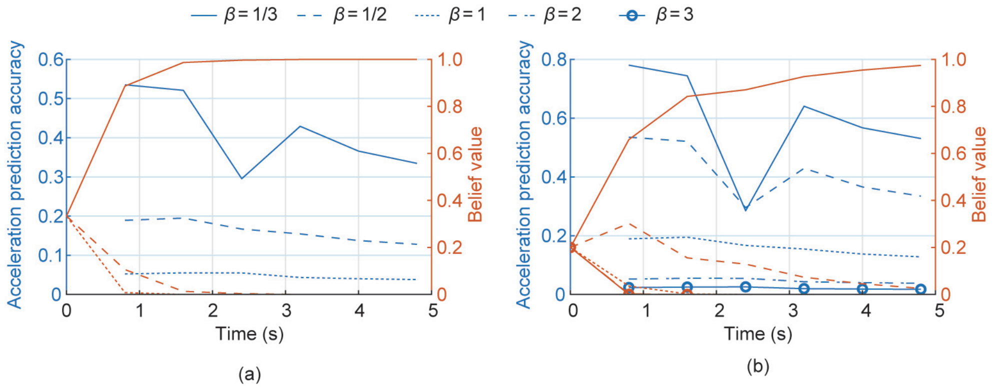


Fig. 8. State probability distributions of the stochastic FRS using different approaches in the simulated cut-in event (current time t = 2.4 s). Only position states with > 0.01 probability are displayed. (a) HSRS, (b) PSRS, (c) PSRS-3β, (d) PSRS-5β.

surrounding vehicle to 27 m·s^{-1}. The increased longitudinal velocity of the cut-in vehicle resulted in a larger longitudinal gap when it entered the target lane, resulting in a safety–critical but collision-free cut-in event. During the cut-in process, the belief dynamic changes shown in Fig. 11 had a very similar trend to that in the cut-in crash event because the two events had the same parameter settings, except for the longitudinal speed difference. Fr the predicted collision probability shown in Fig. 12, the heuristic-based reachable set approach HSRS predicted the highest collision probability compared with the proposed prediction-based approach PSRS. Specifically, HSRS reached a maximum collision probability greater than 0.4 at t = 4 s, which corresponded to the time when the surrounding cut-in vehicle crossed the lane marker. This was because HSRS could not accurately predict the future positions of the surrounding vehicles and incorrectly estimated the high collision probability between two vehicles. The simplified heuristic approach, HSRS, also calculated a collision probability of 0.07 when the surrounding vehicle completed the cut-in process and became the leader of the ego vehicle in the target lane. Among the three PSRS approaches with different confidence awareness coefficients, we observed that the maximum collision probability was obtained at the critical time t = 4 s. Without the infusion of confidence awareness, PSRS calculated a collision probability of 0.16, whereas PSRS-3β/-5β obtained a smaller collision probability 0.07/0.03, owing to a more concentrated stochastic FRS. Given that the simulated cut-in event was safety–critical but collision-free, the PSRS approach accurately captures the potential collision risk with a smaller collision probability than the HSRS. In particular, the estimated collision probability by PSRS-5β was always below the predefined threshold of 0.05, indicating that the



(a) (b)

Fig. 9. Belief dynamic changes for the simulated cut-in event resulting in a crash at t = 5.0 s. The blue lines represent the change in acceleration prediction accuracy with different coefficient settings, and the orange lines indicate the corresponding dynamic belief values. (a) PSRS-3β, (b) PSRS-5β.

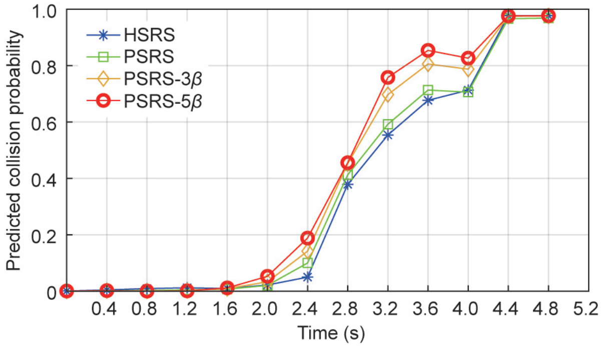


Fig. 10. Collision probability estimation for a cut-in crash event using different approaches. The initial longitudinal speeds of the ego and surrounding vehicles were 30 and 25 m·s^{-1}, respectively. All four approaches successfully detected the crash at t = 5.0 s. The estimated collision probability by PSRS-5β had the most agile and effective performance owing to the employed confidence-aware stochastic FRS.

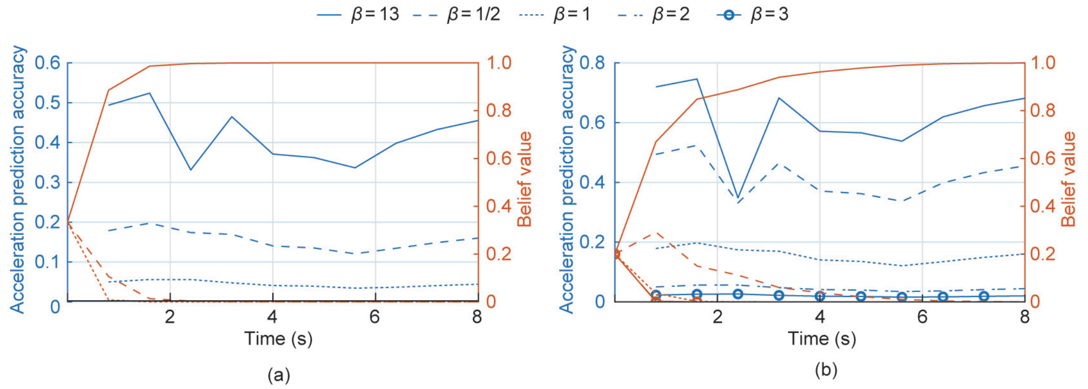
confidence-infused prediction approach PSRS-5β could not only accurately identify risky cut-in crash but also avoid false–positive results in a safety–critical but collision-free event.

Under the same simulated cut-in scenario settings, we varied the initial longitudinal speeds of the two vehicles from 20 to 35 m·s^{-1} in increments of 1 m·s^{-1}, resulting in 16 × 16 = 256 cut-in events. Based on the employed longitudinal and lateral driving behavior models, 33 crashes were identified when $4 \le \boldsymbol{v}_{\text{ego}} - \boldsymbol{v}_{\text{s}} \le 6$ m·s^{-1}. To compare the collision detection performance, we statistically analyzed the simulated 21 cut-in events in which a crash occurred. The analysis results are shown in Fig. 13, where the timeliness value represents the average time required to crash when the collision probability reaches a threshold. Regardless of the selection of the collision probability threshold, our proposed approaches infusing confidence awareness (i.e., PSRS-5β/PSRS-3β) achieved a larger timeliness value, indicating a more adequate reaction time to potential crashes. The selection of a suitable collision probability threshold can vary in different scenarios, which we leave for future research.

5.4. Integrated collision–detection framework

We propose an integrated collision–detection framework based on BRS and stochastic FRS. First, we provide the comparative results between the two reachable set techniques: BRS and FRS. This explains the selection of the BRS rather than the FRS to formally check driving safety in the first step of the framework.

As shown in Fig. 14, we set the longitudinal speed to 30 and 28 m·s^{-1} for the ego and surrounding vehicles, respectively, and illustrate comparative results using both the BRS and FRS as an example. The BRS was directly employed to identify unsafe areas when its cached state value function was less than zero. To obtain the unsafe area identified by the FRS, we first enumerated the initial relative surrounding vehicle positions and checked whether the two vehicles could collide using the FRS with a prediction horizon. We then enclosed all the relative positions that could result in a crash and identified the enclosed area in the blue line as unsafe. Note that we limited the unsafe area between −3.75 to 3.75 m in



(a) (b)

Fig. 11. Belief dynamic changes for the simulated safety–critical cut-in event that did not result in a crash. The blue lines represent changes in acceleration prediction accuracy with different coefficient settings, and the orange lines indicate the corresponding dynamic belief values. (a) PSRS-3β, (b) PSRS-5β.

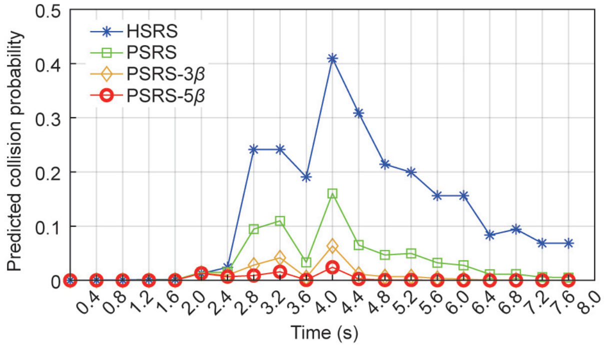


Fig. 12. Collision probability estimation for a safety–critical but collision-free cut-in event using different approaches. The initial longitudinal speeds of the ego and surrounding vehicles were 30 and 27 m·s^{-1}, respectively. All four approaches obtained a maximum collision probability at t = 4.0 s when the cut-in vehicle was about to cross the lane marker. The estimated collision probability by PSRS-5β was always below the predefined risk threshold of 0.05.

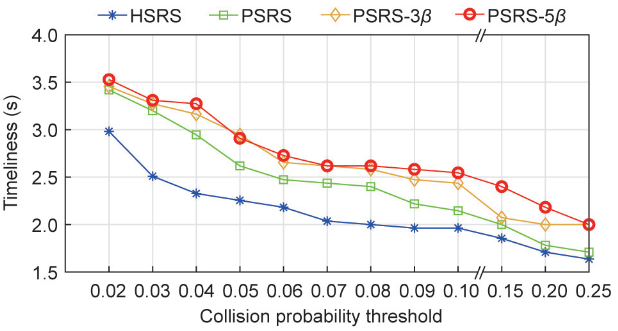


Fig. 13. Average timeliness performance under different collision thresholds. The timeliness value represents the time that remained to the crash when the estimated collision probability reached a threshold.

the lateral direction because we were required to address only potential risky interactions between adjacent lanes. Similarly, we did not check relative longitudinal positions behind the ego vehicle. As shown in Fig. 14, the FRS identified a larger unsafe area than the BRS. This was reasonable because the BRS further considered the ego reaction to the surrounding vehicle, resulting in a smaller unsafe area. Specifically, the identified unsafe areas were symmetric (Fig. 14(a)) because the lateral speed and relative heading angles of the initial vehicle states were both zero. In contrast, the surrounding vehicle had an initial lateral speed and a relative heading angle, as shown in Fig. 14(b). Subsequently, based on the FRS, we observed that the area with a lateral position between −3.00 to −3.75 m was now safe owing to the lateral speed of the surrounding vehicle. The BRS had an asymmetric shape owing to the lateral speed, and the area with a negative lateral relative position became safer. This example verifies that the BRS is less conservative than the FRS because the unsafe area identified by the BRS is smaller than that identified by the FRS, and the BRS-identified unsafe area is a subset of the FRS-identified area. This observation is consistent with the definitions of BRS and FRS.

We then provided two specific cut-in events to validate the proposed collision–detection framework. The cut-in events were selected from the simulated trajectories described in Section 5.3.2. During the first cut-in event, the initial longitudinal speeds of the ego and its surroundings were 30 and 28 m·s^{-1}, respectively. This led to non-risky and safe interactions between the two vehicles. As shown in Fig. 15, during a lane change, the integrated framework ensured safety because the relative positions (Figs. 15(a) and (b)) were not inside the unsafe area identified by the BRS. Thus, the establishment of the stochastic FRS was not activated, and the estimated collision probability always remained zero (Fig. 15(c)).

In the second cut-in event, the initial longitudinal speeds of the ego and its surroundings were 30 and 25 m·s^{-1}, respectively, resulting in a collision at approximately t = 5 s, as shown in Fig. 16. When we used the proposed collision–detection framework for this safety–critical event, safety could be ensured at the beginning (green squares in Fig. 16). This was because the surrounding vehicle began the lane-change maneuver at t = 1.0 s, and the driving risk can only be captured until the surrounding vehicle had a clear lane-change intention. Thus, at t = 2.0 s, the proposed framework could no longer ensure driving safety (Fig. 16(c), where the surrounding vehicle was inside the BRS unsafe area of the ego), and a stochastic FRS was established to accurately estimate the collision probability. However, the obtained collision probability was below a predefined threshold of 0.05 at t = 2.0 s, which is marked in blue. Thereafter, the framework provided an estimated collision probability above the threshold until a collision occurred at t = 5.0 s, where the estimated collision gradually increased to 1 (Fig. 16(f)). In summary, the proposed framework can reasonably ensure safety at the beginning and effectively identify potential crashes, owing to the established stochastic FRS.

In terms of computational efficiency, the proposed framework is promising for real-time risk assessment applications, and combining it with BRS can result in further time savings. This is because the BRS cached in a look-up table can quickly identify whether a vehicle is theoretically safe, requiring only a few milliseconds, whereas the stochastic FRS risk assessment can take between 30 and 50 ms depending on the experimental settings. Because most highway driving scenarios do not involve safety–critical vehicle

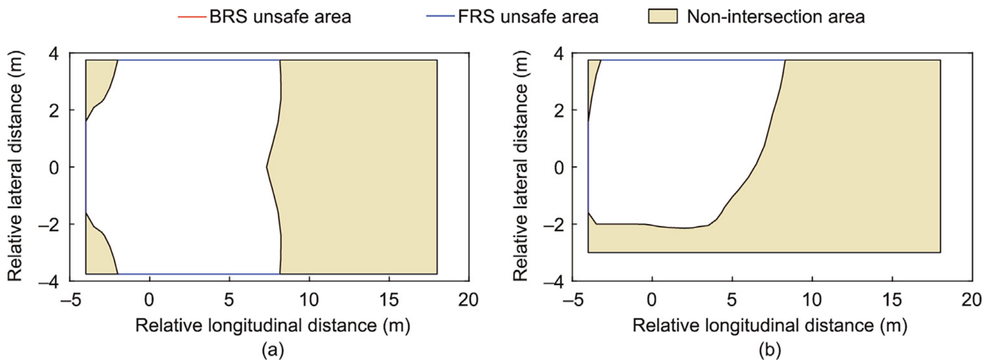


Fig. 14. Comparisons for the BRS and FRS-based unsafe areas. 28 and 30 m·s^{-1} for the surrounding and ego vehicles, respectively. The ego was at the origin. (a) The lateral speed of the surroundings was 0 m·s^{-1}. The relative heading angle was 0°. (b) The lateral speed of the surroundings was −1 m·s^{-1}. The relative heading angle range was 9°–45°.

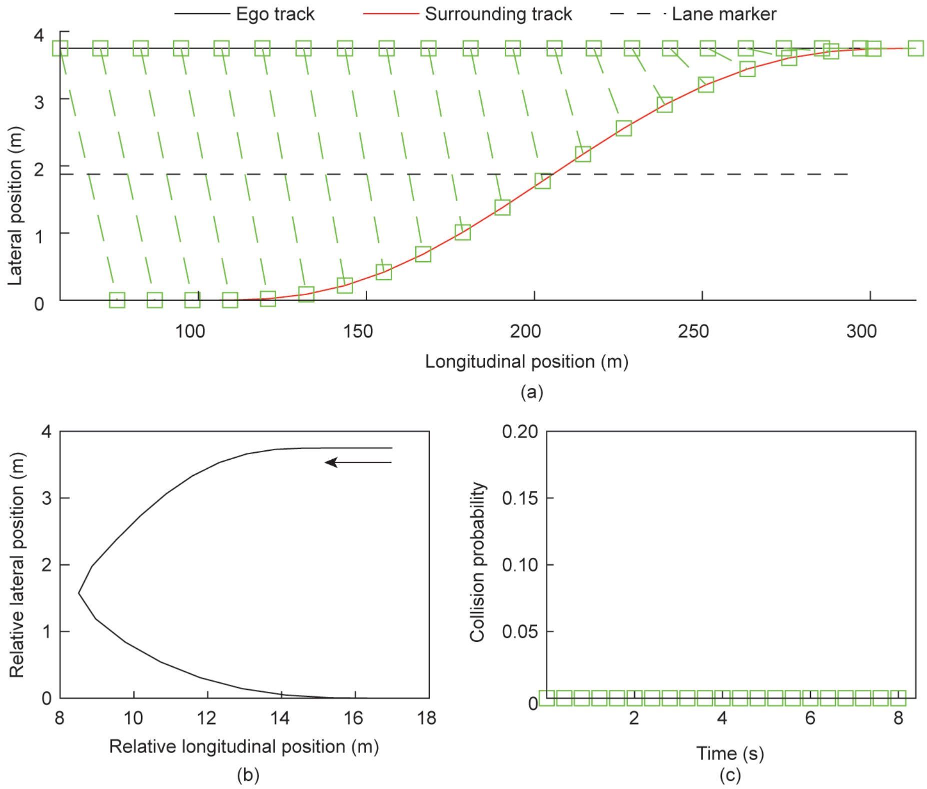


Fig. 15. Collision–detection results using the integrated framework. The initial longitudinal speeds of the ego and surrounding vehicles were 30 and 28 m·s^{-1}, respectively. (a) The squares denote vehicle positions at each time step and the two vehicle positions at the same step are connected by dash lines. Each connected position pair in green indicates that the car–car interaction is identified as theoretically safe by the BRS. (b) The relative vehicle positions, where the arrow indicates the evolution direction. (c) The estimated collision probability marked by green squares remains zero because the relative positions between the two vehicles were outside of the unsafe area identified by the BRS.

interactions [34], the BRS can prevent unnecessary computation time for constructing a stochastic FRS if safety is ensured. Note that real-time capability is achieved based on an advanced GPU enabling 2048 × 28 parallel computations with a power consumption of up to 1.5 kW·h [39]. If the GPU operates continuously for stochastic FRS construction, it can consume 10% of the entire power consumption of an electric vehicle. However, naturalistic driving data indicate that vehicle interactions are not critical in most instances [34]. This implies that safety can be ensured by BRS in most scenarios. Consequently, the integrated risk assessment framework can benefit from a significant reduction in power consumption by not using GPU computation in most instances.

6. Conclusions

We developed a reachability-based framework for collision detection during highway driving. Inspired by two different RA approaches, a cached BRS is first employed to formally verify whether the current interaction safety can be theoretically ensured. Otherwise, a prediction-based confidence-aware stochastic FRS is calculated online at each predicted time step for collision probability estimation. If the estimated collision probability exceeds a predefined threshold, the ego vehicle can execute brakes or swerve to avoid potential crashes. Thus, the proposed framework can ensure risk-free interactions in non-risky events and provide accurate collision estimation for safety–critical events.

Using both naturalistic driving data and simulated safety–critical cut-in events, we conducted extensive experiments to validate the performance of the proposed acceleration prediction model, which effectively considered interactions between vehicles for constructing a stochastic FRS. Furthermore, we analyzed how infusing confidence beliefs can improve acceleration prediction accuracy. Simulation results showed that the proposed FRS-based approach PSRS-5β with five confidence levels can adjust complicated highway scenarios, resulting in more agile collision–detection results. The integrated framework was tested for both risky and non-risky events. We demonstrated that the proposed framework cannot only accurately identify potential collision events but also has the potential to avoid false–positive results in safety–critical but collision-free events.

The proposed approach assumes the use of an advanced GPU for efficient parallel computing in the stochastic FRS. Further efforts are required to implement hardware with lower power consumption for real-world testing. The developed acceleration predictor was trained on the highD dataset, which contains only highway trajectories. We can only apply our approach to diverse scenarios such as urban and country-road environments by training the predictor using a more diverse dataset. Future research can integrate risk assessment to enable safer and more efficient motion planning and employ the proposed risk assessment framework for an actual vehicle.

Acknowledgments

This research was supported by the proactive SAFEty systems and tools for a constantly UPgrading road environment (SAFE-UP) project. SAFE-UP has received funding from the European Union's Horizon 2020 Research and Innovation Program (861570).

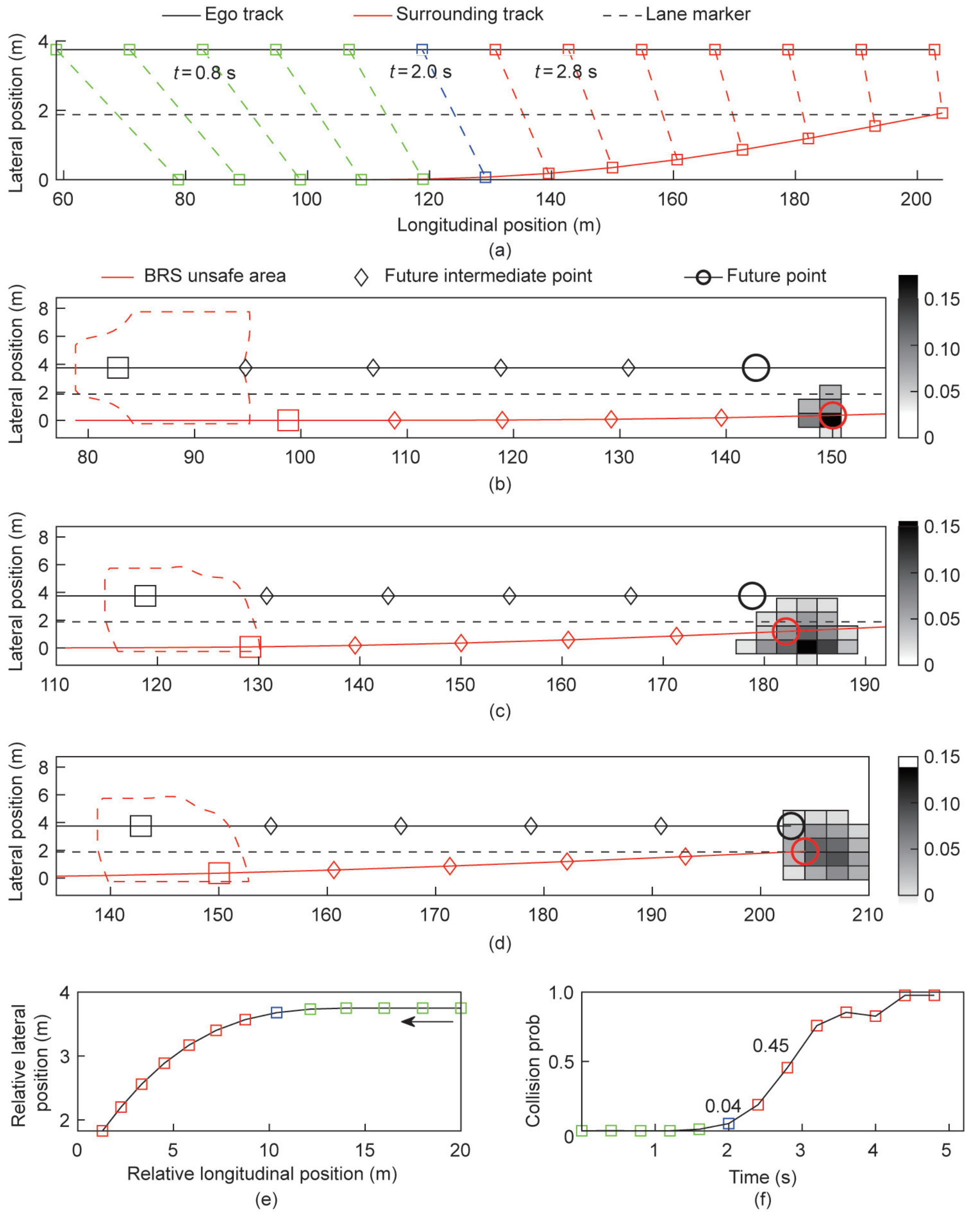


Fig. 16. Collision detection using the integrated framework. The initial longitudinal speeds of the ego and surrounding vehicles were 30 and 25 m·s^{-1}, respectively. (a) The squares denote vehicle positions at each time step and the two vehicle positions at the same step are connected by dash lines. Each connected position pair in green indicates that the car–car interaction was identified as theoretically safe by the BRS, the pair in blue indicates the interaction had an estimated collision probability below a threshold (0.05 in this case), and the pairs in red indicate the interaction had a collision probability above the threshold. (b–d) The shapes of BRS and FRS at t = 0.8, 2.0, and 2.8 s, respectively. Note that to calculate the collision probability, we established the stochastic FRS at each future time step including the four intermediate time points, and we only show the stochastic FRS at the last prediction time step for convenience. The current position of the surrounding vehicle was outside the BRS unsafe area of the ego, thus the stochastic FRS did not require to be constructed in practice in (b). We show only the stochastic FRS at t = 0.8 s in (b) for completeness. The relative vehicle positions were shown in (e), where the arrow indicates the evolution direction. The estimated collision probability in (f) is marked as green at the beginning, then blue and red at the end phase.

Compliance with ethics guidelines

Xinwei Wang, Zirui Li, Javier Alonso-Mora, and Meng Wang declare that they have no conflict of interest or financial conflicts to disclose.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.eng.2023.10.010.

References

[1] Kalra N, Paddock SM. Driving to safety: how many miles of driving would it take to demonstrate autonomous vehicle reliability? Transp Res Part A Policy Pract 2016;94:182–93.

[2] Lefèvre S, Vasquez D, Laugier C. A survey on motion prediction and risk assessment for intelligent vehicles. ROBOMECH J 2014;1(1):1–14.

[3] Mukhtar A, Xia L, Tang TB. Vehicle detection techniques for collision avoidance systems: a review. IEEE T Intell Transp 2015;16(5):2318–38.

[4] Pek C, Manzinger S, Koschi M, Althoff M. Using online verification to prevent autonomous vehicles from causing accidents. Nat Mach Intell 2020;2(9):518–28.

[5] Falcone P, Ali M, Sjoberg J. Predictive threat assessment via reachability analysis and set invariance theory. IEEE T Intell Transp 2011;12(4):1352–61.

[6] Althoff M, Dolan JM. Online verification of automated road vehicles using reachability analysis. IEEE Trans Robot 2014;30(4):903–18.

[7] Althoff M, Frehse G, Girard A. Set propagation techniques for reachability analysis. Annu Rev Contr Robot 2021;4(1):369–95.

[8] Leung K, Schmerling E, Zhang M, Chen M, Talbot J, Gerdes JC, et al. On infusing reachability-based safety assurance within planning frameworks for human–robot vehicle interactions. Int J Robot Res 2020;39(10–11):1326–45.

[9] Chapman MP, Bonalli R, Smith KM, Yang I, Pavone M, Tomlin CJ. Risk-sensitive safety analysis using conditional value-at-risk. IEEE T Automat Contr 2021;67(12):6521–36.

[10] Lee DN. A theory of visual control of braking based on information about time-to-collision. Perception 1976;5(4):437–59.

[11] Minderhoud MM, Bovy PH. Extended time-to-collision measures for road traffic safety assessment. Accident Anal Prev 2001;33(1):89–97.

[12] Vogel K. A comparison of headway and time to collision as safety indicators. Accident Anal Prev 2003;35(3):427–33.

[13] Mammar S, Glaser S, Netto M. Time to line crossing for lane departure avoidance: a theoretical study and an experimental setting. IEEE T Intell Transp 2006;7(2):226–41.

[14] Saunier N, Sayed T. Probabilistic framework for automated analysis of exposure to road collisions. Transp Res Rec 2008;2083(1):96–104.

[15] Davis GA, Hourdos J, Xiong H, Chatterjee I. Outline for a causal model of traffic conflicts and crashes. Accident Anal Prev 2011;43(6):1907–19.

[16] Kuang Y, Qu X, Wang S. A tree-structured crash surrogate measure for freeways. Accident Anal Prev 2015;77:137–48.

[17] Mullakkal-Babu FA, Wang M, He X, van Arem B, Happee R. Probabilistic field approach for motorway driving risk assessment. Transport Res C Emer 2020;118:102716.

[18] Katare D, El-Sharkawy M. Embedded system enabled vehicle collision detection: an ANN classifier. In: Proceeding of 2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC); 2019 Jan 7–9; Las Vegas, USA. Washington, DC: IEEE; 2019. p. 0284–289.

[19] Saleh K, Hossny M, Nahavandi S. Kangaroo vehicle collision detection using deep semantic segmentation convolutional neural network. In: Proceeding of 2016 International Conference on Digital Image Computing: Techniques and Applications (DICTA); 2016 Nov 30–Dec 2; Gold Coast, QLD, Australia. Washington, DC: IEEE; 2016. p. 1–7.

[20] Laugier C, Paromtchik IE, Perrollaz M, Yong MY, Yoder J, Tay C, et al. Probabilistic analysis of dynamic scenes and collision risks assessment to improve driving safety. IEEE Intel Transp SY 2011;3(4):4–19.

[21] Annell S, Gratner A, Svensson L. Probabilistic collision estimation system for autonomous vehicles. In: Proceeding of 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC); 2016 Nov; Rio de Janeiro, Brazil. Washington, DC: IEEE; 2016. p. 473–8.

[22] Kim J, Kum D. Collision risk assessment algorithm via lane-based probabilistic motion prediction of surrounding vehicles. IEEE T Intell Transp 2018;19(9):2965–76.

[23] Pek C, Zahn P, Althoff M. Verifying the safety of lane change maneuvers of self-driving vehicles based on formalized traffic rules. In: Proceeding of 2017 IEEE Intelligent Vehicles Symposium (IV); 2017 Jun 11–14; Las Vegas, NV, USA. Washington, DC: IEEE; 2017.

[24] Törnblom J, Nadjm-Tehrani S. An abstraction-refinement approach to formal verification of tree ensembles. In: Guiochet J, Tonetta S, Schoitsch E, Roy M, Bitsch F, editors. SAFECOMP 2019 Workshops, ASSURE, DECSoS, SASSUR, STRIVE, and WAISE; 2019 Sep 10; Turku, Finland. Berlin: Springer; 2019.

[25] Althoff M, Stursberg O, Buss M. Model-based probabilistic collision detection in autonomous driving. IEEE T Intell Transp 2009;10(2):299–310.

[26] Bansal S, Chen M, Herbert S, Tomlin CJ. Hamilton-Jacobi reachability: a brief overview and recent advances. 2017 IEEE 56th Annual Conference on Decision and Control (CDC); In: Proceeding of 2017 Dec 12–15; Melbourne, VIC, Australia. Washington, DC: IEEE; 2017.

[27] Fridovich-Keil D, Bajcsy A, Fisac JF, Herbert SL, Wang S, Dragan AD, et al. Confidence-aware motion prediction for real-time collision avoidance[1]. Int J Robot Res 2020;39(2–3):250–65.

[28] Wang X, Li Z, Alonso-Mora J, Wang M. Prediction-based reachability analysis for collision risk assessment on highways. In: Proceeding of 2022 IEEE Intelligent Vehicles Symposium (IV); 2022 Jun 5–9; Aachen, Germany. Washington, DC: IEEE; 2022. p. 504–10.

[29] Deo N, Trivedi MM. Convolutional social pooling for vehicle trajectory prediction. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops; 2018 Jun 18–23; Salt Lake, UT, USA. Washington, DC: IEEE; 2018.

[30] Wang X, Alonso-Mora J, Wang M. Probabilistic risk metric for highway driving leveraging multi-modal trajectory predictions. IEEE T Intell Transp 2022;23(10):19399–412.

[31] Zhou M, Qu X, Li X. A recurrent neural network based microscopic car following model to predict traffic oscillation. Transport Res C Emer 2017;84:245–64.

[32] Li A, Sun L, Zhan W, Tomizuka M, Chen M. Prediction-based reachability for collision avoidance in autonomous driving. In: Procceding of 2021 IEEE International Conference on Robotics and Automation (ICRA); 2021 May 30–Jun 5; Xi'an, China. Washington, DC: IEEE; 2021.

[33] Su J, Beling PA, Guo R, Han K. Graph convolution networks for probabilistic modeling of driving acceleration. In: Proceeding of 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC); 2020 Sep 20–23; Rhodes, Greece. Washington, DC: IEEE; 2020.

[34] Krajewski R, Bock J, Kloeker L, Eckstein L. The highD dataset: a drone dataset of naturalistic vehicle trajectories on German highways for validation of highly automated driving systems. In: Proceeding of 2018 21st International Conference on Intelligent Transportation Systems (ITSC); 2022 Sep 18–Oct 12; Macao, China. Washington, DC: IEEE; 2018.

[35] Turner D, Andresen D, Hutson K, Tygart A. Application performance on the newest processors and GPUs. In: Proceedings of the Practice and Experience on Advanced Research Computing; 2018 Jul 22–26; Pittsburgh, PA, USA. New York City: Association for Computing Machinery; 2018.

[36] Goutte C, Gaussier E. A probabilistic interpretation of precision, recall and F-score, with implication for evaluation. Advances in Information Retrieval: 27th European Conference on IR Research, ECIR 2005; 2005 Mar 21–23; Santiago de Compostela, Spain. Berlin: Springer; 2005.

[37] Mullakkal-Babu FA, Wang M, van Arem B, Shyrokau B, Happee R. A hybrid submicroscopic–microscopic traffic flow simulation framework. IEEE T Intell Transp 2020;22(6):3430–43.

[38] Kurtc V. Studying car-following dynamics on the basis of the highD dataset. Transp Res Rec 2020;2674(8):813–22.

[39] Mittal S, Vetter JS. A survey of methods for analyzing and improving GPU energy efficiency. ACM Comput Surv 2014;47(2):1–23.