Semiautonomous Longitudinal Collision Avoidance Using a Probabilistic Decision Threshold

Jeffrey Johnson Yajia Zhang Kris Hauser
School of Informatics and Computing, Indiana University at Bloomington
{jj56,zhangyaj,hauserk}@indiana.edu

***Abstract*— Automated emergency maneuvering systems can avoid or reduce the severity of collisions by taking control of a vehicle away from the driver during high-risk situations. The choice of when to switch to emergency control is challenging in the presence of uncertain information (imperfect sensors, road conditions, uncertain object behavior, etc.) and many dynamic obstacles. This paper considers longitudinal collision avoidance problems for a vehicle traveling along a known path. A probabilistic decision threshold framework is presented in which the user's control is overridden if the probability that it would lead the system into an unsafe state exceeds some threshold. We apply the technique to collision imminent braking for obstacles traveling along the vehicle's path, and present preliminary results extending the technique to the scenario of obstacles crossing the vehicle's path.**

I. Introduction

Over 6.3 million automobile crashes occurred in the U.S. in 2007, including 1.8 million injury crashes and 37,435 fatalities at a cost of hundreds of billions of dollars [9]. Although the numbers of injuries and fatalities per traveled mile have decreased significantly due to advances in passive safety equipment (seat belts, air bags, stability control, etc) over the last four decades, these numbers have leveled off over the last two decades. Semiautonomous active safety systems, which override driver control of the vehicle in emergency situations, are one promising approach to achieving significant further improvements in safety.

These systems must distinguish between emergency scenarios, assess collateral impact of collision-mitigating or collision-avoidance strategies, and understand the behavior of drivers, including the drivers of other cars, in response to the car's actions. Also, they must carefully balance keeping the driver feeling "in control" and taking necessary deviations from the driver's actions (or inaction) for increased safety. It is also likely that too much automation could increase the risk of inattentive or careless driving and also increase liability for auto manufacturers; hence a *minimal necessary interference* principle should be applied to the design of semiautonomous safety systems at least for the forseeable future.

This paper applies this principle to the prevention of longitudinal collision, specifically, rear end collisions during single-lane driving and transverse collisions during intersection crossing. Given a known path of the vehicle but unknown speed and driver input, the problem is applying longitudinal controls (accelerating and braking) only when necessary to avoid collision with moving or static objects in the environment. Uncertainty is a major challenge in the driving environment due to noisy distance readings; unknown behavior of the object; errors in speedometer readings due to tire wear and environmental factors; and unknown stopping time due to brake wear and road surface characteristics. In the context of rear-end collision avoidance we apply a probabilistic decision thresholding technique that activates control when the risk of collision exceeds some threshold.

Assuming that the vehicle tracks a state distribution using an Extended Kalman Filter, we used Monte-Carlo simulations to evaluate the technique's performance on a suite of scenarios including dry and wet pavement, static and braking obstacles, and false positives and negatives in object detection. We aggregated four performance metrics — collision velocity, completion time relative to an ideal driver, stop distance relative to the obstacle, and jerkiness — across 10 scenarios into a risk index, which quantifies the overall severity of collisions, and an interference index, which quantifies the overall disturbance to the human driver. Plotting these indices demonstrates a clear tradeoff between increased interference and increased safety as the activation threshold is varied.

We also describe first steps toward solving the intersection crossing problem in which the car accelerates or decelerates in order to avoid collision with cars in opposing lanes. We introduce a conservative reachable set computation in the vehicle's position/velocity/time space for an arbitrary number of lanes of traffic in a fully observable environment. Using this computation we intend to address both accelerating and braking under uncertainty in a tractable manner using the same probabilistic decision threshold technique.

II. Related Work

Collision-imminent braking systems in some existing Volvo and Mercedes-Benz models use a variety of sensors to detect collision-imminent scenarios and apply brakes to reduce the severity of a crash. We are interested in extrapolating collision-*imminent* braking to its inevitable conclusion: collision-*prevention* braking.

Autonomous driving has recently become a tremendously active field of research, inspired by major successes such as the DARPA Grand Challenge [12]. A Google system that has logged over 140,000 autonomous miles, including 1000 miles without intervention from the human driver [10]. Despite these major advances, there is still a major gap between these systems and human drivers. Although human drivers occasionally err, they are extremely reliable in general: a fatality crash occurs approximately once in 100 million miles driven [9]. Even if an autonomous vehicle relies on the human driver input once every 10,000 miles, the driver must be attentive 99.99% of the time for the system to perform as well as a human alone!

By contrast, active safety systems take control of the vehicle only when emergency happens or potential accident is foreseen to mitigate or avoid the consequences of an accident. There are several strategies for choosing when to override driver control. A longitudinal collision-mitigation braking strategy was described by Hillenbrand et al (2006) that gradually applies stronger braking as the collision boundary is approached, which smoothes the control output and copes some uncertainty [5]. Anderson et al (2009) present a 2D hazard avoidance scheme based on model predictive control which allows varying levels of autonomy based on risk assessed by control magnitudes [2]. Our approach introduces the additional considerations of uncertainty which provides a more natural definition of risk. Karlsson et al (2004) introduced a statistical decision rule that applies the brake if the probability of impact is greater than some threshold α [7]. This approach is advantageous in the presence of uncertainty. Similar thresholding techniques were applied to autonomous driving in environments mapped using 2D range finders [1]. We generalize this approach in this research to both acceleration and braking using a probabilistic minimum necessary interference criterion that treats safety as a hard probabilistic constraint and driver interference as a soft constraint. Furthermore, we consider unknown road surface characteristics and obstacle behaviors in our uncertainty model. By incorporating this uncertainty into decision-making, fewer collisions result on wet pavement, but the vehicle behaves more conservatively on dry pavement.

The path-time decomposition was introduced by Kant and Zucker (1986) who examined the problem of dynamic obstacle avoidance along a given path under velocity constraints[6]. Liu and Arimoto (1992) introduced an algorithm for the more general problem of shortest path planning among polygonal and curved obstacles[8]. We extend this approach to include path velocity and acceleration bounds in order to compute reachable sets in unprotected lefthand turns. A straightforward method for integrating uncertainty in obstacle velocity and behavior is considered as well.

III. Safety-Constrained Minimal Interference Principle

The vehicle's policy π is given the user's desired control u_t^d and sensor input z_t. Using z_t it infers a distribution $P(x_t)$ over hypothetical car-obstacle system states. Although our model can be generalized to two-dimensional motion with steering and velocity control, here we will only consider the longitudinal control problem in which the vehicle travels along a one dimensional space of a known single-lane road, which may be curved or straight. The output of the system is a continuous control $u \in [-1, 1]$, where $u = 0$ indicates no control, $u = -1$ indicates maximum braking, and $u = 1$ indicates maximum acceleration.

We define the *safety-constrained minimum interference* control $u_t^\star$ at probability α as the result of the following optimization:

$$\begin{aligned} u_t^\star &= \arg \min_{u \in [-1,1]} |u - u_t^d| \\ \text{s.t.} \quad & P(\text{safe}|u_t = u) \geq \alpha \end{aligned} \tag{1}$$

where we define $P(\text{safe}|u_t = u)$ as the probability that the system remains safe given the choice of u at the current time step and the safest sequence of controls thereafter. If no such u meets the α threshold, we set

$$u_t^\star = \arg\max P(\text{safe}|u_t = u). \tag{2}$$

This framework has several advantages in that the user's control will be replicated exactly if it is sufficiently safe, and safety and driver interference can be tuned using a single parameter α. The major challenge in this framework is evaluating the $P(\text{safe}|u_t = u)$ because it essentially requires solving a stochastic optimal control problem with nonlinear noise terms and constraints. To address this we make the approximation that the probability can be approximated by integrating over the optimal hypotheses evaluated under $P(x_t)$ *assuming that the underlying state hypothesis is true*. In other words,

$$P(\text{safe}|u_t = u) \approx \int_x S(x, u) P(x_t = x) dx \tag{3}$$

where $S(x, u)$ evaluates whether the system can remain safe under known state x and initial control u. Below we will present two concrete implementations whereby $S(x, u)$ can be analytically evaluated, which makes the evaluation of (3) tractable.

IV. Collision Imminent Braking

A. Assumptions and System Structure

First we assume the vehicle is moving in the same direction as the obstacle and the obstacle does not move in reverse. We assume that the vehicle is equipped with a speedometer and a range finder (e.g., radar or lidar). The behavior of an obstacle is considered as a black box, and the vehicle needs to infer whether an obstacle is still, accelerating, or braking through the information received through its sensors. In order to synthesize this information, we suppose the car runs an Extended Kalman Filter (EKF) to track a probability distribution over the system state [4]. It then incorporates this distribution into a probabilistic decision rule.

B. Stochastic Dynamics Model

The state of the car can be described using car's position p_c, velocity v_c and the maximum deceleration $a_{c_{max}}$ that the car currently can apply. Additionally, when an obstacle is present, the obstacle position p_o, velocity v_o, and acceleration a_o are modeled as part of the system state. The vehicle receives a noisy speedometer v_s and range reading d. Decisions

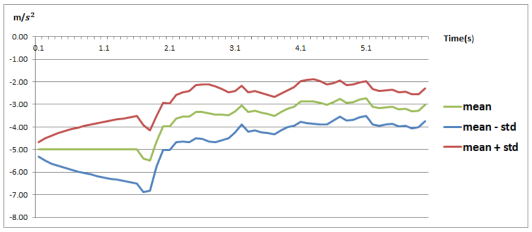

Fig. 1: The vehicle starts with an estimate of -5 m/s^2 maximum deceleration. As it starts braking on a wet pavement at $t = 1.5\,s$, the Kalman filter adjusts the estimate toward the correct value of -3 m/s^2.

TABLE I: Dynamics and Observation Models

Road Surface $\dot{a}_c$	Maximum applicable acceleration is a random walk $\dot{a}_c \sim \mathcal{N}(0, \Delta t)$
Actuation Errors $\dot{v}_c$	Proportional to control and maximum deceleration, $\dot{v}_c = ua_c(1 + \epsilon_u)$, with $\epsilon_u \sim \mathcal{N}(0, 0.01^2)$
Object Behavior $\dot{a}_o$	Random noise with 99.99% within $[-5.0\,m/s^2, 5.0\,m/s^2]$, $\dot{a}_o \sim \mathcal{N}(0, 1.25^2)$
Speedometer v_s	Multiplicative noise on actual velocity, $v_s = v_c(1 + \epsilon_s), \epsilon_s \sim \mathcal{N}(0, 0.025^2)$ (99.99% within 10% of current velocity)
Range Reading d	Combined linear and multiplicative noise $d = n_{dL} + (p_o - p_c)(1 + n_{dM})$, with $n_{dL}, n_{dM} \sim \mathcal{N}(0, 0.0125^2)$ (99.99% within by 5 cm + 5% of true distance)

are made at a time step of Δt (0.1 s in our implementation). The stochastic dynamics and sensor noise models are listed in Table I.

C. Extended Kalman Filter

The vehicle is assumed to employ an extended Kalman filter (EKF) in order to estimate the state from the stochastic dynamics and observations. An EKF is a version of the Kalman filter that addresses nonlinear systems by linearizing about the estimated mean and covariance [13]. The dynamics at time step t can be written in the following form:

$$x_{t+1} = f(x_t, u_t) + w_t \quad (4)$$

$$z_t = h(x_t) + v_t \quad (5)$$

$$w_t \sim \mathcal{N}(0, Q_t) \quad (6)$$

$$v_t \sim \mathcal{N}(0, R_t) \quad (7)$$

Here, x_t denotes the state $(p_c, v_c, a_{c_{max}}, p_o, v_o, a_o)$, u_t denotes the braking control input, z_t is the observation (d, v_s) at time step k. w_t is the process error term with Q_t as its covariance matrix. v_t is the measurement noise term with R_t as its covariance matrix.

At each step, the EKF maintains a state estimate $\hat{x}_t$ and covariance matrix P_t. Upon reading the observation z_t from the vehicle's sensors, the EKF performs a Kalman update using the system linearized about $\hat{x}_t$ to obtain a new state estimate $\hat{x}_{t+1}$ and covariance P_{t+1}. (Figure 1)

Because obstacles may appear and disappear from the range sensor reading, the obstacle state and distance measurements are included in the EKF update only when an obstacle is detected. When an obstacle appears for the first time, its position estimate is initialized to the raw range sensor estimate $\mathcal{N}(d, (0.0125d)^2)$. Its velocity is initialized to a broad distribution $\mathcal{N}(\hat{v}_c/2, (\hat{v}_c/2)^2)$, and its acceleration is initialized to $\mathcal{N}(0, (2.5\,m/s^2)^2)$.

Although the EKF is known to suffer from problems in highly nonlinear systems, in our case the system is close to linear and the EKF seems to provide sufficiently accurate performance. Nevertheless our decision-making algorithms still apply to more general state estimators, like particle filters.

V. Braking Policies And Evaluation

A braking policy $\pi(\hat{x}, P)$ produces a braking output u given the state estimate $\hat{x}$ from the Kalman filter and its covariance matrix P. Given perfect state information, the optimal policy is essentially trivial (known-state policy). But in the presence of uncertainty, optimality is not easy to define. We design a probabilistic approach to deal with uncertainty and produce human-like braking behavior.

A. Known-State Policy

The basic optimal braking policy $\pi_D(x)$ in the presence of complete state certainty is a bang-bang control given as follows.

Algorithm 1 Bang-Bang Policy

$p_c{}' \leftarrow p_c + v_c{}^2/(-2a_{c_{max}}) + C$
$t' \leftarrow v_c/(-a_{c_{max}})$
if $v_o + a_o t' \geq 0$ **then**
  $p_o{}' \leftarrow p_o + v_o t' + 1/2 a_o t'^2$
else
  $p_o{}' \leftarrow p_o + v_o{}^2/(-2a_o)$
end if
if $p_c' > p_o'$ **then**
  return $u = -1$
else
  return $u = 0$
end if

Here, C is a constant that is used for indicating some safety margin, which we set to 1 m. p_c' is the estimated stopping position of the car if it initiates maximum braking. p_o' defines the estimate position of the obstacle when the car stops, and it will either stop after the car does (first conditional branch) or before (second branch). If $p_c' > p_o'$, there will be a collision between car and obstacle, otherwise, no collision.

Using this policy we can implement (1) in a straightforward manner. First note that stronger braking is always guaranteed to be safer, which helps with the minimization. Then the process boils down to finding the weakest braking control that keeps the system safe with probability α. To approximate the integral in (3), we sample N candidate states $x^{(i)} \sim \mathcal{N}(\hat{x}, \hat{P})$, $(i \in [1, N])$ and compute the weakest (highest value) control $u^{(i)} = \pi_D(x^{(i)})$. We let $\pi(\hat{x}, \hat{P})$ take on the value of the control output at the $100(1-\alpha)$ percentile.

B. Smoothing in Post-Processing

The control policies discussed above are Markovian in that they produce the control output merely based on the state and covariance estimate at the current time step. Jerkiness might occur if control outputs are quite different in consecutive time steps. Therefore, We implement a discounting method that produces a weighted sum of $(1-D)$ of the smoothed control in its previous time step and D of the current π output. The constant D is known as the *discount factor*, and when D is low the output is highly smoothed, and when $D = 1$ the output is identical to the unsmoothed signal. Through experiments we found that the setting $D = 0.5$ strikes a good balance control smoothness and responsiveness.

C. Policy Evaluation

A good braking policy should be able to achieve high safety and low unnecessary control interference. In order to evaluate the policy we define the a risk index (RI) and interference index (II) which are functions of the following metrics:

1) *Collision velocity* (CV): the relative velocity if the car hits the obstacle, 0.0 otherwise.
2) *Discontinuity Time* (DT): measures the amount of jerk experienced by passengers. We integrate the time over which the acceleration at two time subsequent time steps is greater than some threshold, which we set to $4\,\text{m/s}^2$.
3) *Excess Time* (ET): measures the amount of time consumed by excessive braking in a scenario. A scenario is considered completed if the car stops, collides with an obstacle, or passes some marker. ET is computed by measuring the policy completion time and subtracting the completion time for an optimal collision-free policy with perfect state information.
4) *Stopping Distance* (SD): measures the distance to the obstacle after the vehicle stops, or 0.0 if the scenario is completed in any other manner.

RI and II are computed as follows:

$$RI = (CV_{avg}/CV_{safe})^2$$
$$II = c_1 DT + c_2 ET + c_3 SD$$

CV_{safe} is a constant that denotes a relatively safe velocity at which a collision is unlikely to lead to serious injury (set to 5 m/s in our implementation). RI is made proportional to the kinetic energy of the collision. Policies with $RI < 1$ are relatively safe. c_1, c_2, and c_3 are proportionality constants that are set to $2\,s^{-1}$, $1\,s^{-1}$, and $1/2\,m^{-1}$ respectively based on some amount of tuning. The goal of a braking policy is achieving low RI and II. It is important to distinguish between the interference objective used in (1), which is an instantaneous criterion used in the vehicle's internal decision mechanism, and II, which is a global measure of emergent system performance.

We designed five test scenarios $S_1, \ldots, S_5$ simulating actual environments that a collision imminent braking system would face in practice. These scenarios are illustrated in

Fig. 2: *Fixed Obstacle* scenario. The car moves toward a fixed obstacle 100 m ahead with initial velocity $20\,m/s$.

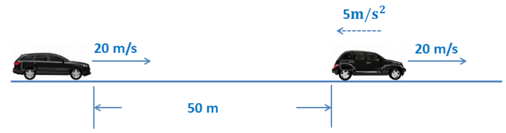

Fig. 3: *Emergency Braking Obstacle* Scenario. The car moves toward the other car at position 50 m ahead which is decelerating with $5m/s^2$. The initial velocity for both cars is $20\,m/s$.

Figures 2–6. We also investigated two variants of all five scenarios, in which the maximum deceleration $a_{c_{max}}$ is held constant at two different values.

1) *Dry Pavement* (DP): $a_{c_{max}} = -5m/s^2$.
2) *Wet Pavement* (WP): $a_{c_{max}} = -3m/s^2$.

Note that the behavior of the obstacle and simulation constants are *not* known in advance by the car, and it must rather infer them through sensor readings.

We tested the α-thresholding policies with $\alpha \in [10, 100]$ with 10 as interval and 95 to 99 with 1 as interval. Both smoothed and unsmoothed variants are tested. Each policy is tested 10 times in each of the 10 scenarios in the set $\mathcal{S} = \{S_1, S_2, S_3, S_4, S_5\} \times \{DP, WP\}$ described above using a Monte-Carlo simulation. Figure 7 depicts the results along with the ideal control at the origin. The smoothed 100-threshold is closest to ideal.

VI. Intersection Crossing

Intersection crossing requires consideration of both braking as well as acceleration in order to avoid crossing too slowly. It also requires considering the behavior of multiple obstacles which makes optimal decision boundaries more complex even in the known-state case. Here we present an analytical conservative computation of the safety of a given state $S(x)$ in the presence of multiple-lane intersection crossings. We are currently performing experiments applying this technique under uncertainty using the construction of (1).

A. Constraints in the Path-Time Space

The state is defined as the time t, arc-length parameterized position p, and tangential velocity v along a known path, which can be straight as in street crossings or curved as in unprotected lefthand turns. Acceleration and deceleration is assumed bounded. The car begins at state (p_t, v_t) and ends a maneuver at final position p_T with a range of admissible final velocities $[\underline{v_T}, \overline{v_T}]$.

The planning problem is to find a trajectory between (p_t, v_t) and the goal region that respects acceleration and deceleration bounds and also avoids obstacles. The obstacle avoidance constraint for obstacle O_i, $i = 1, \ldots, n$ can be

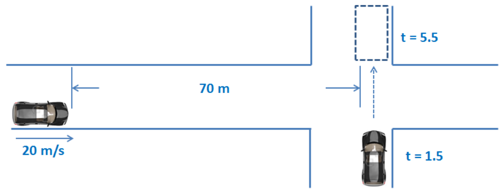

Fig. 4: *Transient Object* scenario. At first the car is at position 0 and moving with no obstacles detected. After 1.5 seconds, the car will detect an obstacle ahead at position 70. The obstacle will appear in the radar for 4 seconds and then disappear.

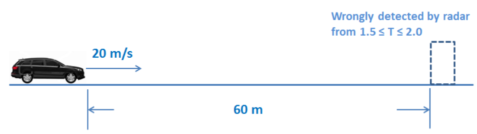

Fig. 5: *False Positive* scenario. The car starts at $20m/s$ with no obstacles detected. After 1.5 seconds, the car will wrongly detect an obstacle at position 60. This false positive period lasts for 0.5 seconds.

geometrically constructed as a forbidden region PO in the path-time space (p, t) by examining the set of (p, t) points that would cause the vehicle to overlap O_i [3]. This requires that the length and width of O_i are known and that future behavior of O_i is known. We currently assume that each O_i travels along a given path with confidence intervals of velocity and acceleration, which is assumed constant.

For each obstacle O we construct conservative rectangular forbidden regions $[\underline{p_O}, \overline{p_O}] \times [\underline{t_O}, \overline{t_O}]$ in the (p, t) plane. The forbidden interval $[\underline{t_O}, \overline{t_O}]$ is constructed by examining the minimum and maximum extents of the vehicle along O's path, and examining the time that O occupies these extents plus a minimum time-to-collision margin [11]. The interval $[\underline{p_O}, \overline{p_O}]$ is based upon the vehicle dimensions plus certain comfort margins on either side.

B. Analytical Planning with Piecewise Constant Accelerations

Given this construction of forbidden regions, we extend the path-time planning approach of [6] to consider acceleration constraints. This requires planning in the path-velocity-time space, and it is useful to note that any optimal trajectory will either connect directly to the goal state, or pass tangentially along either the upper-left or lower-right corner of one or more obstacles [8]. Based on this observation we consider searching among the set of piecewise constant acceleration controls with discontinuities at lower-right and upper-left obstacle corners.

The search explores a graph G where each node is associated with set in the (p, v, t) space. Typically these sets are specified as an interval of reachable velocities R at a point in the (p, t) plane, except the initial node is specified as a point (p_t, v_t, t), while the goal region has specified position, unspecified time, and a range of admissible velocities. Each

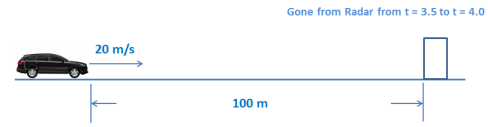

Fig. 6: *False Negative* scenario. The car starts at $20m/s$, and a fixed obstacle is 100 m away. After 3.5 s, the car loses track of the obstacle for 0.5 s seconds and then finds the obstacle again.

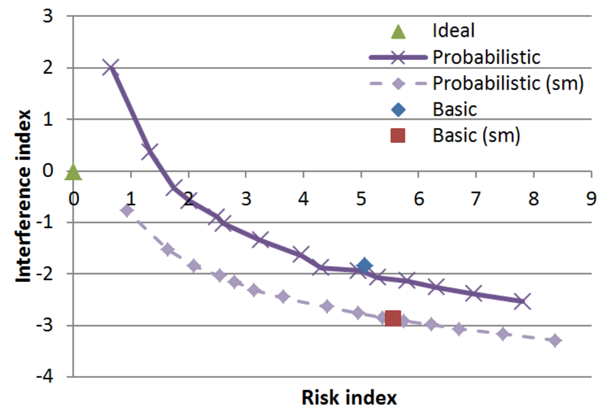

Fig. 7: Performance of smoothed and unsmoothed variant of α-thresholding policies with $\alpha \in [10, 100]$ with 10 as interval. Risk index and interference index exhibit an inverse relationship as α increases. The triangle dot (origin) represent the ideal control.

expansion step checks whether the node can be connected with a dynamically feasible, obstacle free trajectory to a vertex (p', t') of an obstacle region (either the upper left or lower right vertex) or to the goal region. If so, each connected component of the set of reachable velocities is instantiated as a new node.

The planner makes heavy use of the NextReachableSet subroutine to propagate the set of reachable velocities, excluding obstacle avoidance constraints, from one node to another. A variant is used to connect nodes to the goal region.

Algorithm 2 Compute the dynamically reachable set of velocities R_{i+1} after a change in position $\Delta p, \Delta t$ from an initial point with velocities in interval R_i.

NextReachableSet(Δp, Δt, R_i)

$A = [\frac{\Delta p - (1/2) a_{max} \Delta t^2}{\Delta t}, \frac{\Delta p - (1/2) a_{min} \Delta t^2}{\Delta t}] \cap R_i$

$B = [2\frac{\Delta p}{\Delta t} - upper(A), \; 2\frac{\Delta p}{\Delta t} - lower(A)]$

return $R_{i+1} = A \cap [0, \; inf)$

The recursive depth first search is given in Algorithm 3. The algorithm first checks for a connection to the goal region. If successful, the range of reachable velocities at the goal is returned. The recursion can then be terminated, or continued to collect all trajectories that reach the goal. Next, it examines all subsequent trajectories from the node through obstacle vertices.

Of the dynamically feasible trajectories computed by Nex-

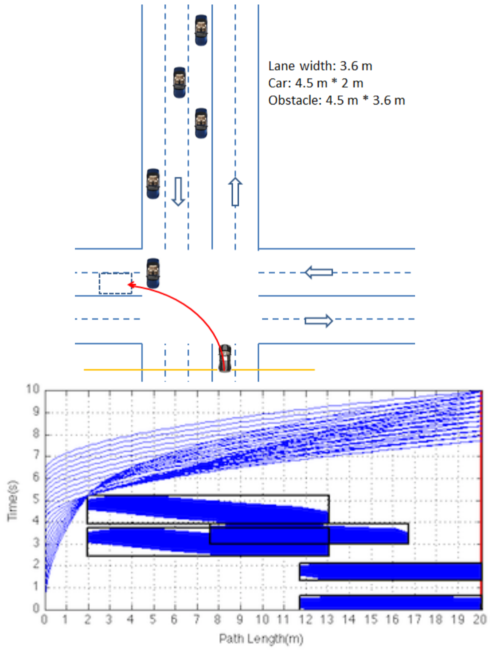

Fig. 8: **Above:** A lefthand turn scenario. The red line is the car's path with reference point at the middle of its rear axis. The orange line is considered a reference line. The distances of obstacles to the line are 13 m, 28 m, 44 m, 37 m, 52 m. All obstacles are moving at 10 m/s. **Below:** Feasible trajectories that arrive within 10 s. Rectangles are path-time obstacles.

tReachableSet, it calls the CollisionCheckRegion subroutine to partition the range of reachable velocities into a set of disjoint intervals **R**. (It is possible for small (p,t) obstacles to partition the trajectories into multiple components). This test is performed by examining where the (p,t) trajectories overlap each obstacle as the acceleration is swept between a_{min} and a_{max}.

Algorithm 3 Perform a recursive depth first search for a feasible path from any state (p,t,v) where $v \in R_i$.

ExpandNode(p, t, R_i)
  If $\exists v \in R_i$ s.t. (p,t,v) can be connected to the goal, output the path leading to the goal.
  for all (p', t') corners of $O_1, \ldots, O_n$ **do**
    $\hat{R} \leftarrow NextReachableSet(p'-p, t'-t, R_i)$
    $\mathbf{R} \leftarrow CollisionCheckRegion(p, t', R_i, \hat{R})$
    for all $R \in \mathbf{R}$ **do**
      $V \leftarrow V \cup N$, with $N = (p', t', R)$
      $ExpandNode(p', t', R)$
    end for
  end for

Since $p' > p$ and $t' > t$, the recursion necessarily terminates in at most n steps. Further, it can be shown that the graph contains no more than $2n+2$ vertices, and hence a naive implementation of the search runs in $O(n^3)$ time and $O(n^2)$ space. The assumption that discontinuities in control only occur at corner points simplifies the problem, but may lead the algorithm to failure in a cluttered space that requires carefully chosen acceleration and braking. We believe that these situations rarely arise in practice, and plan to address them in future work. We are currently working to integrate our probabilistic decision thresholds with this algorithm.

VII. Conclusion

We present a generic framework for conducting semi-autonomous collision avoidance and two concrete implementations that can be used under that framework. First, a probabilistic based collision-avoidance braking strategy in terms of their behavior in the presence of uncertainty in vehicle dynamics, sensor noise, and unpredictable obstacle behavior. A number of Monte-Carlo simulations demonstrate that this probabilistic braking strategy can achieve good peformance in different testing scenarios. Second, a technique for determining safe trajectories for unprotected lefthand turns, which was shown to be time-optimal under the given constraints.

Acknowledgement

This work is partially supported by the Indiana University Collaborative Research Grant fund of the Office of the Vice President for Research.

References

[1] M. Althoff, O. Stursberg, and M. Buss. Model-based probabilistic collision detection in autonomous driving. In *Intelligent Transportation Systems*, 2009.

[2] S. .Anderson, S. Peters, K. Iagnemma, and T. Pilutti. A unified approach to semi-autonomous control of passenger vehicles in hazard avoidance scenarios. In *IEEE International Conference on Systems, Man and Cybernetics (SMC)*, 2009.

[3] T. Fraichard. Dynamic trajectory planning with dynamic constraints: a 'state-time space' approach, 1993.

[4] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson, and P.-J. Nordlund. Particle filters for positioning, navigation, and tracking. In *Transactions on Signal Processing*, 2002.

[5] J. Hillenbrand, A. Spieker, and K. Kroschel. Efficient decision making for a multi-level collision mitigation system. In *Intelligent Vehicles Symposium*, 2006.

[6] K. Kant and S. W. Zucker. Toward efficient trajectory planning: the path-velocity decomposition. *Int. J. Rob. Res.*, 5:72–89, September 1986.

[7] R. Karlsson, J. Jannson, and F. Gustafsson. Model-based statistical tracking and decision making for collision avoidance application. In *American Control Conference*, 2004.

[8] Y.-H. Liu and S. Arimoto. Path planning using a tangent graph for mobile robots among polygonal and curved obstacles. *Int. J. Rob. Res.*, 11:376–382, August 1992.

[9] National Highway and Traffic Safety Administration. Fatality analysis reporting system general estimates system: 2008 data summary. national highway and traffic safety administration report no.: Dot hs 811 171.

[10] New York Times. Google cars drive themselves, in traffic, October 9 2010.

[11] B. Sultan and M. McDonald. Assessing the safety benefit of automatic collision avoidance systems. In *International Technical Conference on the Enhanced Safety of Vehicles (ESV)*, 2003.

[12] S. Thrun, M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, G. Hoffmann, K. Lau, C. Oakley, M. Palatucci, V. Pratt, P. Stang, S. S. C. Dupont, L.-E. Jendrossek, C. Koelen, C. Markey, C. Rummel, J. van Niekerk, E. Jensen, P. Alessandrini, G. Bradski, B. Davies, S. Ettinger, A. Kaehler, A. Nefian, and P. Mahoney. Stanley: The robot that won the darpa grand challenge. *J. Robot. Syst.*, 23(9):661–692, 2006.

[13] G. Welch and G. Bishop. An introduction to the kalman filter. Technical report, University of North Carolina at Chapel Hill, 2006.