Enabling Safe Freeway Driving for Automated Vehicles

Changliu Liu and Masayoshi Tomizuka

***Abstract*— The development of automated vehicles brings new challenges to road safety. The behavior of the automated vehicles should be carefully designed in order to interact with the environment and other vehicles efficiently and safely. This paper is focused on the learning and decision making methods for the automated vehicles towards safe freeway driving. Based on a multi-agent traffic model, the decision making problem is posed as an optimal control problem, which is solved by 1) behavior classification and trajectory prediction of the surrounding vehicles, and 2) a unique parallel planner architecture which addresses the efficiency goal and the safety goal separately. The simulation results demonstrate the effectiveness of the algorithm.**

I. Introduction

Automated driving is widely viewed as a promising technology to revolutionize today's transportation systems [1], so as to free the human drivers, ease the road congestion and lower the fuel consumption among other benefits. Substantial research efforts are directed into this field from research groups and companies [2], which are also encouraged by policy makers [3].

When the automated vehicles drive on public roads, safety is a big concern. While existing technologies can assure high-fidelity sensing, real-time computation and robust control, the challenges lie in the interactions between the automated vehicle and the environment which includes other manually driven vehicles [4]. For road safety, the driving behavior for the automated vehicles should be carefully designed. In other words, given sensory information from multiple sensors (e.g. cameras and LIDAR), an automated vehicle should be intelligent enough to find a safe and efficient trajectory to its destination, which takes into account of the complex environment with multiple surrounding vehicles.

Conservative strategies such as "braking when collision is anticipated", known as the Auto Brake function in existing models [5], are not the best actions in most cases (although they may be necessary in certain cases). Taking into account the dynamics and future course of surrounding vehicles, the automated vehicle has multiple choices for a safe maneuver, i.e. i) slow down to keep a safe headway till the headway reaches the safe limit; ii) steer to the left or right to avoid a collision; iii) even speed up if it can get out a dangerous zone by so doing, etc. The autonomy in driving needs high level machine intelligence [6].

This paper proposes a framework in designing the driving behavior for automated vehicles to prevent or minimize occurrences of collisions among vehicles and obstacles while maintaining efficiency (e.g. maintaining high speed on freeway). The three-layer Robustly-Safe Automated Driving (ROAD) system is considered, as shown in Fig.1. The focus of this paper is in layer 2: the learning and decision making algorithm. Freeway driving will be considered as shown in Fig.2. At each time step, the automated vehicle predicts the future courses of all surrounding vehicles and confines its trajectory in a safe region regarding the prediction.

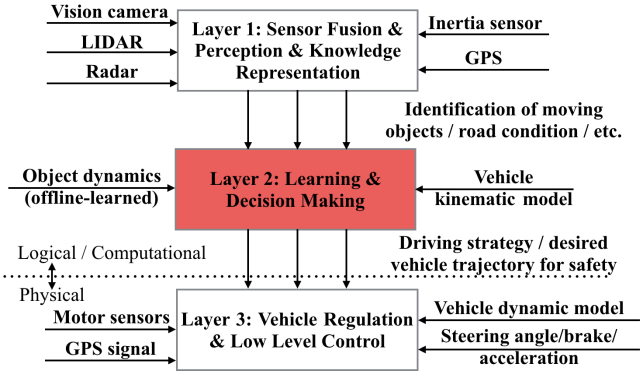


Fig. 1: Architecture for the robustly-safe automated driving (ROAD) system

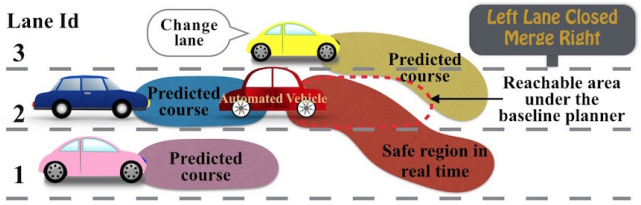


Fig. 2: Illustration of the freeway driving scenario

The remainder of the paper is organized as follows: in section II, a multi-agent traffic model will be introduced and the control problem for automated driving will be formulated. The learning and prediction algorithm will be discussed in section III, followed by the decision making algorithm in section IV. Simulation studies will be presented in section V. Section VI concludes the paper.

II. The Multi-Agent Traffic Model

The scenario shown in Fig.2 will be modeled in the framework of multi-agent systems. Moreover, it is assumed that the lanes in the freeway are along the x-direction with no curvature. The lanes are indexed increasing from the right lane to the left lane.

*This work was supported by a Berkeley fellowship awarded to Changliu Liu and by Denso International America, Inc.

C. Liu and M. Tomizuka are with the Department of Mechanical Engineering, University of California, Berkeley, CA 94720 USA (e-mail: {changliuliu, tomizuka}@berkeley.edu).

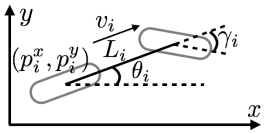


Fig. 3: The kinematic model for vehicle i

A. The System Model

A multi-agent system (MAS) [7] is a system composed of multiple interacting intelligent agents within an environment. All vehicles (the automated vehicle and other manually driven vehicles) on freeway are viewed as agents, which have several important characteristics: 1) autonomy: the agents are self-aware and autonomous; 2) local views: no agent has a full global view of the system; 3) decentralization: there is no designated controlling agent.

From a global view, there can be thousands of agents (vehicles) in the system (e.g. on the freeway). But only the local interactions among the vehicles are of interest. Hence, in the controller of the automated vehicle, only the behavior of the surrounding vehicles will be analyzed, while surrounding vehicles refer to the vehicles that can be detected and within certain distances to the automated vehicle.

Suppose there are N surrounding vehicles locally and are indexed from 1 to N. Let $H = \{1, .., N\}$ be the set of indices for all surrounding vehicles. The automated vehicle has index 0. Since the surrounding vehicles are changing from time to time, mathematically, the topology of the MAS is time varying [8].

For vehicle i, denote its state as x_i, control input as u_i. Its dynamic equation can be written as

$$\dot{x}_i = f_i(x_i, u_i, w_i), \forall i = 0, \cdots, N \tag{1}$$

where w_i is the disturbance introduced by the environment, e.g. wind. For simplicity, only the kinematic model will be considered in layer 2. Figure 3 shows the bicycle model used for all vehicle i, where (p_i^x, p_i^y) is the position of the center of the rear axle, v_i the forward speed, θ_i the vehicle heading ($\theta_i = 0$ when the vehicle is following the lane), γ_i the steer angle and L_i the vehicle wheelbase. Assuming no tire slip angle, the kinematics of vehicle i follow from $\dot{p}_i^x = v_i \cos(\theta_i), \dot{p}_i^y = v_i \sin(\theta_i), \dot{\theta}_i = \frac{v_i}{L_i} \tan \gamma_i$. Since the mapping from γ_i to $\dot{\theta}_i$ is homeomorphic given v_i, $\dot{\theta}_i$ is chosen as an input signal instead of γ_i. For vehicle i, define $x_i = [p_i^x, p_i^y, v_i, \theta_i]^T$ and $u_i = [\dot{v}_i, \dot{\theta}_i]^T$. Hence (1) can be simplified as

$$\dot{x}_i = f(x_i) + \mathcal{B}u_i + \mathcal{B}w_i, \forall i = 0, \cdots, N \tag{2}$$

where

$$f(x_i) = \begin{bmatrix} v_i \cos\theta_i \\ v_i \sin\theta_i \\ 0 \\ 0 \end{bmatrix}, \mathcal{B} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} = [\mathcal{B}_1, \mathcal{B}_2] \tag{3}$$

Let x_e be the state of the environment, e.g. speed limit v_{lim}, stationary obstacles and so on. Then the system state x is defined as $x = [x_0^T, x_1^T, \cdots, x_N^T, x_e^T]^T$.

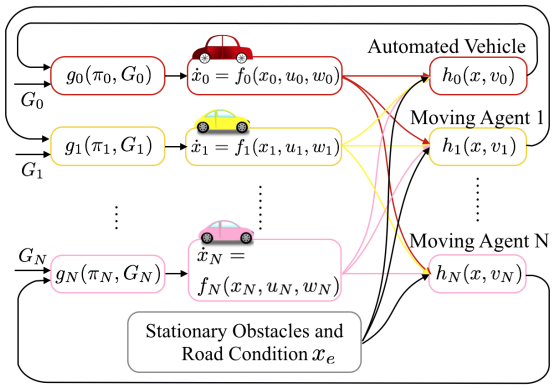


Fig. 4: The block diagram for local interactions

Agent i chooses the control u_i based on its information set π_i and its objective G_i (which can be intended behaviors or desired speed). The information set is a combination of the measurements and the communicated information. In this paper, it is assumed that there is no direct communication among vehicles. In this way, agent i's information set at time T contains all the measurements up to time T, i.e. $\pi_i(T) = \{y_i(t)\}_{t\in[0,T]}$ where

$$y_i = h_i(x, z_i), \forall i = 0, \cdots, N \tag{4}$$

and z_i is the measurement noise. The controller can be written as

$$u_i = g_i(\pi_i, G_i), \forall i = 0, \cdots, N \tag{5}$$

Based on (1), (4) and (5), the block diagram for the multi-agent system is shown in Fig.4, where every row represents one agent. All agents are coupled together in the closed loop system due to measurement feedbacks.

B. The Optimal Control Problem

In layer 2 of the ROAD system, the controller for the automated vehicle, i.e. the function g_0, is to be designed, which should be chosen by balancing the following two factors: 1) **Efficiency**: The objective G_0 should be achieved in an optimal manner through minimizing a cost function $J(x_0, u_0, G_0)$; 2) **Safety**: Efficiency requirement should be fulfilled safely. Denote the system's safe set as X_S, which is a closed subset of the state space X of the system state x that is collision free. Then g_0 should be chosen such that $\forall t$, $x(t) \in X_S$. The following optimal control problem can be formulated [9]:

$$\min_{u_0} \quad E(J(x_0, u_0, G_0)) \tag{6}$$

$$s.t. \quad u_0 \in \Omega, x_0 \in \Gamma(x_e), \dot{x}_0 = f(x_0) + \mathcal{B}u_0 \tag{7}$$

$$x \in X_S \tag{8}$$

where Ω is the control space constraint for vehicle stability, which depends on model uncertainties and disturbances [10], and $\Gamma(x_e)$ is the state space constraint regarding the speed limit. The above optimization is generally hard to solve in view of the safety constraint (8), as the dynamics of the system state x is only partially known and the set X_S is

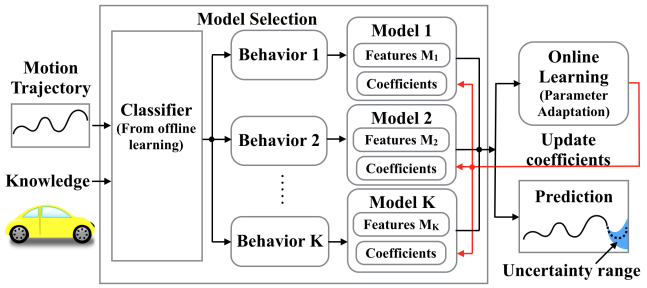
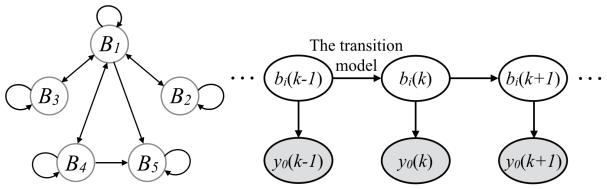


Fig. 5: The structure of the learning and prediction center



(a) The transition model (b) The HMM model

Fig. 6: The behavior transition model and the hidden Markov model

non convex. To solve the problem, the behaviors of the surrounding vehicles will be identified and predicted online (section III) and a parallel structure will be introduced to solve the non convex optimization efficiently (section IV).

III. Identification and Prediction of the Driving Behavior of Surrounding Vehicles

Instead of predicting other vehicles' trajectories directly, human drivers may classify other drivers' intended behavior first. If the intended driving behaviors are understood, the future trajectories can be predicted using empirical models. Mimicking what humans would do, the learning structure in Fig.5 is designed for the automated vehicle to make predictions of the surrounding vehicles, where the process is divided into two steps: 1) the behavior classification, where the observed trajectory of a vehicle goes through an offline trained classifier; and 2) the trajectory prediction, where the future trajectory is predicted based on the identified behavior, by using an empirical model which contains adjustable parameters to accommodate the driver's time-varying behavior.[1] The classification step is needed when the communications among vehicles are limited. Otherwise, vehicles can broadcast their planned behaviors.

In this section, the design of the classifier, the empirical models and the online learning algorithm will be discussed.

A. The Driving Behaviors

Denote the intended behavior of vehicle i at time step k as $b_i(k)$. In this paper, five behaviors are considered:

- Behavior 1 (B_1): Lane following;
- Behavior 2 (B_2): Lane changing to the left;
- Behavior 3 (B_3): Lane changing to the right;
- Behavior 4 (B_4): Lane merging;
- Behavior 5 (B_5): Lane exiting;

where B_1 is the steady state behavior; B_2, B_3 and B_4 are driving maneuvers; and B_5 is the exiting behavior. It is assumed that there must be gaps (lane following) between two maneuvers (B_2, B_3 and B_4). The transitions among behaviors follow from the model shown in Fig.6a.

Let $P(b_i(k)|\pi_0(k)) \in \mathbb{R}^5$ be the probability vector that vehicle i intends to conduct $B_1, \cdots, B_5$ at time step k given information up to time step k. The relationship between $P(b_i(k)|\pi_0(k))$ and $P(b_i(k+1)|\pi_0(k))$ can be described by a Markov matrix $A = P(b_i(k+1)|b_i(k)) \in \mathbb{R}^{5\times 5}$, e.g.

$$P(b_i(k+1)|\pi_0(k)) = A * P(b_i(k)|\pi_0(k)) \tag{9}$$

where A represents the transition model[2] in Fig.6a.

The transition model should be invoked in calculation only when vehicle i is following the lane and is about to conduct a maneuver. When vehicle i is conducting a maneuver, there is no need to calculate the probability distribution over other behaviors. The transition model can be used again when the maneuver is completed or aborted and vehicle i starts to follow the lane. The intuition is that: although the intention of a driver is unknown (thus needs to be inferred), but his action is observable (thus when he turns his intention into action, there is no need to guess).

B. The Classifier and the Features

The intentions of a driver at different time steps form a Markov process, which, however, are unknown to the automated vehicle. Hence the behavior classification problem becomes an inference problem in the Hidden Markov Model (HMM) [11] as shown in Fig.6b. According to Bayes' rule, at time step k, for $j = 1, 2, \cdots, 5$,

$$\begin{aligned} & P(b_i(k) = B_j|\pi_0(k)) \\ \propto\ & P(b_i(k) = B_j, y_0(0), \cdots, y_0(k)) \\ \propto\ & P(y_0(k)|b_i(k) = B_j)P(b_i(k) = B_j|\pi_0(k-1)) \end{aligned} \tag{10}$$

where $P(b_i(k)|\pi_0(k-1))$ encodes the temporal transitions of the intended behaviors and can be obtained using (9). $P(y_0(k)|b_i(k))$ is the measurement model, which can be constructed from data offline[3]. To represent the high-dimension data efficiently, the measurement y_0 is divided into several features for each surrounding vehicle i (not limited to the following ones):

- Feature 1: Longitudinal acceleration $f_i^1 = \dot{v}_i$.
- Feature 2: Deceleration light $f_i^2 = (1, 0) = (\text{on}, \text{off})$.
- Feature 3: Turn signal $f_i^3 = (1, 0, -1) = (\text{left}, \text{off}, \text{right})$.
- Feature 4: Speed relative to the traffic flow $f_i^4 = v_i - \bar{v}$.

[1] Consider Fig.4. When G_i denotes vehicle i's intended behavior, the behavior classification is a backward process to identify G_i, while the trajectory prediction is a forward process to predict vehicle i's closed loop behavior $\dot{x}_i = f(x_i) + \mathcal{B}g_i(\pi_i, G_i)$ based on identified G_i.

[2] A can be trained using real world labeled data $\underline{b}_i(k)$, e.g. $A_{pq} = P(b_i(k+1) = B_p|b_i(k) = B_q) := \sum_{i,k} I(\underline{b}_i(k+1) = B_p, \underline{b}_i(k) = B_q) / \sum_{i,k} I(\underline{b}_i(k) = B_q)$ where I is the indicator function.

[3] Similar to the transition model A, the measurement model can also be obtained by supervised training, together with necessary curve fitting.

initialization $P(b_i(0)) = [1, 0, 0, 0, 0]^T$, $k = 0$;
while *Classifier is Active* **do**
  $k = k + 1$;
  read current $y_0(k)$, calculate $f_i^1(k), \cdots, f_i^{10}(k)$;
  if $f_i^6(k) == -1$ **then**
    $B^* = \arg\max_{B=B_3,B_3,B_4} P(b_i(k-1) = B)$;
    $P(b_i(k) = B^*) = 1$, $P(b_i(k) \neq B^*) = 0$;
    while $f_i^6(k) == -1$ **do**
      $k = k + 1$, read $y_0(k)$, calculate $f_i^6(k)$;
      $P(b_i(k)) = P(b_i(k-1))$;
    end
    $P(b_i(k)) = [1, 0, 0, 0, 0]^T$;
  else
    $P(b_i(k)|\pi_0(k-1)) = A * P(b_i(k-1)|\pi_0(k-1))$;
    calculate $M = P(y_0(k)|b_i(k))$;
    $P(b_i(k)|\pi_0(k)) = M. * P(b_i(k)|\pi_0(k-1))$;
    normalize $P(b_i(k)|\pi_0(k))$;
  end
end

Algorithm 1: Behavior classification for vehicle i

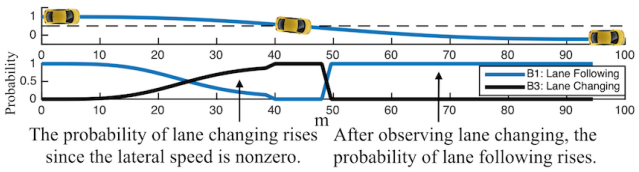


Fig. 7: Behavior classification in a two-lane case

- Feature 5: Speed relative to the front vehicle $f_i^5 = v_i - v_{front}$.[4]
- Feature 6: Current lane Id f_i^6. $f_i^6 = -1$ if the vehicle is occupying two lanes.
- Feature 7: Current lane clearance $f_i^7 = (1, 0, -1) =$ (blocked, clear, ended).
- Feature 8: Lateral velocity, e.g. $f_i^8 = v_i \sin\theta_i$.
- Feature 9: Lateral deviation from the center of its current lane f_i^9.
- Feature 10: Lateral deviation from the center of its target lane (if in the lane changing mode) f_i^{10}.

Algorithm 1 is designed based on the previous discussion. The probability distributions over all possible behaviors are calculated at each time step when the vehicle is following the lane (not occupying two lanes). The update of the distribution stops when the vehicle is conducting a maneuver. After the maneuver is completed, the probabilities will be initialized. Figure 7 illustrates the behavior classification results under algorithm 1, where the measurement model is set empirically as $P(y_0|b_i = B_1) \propto exp(-(f_i^8)^2 - c(f_i^9)^2)$, $P(y_0|b_i = B_3) \propto 1 - exp(-(f_i^8)^2 - c(f_i^9)^2)$ and c is a constant. In the beginning, the vehicle is following a lane. Then the probability of lane changing rises since the lateral speed f_i^6 goes up. When the vehicle crosses the boundary of two lanes, the probability of B_3 goes to 1. After lane changing, the vehicle starts to follow the new lane.

[4]Feature 4 and 5 encodes the interactions among vehicles.

C. The Empirical Models for Trajectory Prediction

The future trajectory is predicted according to the most likely predicted behavior $\arg\max_{B_j} P(b_i(k) = B_j)$. For lane following B_1, $\theta_i \approx 0$ and the lateral deviation of the vehicle can be ignored. The vehicle i only regulates its longitudinal speed to match the speed of the traffic flow and the speed of its front vehicle, e.g.

$$\dot{x}_i = f(x_i) + \mathcal{B}_1[k_1 f_i^4 + k_2 f_i^5] \tag{11}$$

where $k_1, k_2 \in \mathbb{R}$ are online-adjustable parameters. The parameter identification method for k_1 and k_2 will be discussed in the next section. For lane changing B_2 and B_3, the vehicle not only regulates the longitudinal speed, but also regulates the lateral position, hence the turning rate. The empirical model can be described as:

$$\dot{x}_i = f(x_i) + \mathcal{B}_1[k_1 f_i^4 + k_2 f_i^5] + \mathcal{B}_2[k_3 f_i^8 + k_4 f_i^{10}] \tag{12}$$

where $k_1, k_2, k_3, k_4 \in \mathbb{R}$ are online-adjustable parameters.

D. Online Parameter Adaptation

The online parameter adaptation is needed to capture the time varying behavior of drivers. For example, to identify k_1 and k_2 in (11), the reduced order equation will be used: $f_i^1 = \dot{v}_i = k_1 f_i^4 + k_2 f_i^5 = \beta^T \varphi$ where $\beta = [k_1, k_2]^T$ and $\varphi = [f_i^4, f_i^5]^T$. Based on the historical data of f_i^1, f_i^4 and f_i^5, k_1 and k_2 can be identified using the recursive least square algorithm [12]. Let $\hat{\beta}(k) = [\hat{k}_1(k), \hat{k}_2(k)]^T$ be the estimates of $[k_1, k_2]$ at time step k. Then the estimates can be updated by the following equation,

$$\hat{\beta}(k+1) = \hat{\beta}(k) + F\varphi(k)\left(f_i^1(k+1) - \hat{\beta}(k)^T\varphi(k)\right) \tag{13}$$

where F is the learning gain. Although $\hat{k}_1$ and $\hat{k}_2$ are identified using historical data, they can be used to predict the trajectories in the near future if the vehicle's behavior does not change very fast compared to the sampling rate.

IV. Online Decision Making and Control

Based on the predictions of the surrounding vehicles, the automated vehicle needs to find a safe and efficient trajectory satisfying the optimal control problem (6-8). However, since the uncertainties regarding the predictions will accumulate in time, there is no need to solve the original problem in a long time horizon. The decision making architecture in Fig.8 is proposed, which is designed to be a parallel combination of a baseline planner that solves the problem in a long time horizon without the safety constraint (8), and a safety planner that takes care of the safety constraint in real time [13].

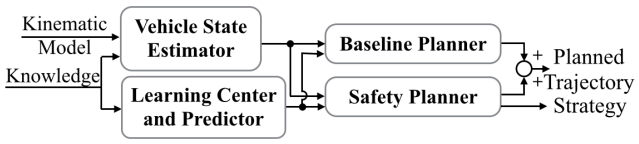


Fig. 8: The structure of the decision making center

A. The Baseline Planner

The baseline planner solves the optimal control problem (6-7) to ensure efficiency, which is similar to the planner in use when the automated vehicle is navigating in an open environment. When the cost function and the control constraint are designed to be convex, (6-7) become a convex optimization problem.

Suppose the objective G_0 (target behavior and target speed v_r) is specified. The baseline planner tries to plan a trajectory to accomplish G_0. When the objective is to follow the lane, the cost function is designed as

$$J = \int_0^\infty \left[(v_0 - v_r)^2 + q(f_0^9)^2 + u_0^T R u_0\right] dt \tag{14}$$

where $q \in \mathbb{R}^+$ and $R \in \mathbb{R}^{2\times 2}$ is positive definite. When the objective is to change lane, the automated vehicle should change the lane smoothly within time T. The cost function is designed as

$$J = \int_0^T \Phi(t)dt + \Gamma(f_0^{10}(T), \theta_0(T)) \tag{15}$$

where $\Phi = (v_0 - v_r)^2 + q_1(f_0^{10})^2 + q_2(\theta_0)^2 + u_0^T R u_0$ is the cost to go; $\Gamma = s_1(f_0^{10}(T))^2 + s_2(\theta_0(T))^2$ is the terminal cost; $q_1, q_2, s_1, s_2 \in \mathbb{R}^+$; and $R \in \mathbb{R}^{2\times 2}$ is positive definite.

The computation in the baseline planner can be done offline. The resulting control policy will be stored for online application, to ensure real-time planning.

B. The Safety Planner

The safety planner modifies the trajectory planned by the baseline planner locally to ensure that it will lie in the safe set X_S in real time.

During lane following, the safety constraint requires the automated vehicle to keep a safe headway. Thus $X_S(B_1) = \{x : d_1(x) \geq d_{min}\}$ where $d_1(x)$ calculates the minimum distance between the automated vehicles and the vehicle or obstacles in front of it. During lane changing, the safety constraint requires the automated vehicle to keep a safe distance from vehicles on both lanes. Thus $X_S(B_2), X_S(B_3) = \{x : d_2(x) \geq d_{min}\}$ where $d_2(x)$ calculates the minimum distance between the automated vehicle and all surrounding vehicles and obstacles in the two lanes.

Mathematically, the safe set is described using a safety index ϕ, which is a real-valued continuously differentiable function on the system's state space. The state x is considered safe only if $\phi(x) \leq 0$ as shown in Fig.9. In Fig.2, the safe region for the automated vehicle is affected by the future trajectory of the surrounding vehicles. Based on the prediction of other vehicles, if the baseline trajectory leads to $\phi \geq 0$ now or in the near future, the safety planner will generate a modification signal to decrease the safety index by making $\dot{\phi} < 0$.

In order for the above strategy to work, the safety index needs to satisfy the following conditions: 1) $\partial\dot{\phi}/\partial u_0 \neq 0$; 2) the unsafe set X_S^c is not reachable given the control law $\dot{\phi} < 0$ $when$ $\phi \geq 0$ and the initial condition $x(t_0) \in X_S$. The first condition ensures that ϕ can always affected by u_0, while the

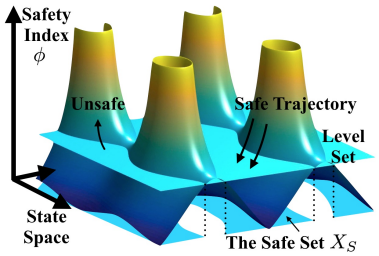


Fig. 9: Illustration of the safety index and the safe set

second condition ensures that the modified trajectory always belongs to X_S. It is shown in [14] that such a function ϕ exists for any safe set described by $X_S = \{x : d(x) \geq d_{min}\}$ if the function $d(\cdot)$ is smooth. Moreover, it is shown in [15] that the safety index $\phi = D - d_j^2(x) - \alpha\dot{d}_j(x)$ $(j = 1, 2)$ has the desired performance for systems with state equation (2), where $D > d_{min}^2$, and $\alpha > 0$ are constants. To ensure safety, the control input of the automated vehicle must be chosen from the set of safe control $U_S(t) = \{u_0(t) : \dot{\phi} \leq -\eta_0 \; when \; \phi \geq 0\}$ where $\eta_0 \in \mathbb{R}^+$ is a safety margin. By (2), the set of safe control when $\phi \geq 0$ can be written as

$$U_S(t) = \{u_0(t) : L(t)\, u_0(t) \leq S(t)\} \tag{16}$$

where $L(t) = \frac{\partial\phi}{\partial x_0}\mathcal{B}$, $S(t) = -\eta_0 - \sum_{j\in H}\frac{\partial\phi}{\partial x_j}\dot{x}_j - \frac{\partial\phi}{\partial x_0}f$ and $\dot{x}_j$ is the prediction made by the trajectory predictor.

If the baseline control input $u_0(t)$ is anticipated to violate the safety constraint, the safety controller will map it to the set of safe control $U_S(t)$ according to the following quadratic cost function

$$u_0/U_S = \min_{u\in U_S\cap\Omega} J_0(u) = \frac{1}{2}(u - u_0)^T W (u - u_0) \tag{17}$$

where W is a positive definite matrix and defines a metric in the vehicle's control space. To obtain optimality, W should be close enough to the metric imposed by the cost function J in (14) and (15), e.g. $W \approx d^2J/du_0^2$ where J is convex in u_0. In the lane following mode, if the lateral deviation f_0^9 is large due to obstacle avoidance and the safety controller continues to generate turning signal $\theta \neq 0$, the vehicle will enter the lane changing mode.

V. Simulation

Simulation studies are conducted based on the algorithms discussed above. The automated vehicle is assumed to start with the rightmost lane with the desired longitudinal speed $v_r = 30m/s \approx 67mile/h$. The sampling time in the simulation is $0.05s$. Two initial objectives are considered: 1) following the lane and 2) changing lane to the left. Under each objective, three different cases are considered, to be elaborated below.

A. Objective 1: Lane Following

1) Case 1 - Stationary Obstacle: Figure 10 shows the case when the automated vehicle suddenly noticed a stationary obstacle 40m ahead. The safety controller went active. By mapping the baseline input u_0 to U_S in (17), the command for deceleration and turn was generated. Then the automated

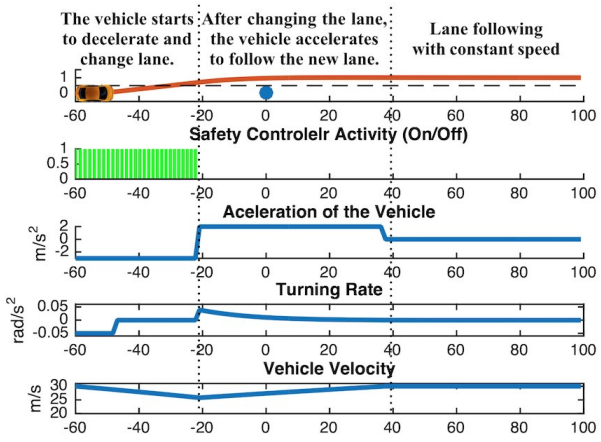


Fig. 10: Case 1 in Objective 1: Reaction of the automated vehicle when there is a stationary obstacle 40m ahead.

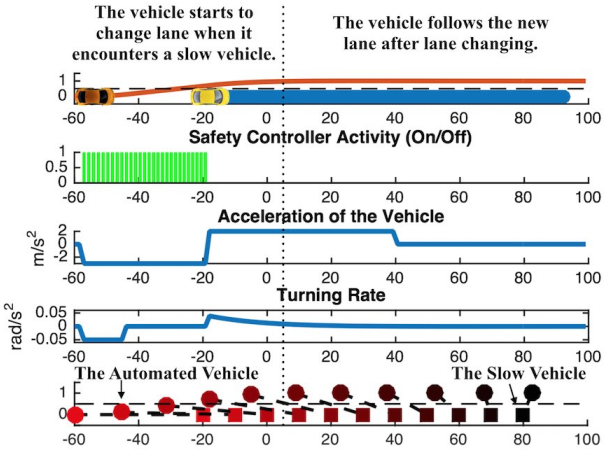


Fig. 11: Case 2 in Objective 1: Reaction of the automated vehicle when it encountered a slow vehicle in the front.

vehicle slowed down and changed lane to the left to avoid the obstacle. After lane changing, the vehicle accelerated to the desired speed again.

2) Case 2 - Slow Front Vehicle: Figure 11 shows the case when the front vehicle was too slow. To illustrate the interaction, the trajectories of both vehicles are down sampled and shown in the last plot in Fig.11, where circles represent the automated vehicle and squares represent the slow vehicle. Different colors correspond to different time steps, the lighter the earlier. At the beginning, since it was not possible for the automated vehicle to keep the desired speed behind the slow car, it started to change lane to the left. After changing the lane, it overtook the slow vehicle.

3) Case 3 - Fast Cut-in Vehicle: Figure 12 shows the scenario when the automated vehicle was overtook by a fast vehicle. When the automated vehicle observed large lateral velocity from the fast vehicle, it predicted that the fast vehicle would change lane. Under the command from the safety controller, the automated vehicle slowed down. After the lane changing of the fast vehicle, the automated vehicle accelerated again to meet the desired speed and keep

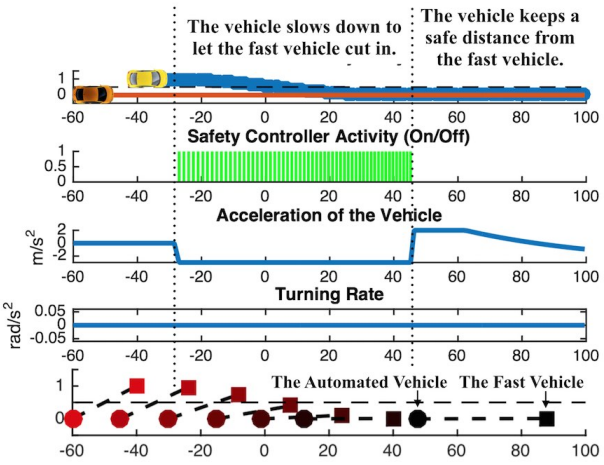


Fig. 12: Case 3 in Objective 1: Reaction of the automated vehicle when it was overtook by a fast vehicle.

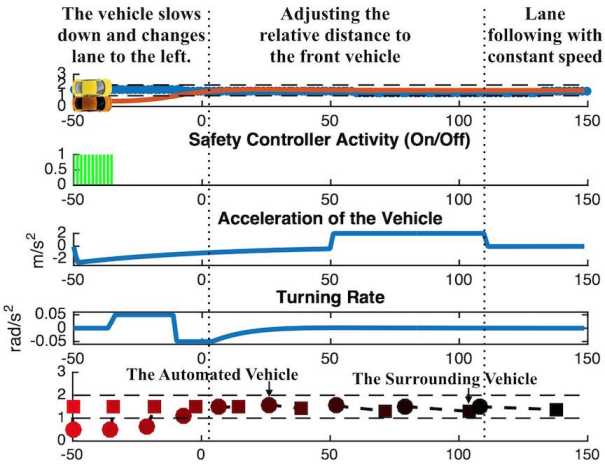


Fig. 13: Case 1 in Objective 2: Behavior of the automated vehicle when there was a vehicle next to it in the target lane.

a safe headway to the fast vehicle.

B. Objective 2: Lane Changing

In this simulation, the trajectory of the surrounding vehicle is manually controlled, in order to test the real time interactions.

1) Case 1 - A Vehicle Moving Side by Side in the Target Lane: Figure 13 shows the case when the vehicle in the target lane was traveling next to the automated vehicle with approximately same speed. It was not safe to change lane in this case. Then the automated vehicle slowed down to create a gap between the two vehicles. When the distance between the two vehicles was big enough, the automated vehicle then changed to the target lane. After adjusting the relative distance to the front vehicle, the automated vehicle then followed the new lane at constant speed.

2) Case 2 - A Slowing Down Vehicle in the Target Lane: Figure 14 shows the case when the vehicle in the target lane was slowing down. At first, the automated vehicle tried to use the strategy in case 1. However, when it noticed that the

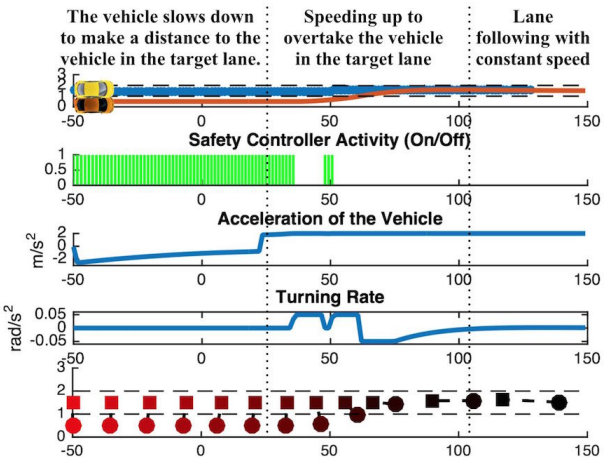


Fig. 14: Case 2 in Objective 2: Behavior of the automated vehicle when the vehicle in the target lane was slowing down.

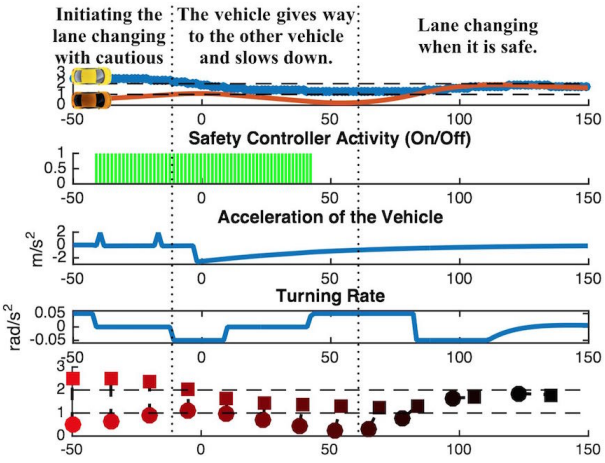


Fig. 15: Case 3 in Objective 2: Behavior of the automated vehicle when another vehicle changed to the target lane simultaneously from the opposite direction.

yellow car also slowed down, it then sped up to overtake the yellow car.

3) Case 3 - Simultaneous Lane Changing from Opposite Directions: Figure 15 shows the case when another vehicle changed to the target lane simultaneously with the automated vehicle, but from the opposite direction. At the beginning, the yellow car was anticipated to follow its lane. Hence it was safe for the automated vehicle to change lane. When the lateral velocity of the yellow car became larger, the probability of B_3 went up and a possible future collision was anticipated. Then the safety planner went active. When the yellow car was about to cross the lane boundary, the automated vehicle turned back to its previous lane and slowed down. The automated vehicle finally changed lane using the strategy in case 1: slowing down first and changing lane when the distance between the two vehicles was big enough.

VI. Conclusion and Future Work

In this paper, the design of layer 2 in the ROAD system was discussed. The multi-agent traffic model was proposed and an optimal control problem was formulated for vehicle trajectory planning. To solve the problem, the behaviors of surrounding vehicles was identified and their future trajectories was predicted. Based on the predictions, the optimal control problem was solved online using an unique architecture: a parallel combination of a baseline planner which solved the problem without the safety constraint and a safety planner which took care of the safety constraint online. The proposed algorithms were verified in the simulations.

In the future, more real world data will be collected to refine the classifier and empirical models. The determination method of the objectives and its integration with the current planner structure will be studied. The optimality of the proposed parallel planner structure will also be investigated. Integrated road simulation with high fidelity vehicle models and curved roads will be conducted.

References

[1] L. D. Burns, "Sustainable mobility: a vision of our transport future," *Nature*, vol. 497, no. 7448, pp. 181–182, 2013.

[2] (2015) Driverless car market watch. [Online]. Available: http://www.driverless-future.com

[3] P. E. Ross, "California to issue driving licenses to robots," *Spectrum, IEEE*, May 2014.

[4] C. Urmson, J. Anhalt, D. Bagnell, C. Baker, R. Bittner, M. Clark, J. Dolan, D. Duggins, T. Galatali, C. Geyer *et al.*, "Autonomous driving in urban environments: Boss and the urban challenge," *Journal of Field Robotics*, vol. 25, no. 8, pp. 425–466, 2008.

[5] "Volvo collision avoidance features: Initial results," Highway Loss Data Institute, Tech. Rep. 5, April 2012.

[6] K. G. Vamvoudakis, P. J. Antsaklis, W. E. Dixon, J. P. Hespanha, F. L. Lewis, H. Modares, and B. Kiumarsi, "Autonomy and machine intelligence in complex systems: A tutorial," in *2015 American Control Conference*. IEEE, 2015, pp. 5062–5079.

[7] A. Doniec, R. Mandiau, S. Piechowiak, and S. Espié, "A behavioral multi-agent model for road traffic simulation," *Engineering Applications of Artificial Intelligence*, vol. 21, no. 8, pp. 1443 – 1454, 2008.

[8] W. Ren, R. W. Beard *et al.*, "Consensus seeking in multiagent systems under dynamically changing interaction topologies," *IEEE Transactions on Automatic Control*, vol. 50, no. 5, pp. 655–661, 2005.

[9] S. J. Anderson, S. C. Peters, T. E. Pilutti, and K. Iagnemma, "An optimal-control-based framework for trajectory planning, threat assessment, and semi-autonomous control of passenger vehicles in hazard avoidance scenarios," *International Journal of Vehicle Autonomous Systems*, vol. 8, no. 2-4, pp. 190–216, 2010.

[10] Y. Gao, A. Gray, H. E. Tseng, and F. Borrelli, "A tube-based robust nonlinear predictive control approach to semiautonomous ground vehicles," *Vehicle System Dynamics*, vol. 52, no. 6, pp. 802–823, 2014.

[11] O. Cappé, E. Moulines, and T. Rydén, *Inference in Hidden Markov Models*. Springer Science & Business Media, 2006.

[12] G. C. Goodwin and K. S. Sin, *Adaptive Filtering Prediction and Control*. Courier Dover Publications, 2013.

[13] C. Liu and M. Tomizuka, "Algorithmic safety measures for intelligent industrial co-robots," in *Proceedings of Robotics and Automation (ICRA), 2016 IEEE International Conference on*. IEEE, 2016, p. to appear.

[14] ——, "Control in a safe set: Addressing safety in human robot interactions," in *Proceedings of the ASME 2014 Dynamic Systems and Control Conference (DSCC)*. ASME, 2014, p. V003T42A003.

[15] ——, "Safe exploration: Addressing various uncertainty levels in human robot interactions," in *2015 American Control Conference*. IEEE, 2015, pp. 465 – 470.