Autonomous Vehicle Social Behavior for Highway Entrance Ramp Management

Junqing Wei, John M. Dolan and Bakhtiar Litkouhi

***Abstract*— "Socially cooperative driving" is an integral part of our everyday driving, hence requiring special attention to imbue the autonomous driving with a more natural driving behavior. In this paper, an intention-integrated Prediction- and Cost function-Based algorithm (iPCB) framework is proposed to enable an autonomous vehicle to perform cooperative social behavior. An intention estimator is developed to extract the probability of surrounding agents' intentions in real time. Then for each candidate strategy, a prediction engine considering the interaction between host and surrounding agents is used to predict future scenarios. A cost function-based evaluation is applied to compute the cost for each scenario and select the decision corresponding to the lowest cost. The algorithm was tested in simulation on an autonomous vehicle cooperating with vehicles merging from freeway entrance ramps with 10,000 randomly generated scenarios . Compared with approaches that do not take social behavior into account, the iPCB algorithm shows a 41.7% performance improvement based on the chosen cost functions.**

I. Introduction

The availability of rapid freeway and highway transportation has strongly contributed to society's progress over the last century. However, in recent decades, traffic congestion on road networks has become a bottleneck for the further development of cities. Autonomous vehicles have shown the potential to lessen this problem by reducing the number of traffic accidents and enhancing the capacity and efficiency of the transportation system. Since the 1980s, autonomous vehicle intelligence has increased from lane centering to actually driving on public roads with lane-changing capability. Nevertheless, in the short term, human-driven vehicles will continue to predominate. For human drivers, an intuitive form of cooperation occurs when another vehicle is nearby, consisting in an estimate of the other driver's intention and a corresponding reaction. Without this ability, in scenarios such as entrance ramps, it is hard for an autonomous robot to behave in what might be termed a socially acceptable way. This will make it difficult for human drivers to understand, predict and cooperate with autonomous vehicles, and may lead to dangerous situations.

Therefore, to enhance autonomous driving in the real world, the decision-making system will benefit from the ability to socially cooperate with human-driven vehicles. In this paper, the importance of social cooperation in the specific instance of dealing with cars merging from entrance ramps is shown. A novel social behavior framework is implemented to exhibit the needed social behavior for this driving scenario.

This work was supported by General Motors through the GM-Carnegie Mellon Autonomous Driving Collaborative Research Laboratory.

Junqing Wei is with the Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA 15213, USA junqingw@cmu.edu

John M. Dolan is with the Department of Electrical and Computer Engineering and Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213, USA

Bakhtiar Litkouhi is with General Motor R&D, Warren, MI 48091, USA

II. Related work

A. Autonomous Driving Systems

Beginning in the late 1980s, a few experimental platforms capable of lane centering and cruise control-level autonomous driving on highways were developed [1], [2]. In 2004-2007, the DARPA Grand Challenge and Urban Challenge provided researchers a practical testing environment to test the latest sensors, computing technology and artificial intelligence algorithms for autonomous driving [3], [4]. In the competition, the autonomous vehicles were able to deal with relatively light human traffic driven by trained competition crews in a closed test field. In 2011, Google released its autonomous driving platforms [5]. The vehicles have each completed over 10,000 miles of autonomous driving on multiple road types and under various traffic conditions. The Google car is capable of dealing with a number of real-world human-driven traffic on public roads. However, these vehicles will not perform as well as human drivers in heavy traffic due to their limited ability to understand and cooperate with surrounding cars as human drivers do with each other.

B. Adaptive Cruise Control

Adaptive Cruise Control (ACC) is one of the most widely deployed advanced driver assist systems [6]. In recent years, lane centering assist has also been developed to enhance human driving comfort and safety on freeways [7]. By integrating these driver assistant systems, a few prototype autonomous driving platforms have been demonstrated by auto manufacturers. Though these commercially viable platforms demonstrate the potential to improve driver's experience and safety, they are mostly capable of limited single-lane highway autonomy. Little effort has been put into cooperative behavior between the single-lane autonomous driving system and surrounding traffic in adjacent lanes.

C. Human Driver Model

Experienced human drivers can, for the most part, understand each other's intentions and smoothly cooperate with one another while driving. Therefore, it is reasonable to utilize a human driver behavior model to control an autonomous vehicle. In the microscopic traffic simulations area, there

are models able to emulate an individual vehicle's behavior [8]. In these models, temporal and some unobservable state information (e.g. intention) of surrounding cars are not used because of the difficulty of data collection and complexity of model training. Therefore, they have difficulty emulating human decisions at the social cooperation level.

D. Motion Planning & Moving Obstacle Avoidance

In an autonomous driving system, a motion planner usually commands the vehicle's lateral and longitudinal movements. The most straightforward way to safely avoid moving obstacles is to regard them as static obstacles cycle by cycle and use the replanning mechanism to react to them in real time. A more advanced approach is to assume moving obstacles will keep constant velocity and heading [9]. In the Urban Challenge, CMU's Tartan Racing team proposed an on-road local planner based on the assumption that moving obstacles will keep constant velocity and also drive along the road [10]. However, none of these assumptions captures the fact that the host autonomous vehicle's movement will affect surrounding vehicles' behavior.

E. Robot Social Behavior

In the human-robot interaction area, researchers have developed planners for robots to navigate in an environment shared with human participants. Kirby et al. proposed a biased oval-shaped occupancy region around humans for use by the robot planner [11]. The associated cost causes the robot to perform more socially acceptabe maneuver by, for example, circumventing to the left of human pedestrians in hallways and keeping a comfortable distance from human participants. However, this approach regards human pedestrians as static obstacles with a specialized cost. This works well for indoor navigation, wherein both pedestrians and the robot move relatively slowly and the robot can easily stop within meters. But for high-speed applications, such as freeway autonomous driving, the plan horizon (proportional to the speed of the car) is usually much longer because of the high inertia of the car. Without a proper prediction of the movement of surrounding cars, the autonomous vehicle cannot make safe and robust decisions in the expected plan horizon.

In the decision theory area, there are frameworks supporting the modeling of robot social behavior, such as POMDP [12]. Using POMDP, Broz et al. introduce human intention as a partially observable state in decision making [13]. One simulation experiment shows that in an intersection scenario, an autonomous vehicle can cooperate with another car by understanding whether it wants to yield or not and perform proper actions. An attractive feature of the POMDP-based approach is that its decision is theoretically optimal. However, because of the computational expense and difficulty in building an accurate probabilistic state transition model, it has limited extensibility. If there is more than one human-driven vehicle to deal with, the difficulty of abstracting the cooperation into a POMDP model increases exponentially. Therefore, this model can only deal with very simple scenarios.

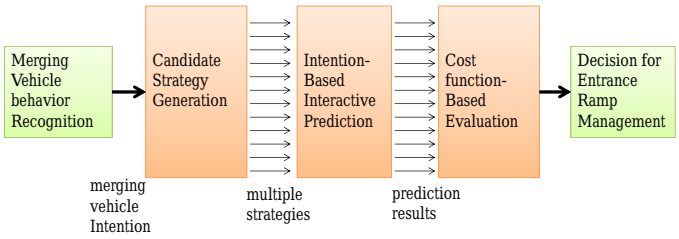


Fig. 1: Framework of the intention-integrated Prediction- and Cost function-Based algorithm (iPCB)

F. Prediction- and Cost function-Based algorithm

In [14], we proposed a Prediction- and Cost function-Based (PCB) algorithm framework for autonomous freeway driving applications in which a Markov Decision Process-based approach to modeling how an autonomous vehicle's behavior affects surrounding agents is developed. The PCB framework was extended via a sampled-based approach to deal with sensor uncertainties and field-of-view constraints [15]

The main contribution of the current paper is to integrate the sample-based PCB algorithm with a Bayesian driving intention recognition model for the autonomous vehicle to perform social behavior. This intention-integrated Prediction- and Cost function-Based algorithm (iPCB) framework is implemented and tested in a scenario of social cooperation with vehicles merging from freeway entrance ramps.

III. iPCB Algorithm Framework

As shown in Figure 1, there are four main modules in the iPCB framework: surrounding vehicle intention estimation, candidate strategy generation, intention-based interactive prediction and cost function-based strategy evaluation. The intention recognition module captures the surrounding vehicle information input from the autonomous vehicle's perception system. It uses a knowledge-based model to output the probability of each surrounding vehicle's intention. The strategy generation module proposes a set of candidate driving strategies for the vehicle to execute. The intention-integrated prediction uses the surrounding vehicles' intentions to predict the future traffic scenarios assuming each of the candidate strategies is applied. The cost function-based evaluation module computes costs for each scenario and sums them together as the strategy cost. The autonomous driving system then chooses the best strategy corresponding to the lowest cost.

IV. Freeway Entrance Ramp Management and iPCB implementation

A. Freeway Entrance Ramp Scenario

Freeway entrance ramp management is chosen in this paper as a difficult scenario in which vehicles (autonomous or not) need to exhibit social behavior, and the iPCB algorithm can be applied. For the entrance ramp management system,

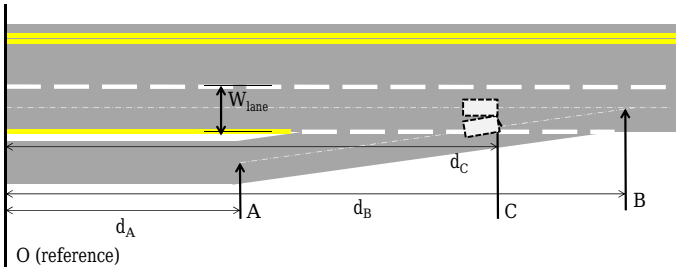


Fig. 2: The abstract representation of an entrance ramp used in the iPCB algorithm

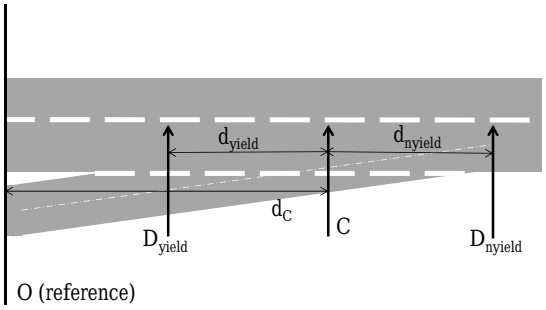


Fig. 3: Important points in generating a prediction model of the merging vehicle behavior

a normal scenario is shown in Figure 2. The host vehicle is driving in the rightmost lane of the freeway. The start point A of the entrance ramp is where the autonomous vehicle begins to consider the merging vehicle. The end point B is where the entrance ramp fully merges onto the main lane. d_A and d_B are the longitudinal distances from a reference point O to points A and B. Point C in Figure 2 is defined as the point at which the interaction between the two vehicles needs to be completed. The position of C, d_C, is computed using Equation 1, in which w_{lane} is the width of the lane, and w_{car} is the width of the merging vehicle.

$$\begin{aligned} d_c &= d_A + (d_B - d_A) \\ &\quad /w_{lane} * (w_{lane} - w_{car}) \end{aligned} \tag{1}$$

After point C, cars should return to their normal lane driving and distance keeping mode. In this paper, all the tests are implemented with parameters $d_A = 40m$, $d_B = 120m$, $w_{lane} = 6m$, $w_{car} = 2m$.

B. Vehicles in Entrance Ramp Management Scenario

In this paper, we only consider the autonomous vehicle's single-lane driving performance, which means the host vehicle (autonomous vehicle) will keep performing lane centering in its current lane. Its sole action is to adjust speed by changing its acceleration a_{host}. The host vehicle's state is given by d_{host}, v_{host} and l_{host}, where d_{host} and v_{host} are respectively the longitudinal distance and speed of the autonomous vehicle, and l_{host} is the lateral position of the host vehicle.

It is assumed that the merging vehicle will merge along a fixed path corresponding to the center line of the entrance ramp. Like the host vehicle, it has observable state d_{merge}, v_{merge} and l_{merge} obtainable from the perception system of the autonomous vehicle. In addition, to represent the autonomous vehicle's understanding of the merging vehicle's behavior, an intention state i_{merge} is included which can be either Yield (Y) or Not Yield (NY).

For other traffic vehicles, including vehicles in adjacent lanes and any vehicles leading or following the autonomous car, d_i, v_i and l_i are used to represent their longitudinal position, speed and lateral distance.

C. Merging Vehicle Intention Recognition

The first step in the iPCB algorithm framework is to estimate the merging vehicle intention I and use probability $p(I = Y)$ or $p(Y)$ and $p(I = N)$ or $p(N)$ to capture the uncertainty of the estimation. The intention recognition mechanism proposed in this paper is based on Bayes' theorem, as shown in Equation 2. As no prior knowledge of the intention of the merging vehicle is available, the probability of yield $p(Y)$ and not yield $p(N)$ are both set to 0.5.

$$\begin{aligned} p(Y|B) &= p(B|Y)p(Y)/[p(Y)p(B|Y) + p(N)p(B|N)] \\ &= p(B|Y) \times 0.5/[0.5p(B|Y) + 0.5p(B|N)] \\ &= p(B|Y)/[p(B|Y) + p(B|N)] \end{aligned} \tag{2}$$

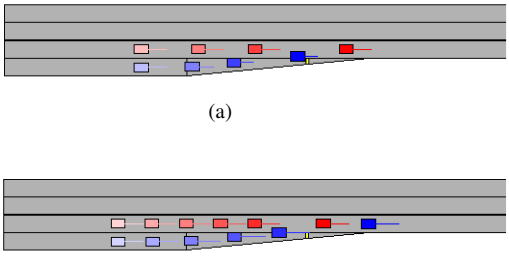
In Equation 2, $p(Y|B)$ is the probability of an intention to yield Y given an observed behavior B, and $p(B|Y)$ and $p(B|N)$ are respectively the probabilities of that same behavior given intention Y or N.

To estimate intent of the merging vehicle, one of the most relevant factors is its acceleration. A decelerating vehicle is more likely to intend to yield to us, while this is less likely for an accelerating vehicle. Therefore, the acceleration is computed from the velocity measurement of the merging vehicle, as shown in Equation 3, in which $v(t)$ is the observed merging vehicle velocity at time t, t_{filter} is the filtering horizon.

$$acc(t) = (v(t) - v(t - t_{filter}))/t_{filter} \tag{3}$$

A merging vehicle behavior model $B|I$ is created representing the behavior B (acceleration or deceleration) expected from the merging vehicle given a particular intention I. If the merging vehicle intends to yield to the host vehicle, it will tend to merge onto the main lane with a proper forward distance to the host vehicle. Otherwise, it will try to get ahead of the host vehicle and keep a reasonable backward distance when it cuts in. Figure 3 shows a few important points for this model: C is the end of the interaction region, and d_{yield} and d_{nyield} are respectively where the merging vehicle is supposed to be when the host vehicle arrives at C for the yield and not-yield cases.

$$\begin{aligned} d_{yield} &= d_C - (d_{min} + k_v v_{host}) \\ d_{nyield} &= d_C + (d_{min} + k_v v_{host}) \\ \Delta t &= (d_{yield,nyield} - d_{merge})/v_{merge} \\ &\quad - (d_C - d_{host})/v_{host} \\ acc_{merge} &= c_{gain}\Delta t \end{aligned} \tag{4}$$



(a)

(b)

Fig. 4: Simulation results of the merging vehicle prediction model (a) given the intention to yield (b) given the intention not to yield

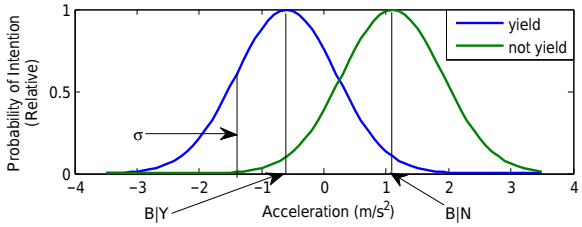
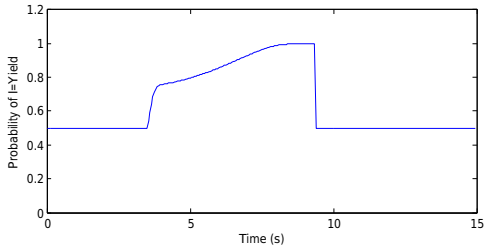


Fig. 5: A Gaussian distribution is applied to the deterministic model $B|I$ to get $p(B|I)$

In above, d_{yield} is the distance from C to D_{yield}, d_{nyield} the distance from C to D_{nyield}. They are computed using Equation 4, which is a widely used model of human driver behavior [8]. d_{min} is the minimum desired distance to the leading vehicle, and k_v is a gain causing the desired distance to grow with the host vehicle's speed. d_{merge} and d_{host} are the positions of the merging and host vehicles at any given time. The desired control command for the merging vehicle acc_{merge} is computed using a proportional controller applied to the difference between the arrival times of the merging and autonomous vehicles, as shown in Equation 4, where Δt is the time difference between the two vehicles' arrivals at C, and c_{gain} is the proportional gain converting the time difference into the acceleration command of the merging vehicle.

Simulation results are shown in Figure 4. In Figure 4a, the merging vehicle is given an intention of Yield, so it gradually applies deceleration and eventually merges onto the main lane with the desired distance to the leader. In Figure 4b, the merging vehicle is commanded to have a Not Yield intention, so it accelerates to merge in front of the host vehicle. This model emulates the behavior of a merging vehicle performing speed adjustment to enact its intention.

The foregoing uses a deterministic model of $B|I$. We extend this model by superimposing a Gaussian distribution with its peak at the acc_{merge} values for $B|Y$ and $B|N$ and standard deviation $\sigma = 0.8m/s^2$, as shown in Figure 5. The above gives probabilistic estimates $p(B|Y)$ and $p(B|N)$ of the merging vehicle acceleration given the two possible intentions and is used to capture the uncertainty-based deviation of the merging vehicle's behavior from the



(a)

(b)

Fig. 6: Real-time intention recognition result: (a) estimated probability of intention 'Yield' (b) corresponding scenarios at t=4.0s and t=6.0s.

$B|I$ model. Integrating $p(B|I)$ with the Bayesian rule in Equation 2 yields a merging vehicle intention probability estimator.

To verify the intention estimator's performance, a test was implemented in simulation. In the test, the merging vehicle was given a certain intention ($I = Y$ or $I = N$) and followed the model $B|I$. The real-time intention probability recognition result is shown in Figure 6a: as the vehicle just begins to perform some minor adjustment of its speed around $t = 4.0s$ (see Figure 6b), the intention estimator is able to capture this behavior hint and output the probability of intention. The intention estimation result is very ambiguous at the beginning, but as the merging vehicle gets closer to point C around $t = 6.0s$, the confidence of the estimation result increases sharply.

D. Candidate Strategy Generator

The command for the autonomous vehicle in the freeway entrance ramp management system is an instantaneous velocity command, as mentioned in Section IV-B. However, when the vehicle is looking for the best immediate action, it also needs to consider a series of future control commands, i.e., a velocity profile for the next $t_{predictLength}$ seconds. Due to the real-time requirement and limited computation power, only a limited number of strategies can be searched and evaluated. Therefore, a discretization of the strategy set is used, as shown in Figure 7. In this discretization, velocity profiles are represented by three parameters: t_{adjust}, the total speed adjustment time; and a_{first} and a_{second}, the acceleration amplitude for respectively the first and second half of t_{adjust}. In this paper, $t_{predictLength}$ is chosen to be 10.0 seconds and t_{adjust} is discretized into two values, $3.0s$ or $5.0s$, based on [14]. There are 13 different acceleration options covering the range from $-3.0m/s^2$ to $2.0m/s^2$. The candidate strategy generator outputs a total of 378 strategies, which allows the iPCB algorithm to replan fast enough for the real-time implementation and vehicle test.

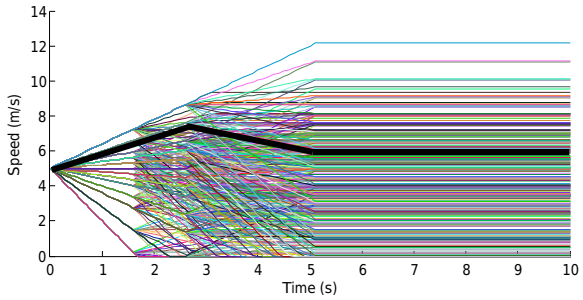


Fig. 7: Candidate strategies generated by the discretized strategy generator: the velocity is being changed from a start to an end value via an adjust period (t_{adjust}) that is split into two equal constant-acceleration segments. One example strategy highlighted in the figure is to accelerate for $2.5s$ with $1m/s^2$ then decelerate with $-0.8m/s^2$ for $2.5s$.

E. Interactive Prediction

After the candidate strategy generation, a prediction engine is used to simulate future scenarios for each strategy. For surrounding vehicles, including the leading or following vehicle of the autonomous car, we use the following prediction model:

$$\Delta d_l = d_l - d_{v(i)}$$
$$a_{v(i)} = \begin{cases} \mu_{free}(v_{lim} - v_{v(i)}) & \text{if } \Delta d_l > 100 \\ \mu_{dk}(d_d - d_l) + \mu_{dkv}(v_l - v_{v(i)}) & \text{if } \Delta d_l < 100 \end{cases} \quad (5)$$

where Δd_l is the distance to vehicle v_i's leader, μ_{free} is the proportional gain for the vehicle to gradually approach the speed limit when there are no obstacles in front of it, μ_{dk} is the proportional gain of the distance keeping controller, and μ_{dkv} is the proportional gain on the velocity difference between the leader and the vehicle v_i.

The intention of a merging vehicle from the intention recognition algorithm will be used for more accurate prediction. The model described in Section IV-C to generate $B|I$ is used. The following preprocessing step is used with the $B|I$ model. When the merging car's yield/not yield decision is obvious enough, the input I will be overridden, as shown in Equation 6, in which Δt is computed using Equation 4. For instance, if the merging vehicle is much slower than and far behind the host vehicle, it will almost surely yield to the host vehicle. In these cases, only one intention will be considered in prediction.

$$I = \begin{cases} Y, & \text{if } \Delta t > 3.0 \\ N, & \text{if } \Delta t < 3.0 \end{cases} \quad (6)$$

An important effect of this override mechanism is that even when merging vehicle intention ambiguity exists at the beginning of the prediction, if the host vehicle performs a certain behavior, it has the ability to force the merging vehicle to converge to a predictable decision.

The interactive prediction module gives the iPCB framework the ability to predict how surrounding vehicles will react to the host vehicle's strategy, which is a key factor in enabling the autonomous vehicle to socially interact with them. Compared with the prediction mechanism of most other motion planning and distance keeping algorithms, which are based on the assumption that surrounding vehicles will keep constant velocity and will not be affected by the host vehicle's behavior, the proposed prediction model is also more accurate.

F. Cost Function-Based Evaluation

The prediction module generates a sequence of predicted scenarios from the current time $t = 0$ to $t = t_{predictLength}$ with constant time step Δt. Each entrance ramp scenario is represented by the longitudinal positions, speeds and lateral positions of the host, merging and surrounding vehicles' $d_{host,merge,i}$, $v_{host,merge,i}$, $l_{host,merge,i}$. A cost for each scenario C_{sce} is then computed, which consists of four different cost functions based on the algorithm proposed in [14].

- Progress cost: The progress cost represents how well a strategy does in finishing a given task by penalizing those strategies which take longer to finish the task. The goal of the distance keeper is to keep a desired distance $d_{desired}$ to its leader, which is computed in Equation 7, in which v is the current velocity of the vehicle, d_{min} is the distance to the leader when the vehicle is stationary, and k_v is the gain of the desired distance increase corresponding to v.

$$d_{desired} = d_{min} + k_v v \quad (7)$$

- Comfort cost: While driving a car, human drivers will generally try to avoid large accelerations for greater comfort. Therefore, a comfort cost $C_{comfort}$ is included to represent this.
- Safety cost: The safety cost of a scenario consists of two terms: the clear distance cost $C_{distance}$ and the braking distance cost C_{brake}. The clear distance cost $C_{distance}$ penalizes moving too close to surrounding vehicles. However, this cost is not informative enough for us to avoid collision in some situations, since it does not consider the vehicles' relative velocities. Therefore, another safety cost based on the braking distance difference Δd_{brake} between two vehicles is also considered.
- Fuel consumption cost: The fuel consumption cost is proportional to the fuel usage as estimated by the CMEM statistical model [16].

The parameters and the shapes of these cost functions are selected based on case tests and statistical tests in a simulator with simulated traffic vehicles [14]. The total cost of a scenario is the weighted sum of all these costs:

$$C_{sce} = \mu_1 C_{progress} + \mu_2 C_{comfort} + \mu_3 C_{safety} + \mu_4 C_{fuel} \quad (8)$$

By summing the scenario costs together, the cost for the i^{th} strategy given an intention is generated as follows:

$$C_{str(i)}|I = \sum_{t=0}^{t_{Predict}} (C_{sce(i,t)}|I)$$
$$C_{str(i)} = p(I = Y)C_{str(i)}|Y + p(I = N)C_{str(i)}|N \quad (9)$$

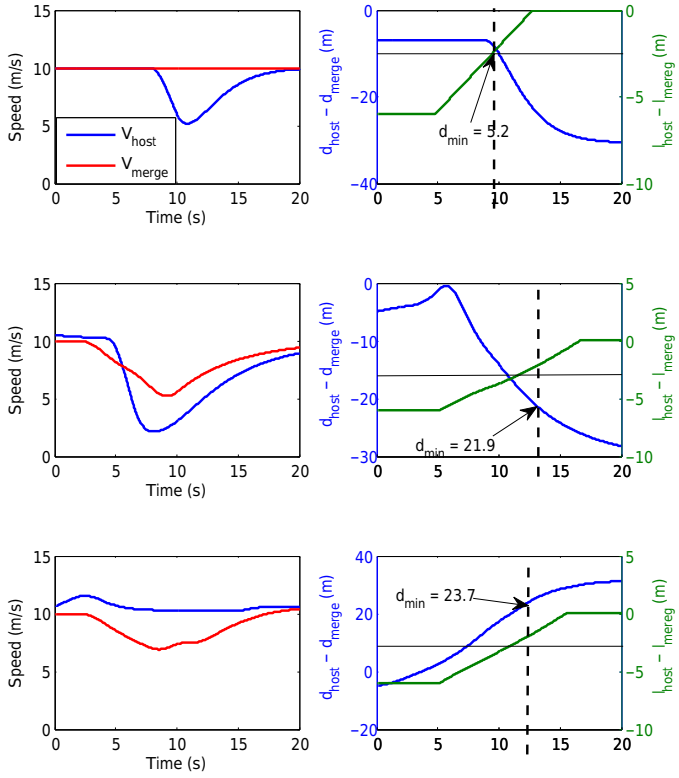


Fig. 8: Comparison of different approaches in dealing with merging vehicles: adaptive cruise control (ACC) (top), geographic information integrated ACC (geoACC) (middle) with scenario that the merging vehicle want to yield, iPCB algorithm (bottom). Left figures are the speed plots of the merging and host vehicle. Right figures show the lateral and longitudinal distances between the two vehicles.

To handle the uncertainty in the state variable I, a sample-based approach proposed in [15] is used to compute the expectation of the strategy cost $C_{str(i)}$. Finally, the best freeway entrance ramp management strategy is selected based on the lowest accumulated cost, which is computed using Equation 9.

V. Performance Evaluation

To evaluate the performance of the proposed iPCB framework for autonomous freeway driving entrance ramp management, case tests focused on qualitative analysis were implemented. A statistical test which ran the algorithm through 10,000 randomly generated scenarios was used for quantitative performance evaluation.

A. Case test

Figure 8 shows the speeds (host and merging) and lateral and longitudinal distances between vehicles for the autonomous vehicle performing single-lane autonomous driving with three different algorithms. Because the adaptive cruise control system (top plots) only considers vehicles in the same lane as the host vehicle, it does not react to the merging vehicle until it crosses the lane divider, which causes emergency braking and an uncomfortably small distance between vehicles ($5.2m$).

An extension (geoACC) of the basic ACC for better handling of merging vehicles would be to include geographical information such as the shape of the road and entrance ramp and some rules governing whether the autonomous vehicle should yield. If the merging vehicle arrives earlier, the autonomous driver will decide to yield, as indicated by Equation 10, in which D_{host} is the host vehicle decision, and Δt is computed using Equation 4. Otherwise, it will try get in front of the merging vehicle. If the decision is Y (yield), then the autonomous vehicle will perform distance keeping on the merging vehicle while it is still on the entrance ramp. If it is N (not yield), it will ignore the vehicle on the ramp.

$$D_{host} = \begin{cases} N, & \text{if } \Delta t > 0 \\ Y, & \text{if } \Delta t \leq 0 \end{cases} \tag{10}$$

However, if the host and merging vehicles arrive at around the same time, these rules may result in oscillation between the yield and not yield decisions. Another failure case of the geoACC algorithm is when the merging vehicle wants to perform some social interaction which the host vehicle does not understand, as shown in the middle plots of Figure 8. At the beginning, the host vehicle decides to yield to the merging vehicle because $\Delta t \leq 0$. However, the merging vehicle wants to yield to the host vehicle, as well. The result is that the merging vehicle decreases its speed, and the host vehicle applies distance keeping to it, further decreasing its speed. This "social misunderstanding" causes the host vehicle to brake very hard to keep a safe distance to the merging vehicle, but this causes a potentially unsafe situation on highways. For a human driver, as long as it is understood that the merging vehicle wants to yield, the host vehicle decision will switch from yield to not yield in most situations.

Compared to these two approaches based on the current ACC system, the iPCB algorithm can actively perform a behavior that attempts to convey its preference to the other agent in the social cooperation. In the test, the most ambiguous scenario is created, in which the host and merging vehicles have the same longitudinal coordinate and speed, and the merging vehicle is told to have an intention to yield ($I = Y$). In the bottom iPCB algorithm plots of Figure 8, as soon as the merging vehicle performs an intentional behavior (deceleration to yield to the host vehicle), the host vehicle understands it and begins to accelerate slightly to clearly show the merging car its cooperative behavior. This makes the speed variation of both vehicles much smaller and the distance between vehicles when the merging vehicle cuts in very close to the desired distance keeping distance $d_{desired}$.

In summary, the iPCB algorithm performs in the most reasonable and social friendly way among the tested approaches by interacting with merging vehicles on entrance ramps. It increases the smoothness of the velocity adjustment and also keeps the distance between merging and autonomous vehicles in a safe range.

TABLE I: Parameter ranges for statistical tests

Parameter	Min	Max
$d_{merge}(m)$	-60.0	20.0
$v_{merge}(m/s)$	5.0	15.0
i_{merge}	Y or N	
$d_{host}(m)$	-60.0	20.0
$v_{host}(m/s)$	5.0	15.0

TABLE II: Statistical test result

	ACC	geoACC	iPCB
C_{ave}	52.50	56.61	30.58
C_{safety}	7.51	6.56	2.45
C_{dk}	20.73	20.55	21.94
C_{acc}	4.84	6.56	0.66
N_{danger}	52	62	9

B. Statistical test

In the case tests, the iPCB algorithm's general ability to perform social behavior was verified. A statistical test was then implemented in simulation to analyze its ability to deal with a wide variety of different entrance ramp management scenarios. The simulation for each algorithm was run 10,000 times to get a more accurate statistical result. In these tests, the merging vehicle is simulated using the model described in Section IV-C. As shown in Table I, there are 5 variables with uniform distribution that represent the initial condition of a given scenario.

Table II shows the result of this test, where C_{ave} is the average strategy cost; C_{safety}, C_{acc}, C_{dk} are respectively the safety, comfort and distance keeping progress cost; and N_{danger} is the number of cases in which the vehicles needed to apply hard braking (with deceleration larger than $3m/s^2$) to avoid an accident.

The iPCB algorithm clearly has the lowest strategy cost, which means the quality of the decision making at entrance ramps is improved. Compared with the ACC and geoACC approaches, the safety and acceleration costs are reduced considerably, meaning the control of the vehicle is smoother and safer. The number of potentially unsafe scenarios is also greatly reduced by using the iPCB algorithm due to its ability to react earlier to merging vehicles based on their intention. This test also verifies that the iPCB algorithm framework is beneficial across a wide range of entrance ramp scenarios.

VI. Conclusion

In this paper, a robot social behavior framework intention-integrated Prediction- and Cost function-Based algorithm is proposed. Using this algorithm, the robot is able to understand the other agent's intention while considering the uncertainty of the intention estimation. The framework enables the autonomous robot to predict how other agents will react to its own behavior. In this way, the autonomous robot's decisions will be more "sociable", balancing its own objectives with those of other participants.

The iPCB algorithm was implemented for the application of autonomous vehicle entrance ramp social behavior. In simulation, case tests verified that the iPCB algorithm will lead the autonomous vehicle to perform more cooperative driving. Statistical tests over 10,000 randomly generated entrance ramp scenario initial states show an overall decrease of 41.7% in a cost function representing cooperation smoothness and safety.

To further improve autonomous vehicle social behavior on entrance ramps, real data should be collected to better formulate the intention estimation and prediction model. The model will also be extended to deal with different road geometries and more complicated surrounding vehicle traffic. Road tests will also be performed on the Carnegie Mellon University autonomous driving platform to test its performance in the real world.

References

[1] E. D. Dickmanns, "Vehicles capable of dynamic vision: a new breed of technical beings?," *Artificial Intelligence*, vol. 103, pp. 49–76, Aug. 1998.

[2] D. Pomerleau, "Alvinn: An autonomous land vehicle in a neural network," in *Advances in Neural Information Processing Systems 1* (D. Touretzky, ed.), Morgan Kaufmann, 1989.

[3] C. Urmson, J. Anhalt, D. Bagnell, C. Baker, R. Bittner, M. Clark, J. Dolan, D. Duggins, *et al.*, "Autonomous driving in urban environments: Boss and the urban challenge," *Journal of Field Robotics Special Issue on the 2007 DARPA Urban Challenge, Part I*, vol. 25, pp. 425–466, June 2008.

[4] S. Thrun, M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, G. Hoffmann, *et al.*, "Stanley: The robot that won the darpa grand challenge," *The 2005 DARPA Grand Challenge*, pp. 1–43, 2007.

[5] J. Markoff, "Google cars drive themselves, in traffic," *The New York Times*, vol. 10, p. A1, 2010.

[6] A. E. Ardalan Vahidi, "Research advances in intelligent collision avoidance and adaptive cruise control," *IEEE Transactions on Intelligent Transportation Systems*, vol. 4, no. 3, pp. 143–153, 2003.

[7] P. Batavia, *Driver Adaptive Lane Departure Warning Systems*. PhD thesis, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, September 1999.

[8] K. I. Ahmed, *Modeling Drivers' Acceleration and Lane Changing Behavior*. PhD thesis, Massachusetts Institute of Technology, Feb 1999.

[9] P. Fiorini, *Robot motion planning among moving obstacles*. PhD thesis, Citeseer, 1995.

[10] D. Ferguson, T. Howard, and M. Likhachev, "Motion planning in urban environments: Part i," in *Proceedings of the IEEE/RSJ 2008 International Conference on Intelligent Robots and Systems*, September 2008.

[11] R. Kirby, *Social Robot Navigation*. PhD thesis, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, May 2010.

[12] A. Cassandra, "A survey of pomdp applications," in *Working Notes of AAAI 1998 Fall Symposium on Planning with Partially Observable Markov Decision Processes*, pp. 17–24, 1998.

[13] F. Broz, *Planning for Human-Robot Interaction:Representing Time And Human Intention*. PhD thesis, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, December 2008.

[14] J. Wei, J. Dolan, and B. Litkouhi, "A prediction-and cost function-based algorithm for robust autonomous freeway driving," in *Intelligent Vehicles Symposium (IV), 2010 IEEE*, pp. 512–517, IEEE, 2010.

[15] J. Wei, J. Dolan, J. Snider, and B. Litkouhi, "A point-based mdp for robust single-lane autonomous driving behavior under uncertainties," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pp. 2586–2592, IEEE, 2011.

[16] M. Barth, F. An, T. Younglove, G. Scora, C. Levine, M. Ross, and T. Wenzel, "Comprehensive modal emission model (cmem), version 2.0 users guide," *University of California, Riverside*, 2000.