Real time implementation of socially acceptable collision avoidance of a low speed autonomous shuttle using the elastic band method

Real time implementation of socially acceptable collision avoidance of a low speed autonomous shuttle using the elastic band method

Haoan Wang[a,c], Antonio Tota[d], Bilin Aksun-Guvenc[a,b,*], Levent Guvenc[a,b,c]

[a]*Automated Driving Lab, Center for Automotive Research, the Ohio State University*
[b]*Department of Mechanical and Aerospace Engineering, the Ohio State University*
[c]*Department of Electrical and Computer Engineering, the Ohio State University*
[d]*Department of Mechanical and Aerospace Engineering, Politecnico di Torino*
[*]*Corresponding author, e-mail: aksunguvenc.1@osu.edu*

ABSTRACT

This paper presents the real time implementation of socially acceptable collision avoidance using the elastic band method for low speed autonomous shuttles operating in high pedestrian density environments. The modeling and validation of the research autonomous vehicle used in the experimental implementation is presented first, followed by the details of the Hardware-In-the-Loop connected and autonomous vehicle simulator used. The socially acceptable collision avoidance algorithm is formulated using the elastic band method as an online, local path modification algorithm. Parameter space based robust feedback plus feedforward steering controller design is used. Model-in-the-loop, Hardware-In-the-Loop and road testing in a proving ground are used to demonstrate the effectiveness of the real time implementation of the elastic band based socially acceptable collision avoidance method of this paper.

Keywords: Socially acceptable collision avoidance, Collision avoidance, Autonomous shuttle, Elastic band method, Hardware-In-the-Loop simulation, Road testing

1. Introduction

Autonomous driving has been divided into six categories with Level 0 being a non-automated and Level 5 being a fully autonomous vehicle according to the Society of Automotive Engineers [1]. Currently available automated driving technology falls under Level 2 and Level 3 which are partial and conditional automation, respectively. Level 2 partial automation is available in series production vehicles with lane centering control for steering automation and adaptive cruise control and collision avoidance for automation in the longitudinal direction. Partial automation is characterized by all driving actuators being automated and the presence of a driver who can intervene when necessary. Recently introduced autopilot systems for cars are examples of conditional automation where the car takes care of driving in some driving modes (like highway driving) but the human operator is always in the driver seat to take over control if necessary. Level 3 autonomous highway driving systems in which almost all highway driving functions are carried out autonomously with the driver needing to take over only if something goes wrong are expected to reach series production by 2020. A Level

4 autonomous highway driving extension in which the driver is still in the driver seat while the vehicle can perform highway driving completely autonomously, without the need for driver interaction, is expected to enter the market around 2025. In future Level 5 autonomous driving, there is no need for a driver as the vehicle takes care of all driving tasks autonomously.

Autonomous shuttles in smart cities used for solving the first-mile and last-mile problem form another well-known, emerging application of autonomous road vehicles that are currently at Level 2 or Level 3. These shuttles operate at relatively lower speeds which definitely improves safety levels. These shuttles operate in significantly less structured environments with unpredictable interaction with vulnerable road users like pedestrians and bicyclists. The roads they follow involve pedestrian crosswalks, intersections with or without traffic lights, roundabouts and sharper turns as lower speed of operation is possible. Successful Level 4 like autonomous driving of these low speed shuttles is possible in fixed routes within blocked traffic environments as all the uncertainties that the vehicle can face are taken out of the picture by using a lane dedicated only to these shuttles (no other traffic) and by using a fixed route. However, a true Level 4 autonomous driving capability of these autonomous shuttles requires autonomous decision making. The most basic and critical decision making is autonomous collision free path planning and collision avoidance maneuvering of these shuttles in a smart city setting where the autonomous shuttles also have to work in areas that are highly populated by groups of pedestrians. A university campus, an outdoor shopping area, downtown areas closed to mainstream traffic are typical examples where low speed autonomous shuttles have to interact with groups of pedestrians and autonomously plan their collision free paths and avoid collisions with them. This is the main focus of this paper.

Collision free path planning and collision avoidance require situational awareness using the autonomous vehicle perception sensors as was done in the work of Aufrere et al [2] where a probabilistic collision prediction and warning system was also presented. As compared to reference [2], we concentrate on the collision free path planning and collision avoidance maneuvering rather than perception and situational awareness in the current paper. The collision prediction and warning system in [2] is based on checking all trajectories out of a set of possible ones for collisions. As compared to this brute force approach of reference [2], we use the fact that the road to be followed defines the initial trajectory which should be modified around the obstacle in a natural and simple manner. The elastic band method of collision avoidance used in this paper is not based on checking all possible trajectories and, thus, naturally works faster in real time. Ferrara and Vecchio [3] have formulated collision cones in their work on collision avoidance of vehicle platoons. They have also considered platoon and pedestrian collision risk and have used sliding mode control as their collision avoidance control method. As compared to the simulation only approach of reference [3], this paper concentrates on real experiments in a proving ground and also in a Hardware-In-the-Loop simulator where real time implementation issues are also considered. The sliding mode control of [3] and the parameter space based robust collision free steering of this paper are both robust controllers in the presence of model uncertainty and disturbances.

The parameter space based controller of this paper is much easier to implement in real time and results in a characterization of all controller gain combinations with a graphical display of the results in comparison to obtaining only one controller with possible chatter problems in the sliding mode control method of [3]. A collision free path planning and following framework is presented by Khajepour *et al* [4] for the collision avoidance of autonomous vehicles. The desired tracking path was generated by a three dimensional virtual potential field based on road and obstacle information in [4]. The elastic band method used in this paper is similar to that in [4] but does not suffer from the singularities of the virtual potential field approach. As the road to be followed results in the trajectory to be followed and forms the original elastic band before deformation about pedestrian(s), the computation is much simpler than trying to find the collision free path within an artificial potential field as in reference [4].

For the path tracking steering controller design, a Multi-constrained Model Predictive Control (MMPC) optimal problem was formulated and used to prevent collisions with both static and dynamic obstacles [4]. The parameter space based robust steering controller is much easier to design and implement in real time as compared to the MMPC approach of [4]. Fu *et al* [5] presented a novel obstacle avoidance algorithm called the navigation circle which is a method for real-time path planning. A collision-free path generated by the navigation circle was optimized through the kinematic model of the autonomous vehicle to obtain a kinematically feasible trajectory in [5]. A real-time path-planning algorithm was proposed by Chu *et al* [6] for off-road autonomous driving in the presence of static obstacles. A set of predefined waypoints was used to generate path candidates and each candidate was evaluated using obstacle data. Safety, smoothness and consistency costs were considered during the selection of an optimal path to evaluate the effects of environment uncertainty and vehicle dynamics. The elastic band method of this paper is a much more efficient method computationally and results in a smooth trajectory without having to search over a set of possible trajectories as in references [5-6]. The method presented here also uses a social distance for collision free path planning and collision avoidance maneuvering about pedestrian(s). It is also possible to use a conservative pedestrian safety zone in the computations to treat moving pedestrian(s).

Based on the comparisons above, this paper uses the elastic band method for collision avoidance as it is both a relatively easy and natural way of implementing collision free path planning for vehicles following a road and as it can also be operated in real time. The elastic band method was first proposed by Quinlan and Khatib [7] for collision free path planning and collision avoidance for mobile robots. The elastic band method was applied to road vehicle collision avoidance by Ararat and Aksun-Guvenc [8]. They presented realistic simulation results with several road vehicles for higher speed highway driving [8]. Driving in city roads involves a mixed traffic environment where there are also pedestrians, i.e. vulnerable road users. In contrast to reference [8], this paper concentrates on low speed autonomous shuttles that operate in large walkways shared with pedestrians. This is a very common situation in university campuses, outdoor shopping areas and downtown areas closed to mainstream traffic. As the autonomous shuttles and pedestrians share the same walkway or road in those

cases, a collision avoidance method that also respects the socially acceptable distance around groups of pedestrians is needed. A modified elastic band based collision avoidance method was, therefore, applied in Emirler *et al* [9] to avoid collision risk with stationary pedestrian groups while keeping a socially acceptable distance.

This paper is an extension of the earlier work of some of the authors in [8] and [9] and concentrates on real time implementation of the method using an actual vehicle and also considers the case of moving pedestrians. As compared to reference [8], the socially acceptable collision avoidance region was added to the calculations here, similar to the more recent reference [9]. In comparison to reference [9], the method and algorithm had to be modified to be able to work directly with a trajectory of GPS waypoints that were broken down into segments that were fit by cubics. Both of these previous papers [8] and [9] were based on simulation studies. The current paper concentrates on real time implementation of the method. Even though the same elastic band method was used, the algorithm had to be changed for real time implementability. The changes were re-coding of the algorithm to calculate the deformed path only locally around the pedestrian(s) in real time after detection, using analytical expressions for derivatives as compared to numerical differentiation, smoothing the shape of the deformed trajectory to have a more feasible path and equating second derivatives of cubic polynomial fits also (as opposed to polynomial continuity and first derivative) at the intersection of the segments for a smoother transition. In comparison to reference [9], the possibility of moving pedestrian(s) was also considered in a conservative manner by adjusting the corresponding distance $d_{pedestrian}$ to accommodate for this. In this paper, a feedforward plus feedback architecture is used as the steering controller as opposed to use of feedback control alone in [9]. The feedforward controller acts like a human driver and the feedback controller is designed using parameter space robust control methods [10-12].

The concept of *social acceptance* has been widely studied in the robotics area. Chan *et al* [13] have defined socially acceptable robotics for object handovers, where a framework was proposed to enable robots to learn proper grasp configurations for handovers through observations. Socially acceptable robotic navigation was introduced and used by Shiomi *et al* [14] and Vasconcelos *et al* [15]. In crowded areas like shopping malls or other high pedestrian density places, the social distance between pedestrians and robots provide people with comfort and safety. In this paper, the same idea of social acceptance is applied to a low speed autonomous shuttle operating in a smart city for automated collision avoidance in a crowded urban area. Such low speed autonomous shuttles are also used to solve the first mile (access to transportation choice) and last mile (from transportation station to final destination) problems and to help the elderly and people with mobility impairment. As compared to references [13-15], the social distance is incorporated directly into the algorithm which runs in real time and works with low speed shuttles as opposed to mobile robots.

The contributions of this paper are as follows. This paper is on an experimental implementation of the socially acceptable collision avoidance system based on the elastic band method. Model-in-the-loop simulations in our previous work in Emirler *et al* [9] and Wang *et al* [16] are extended by Hardware-In-the-Loop testing and

experimental implementation and testing in a test track and the incorporation of moving pedestrians. The main objective and aim of the present paper is to show that the method can be implemented in real time and used in actual vehicles.

The rest of the paper is organized as follows. Section 2 presents the experimental vehicle used and its computing, sensing, communication and actuation architecture. Vehicle modeling, road modeling and some model validation results are presented in Section 3. Section 4 is on the hardware-in-the-loop simulation system and the test track used. The elastic band based collision avoidance algorithm is presented in Section 5. Section 6 formulates the robust PID plus feedforward steering control system design in parameter space. In Section 7, simulation and experimental results are presented to demonstrate the effectiveness of the proposed elastic band based socially acceptable collision avoidance system in a real implementation. The paper ends with conclusions in the last section.

2. Experimental vehicle

The experimental vehicle used is the passenger car autonomous research vehicle in the Automated Driving Lab in the Center for Automotive Research of the Ohio State University. This is a state-of-the-art Ford Fusion Hybrid vehicle automated using a Dataspeed throttle/brake/steer-by-wire interface module and is illustrated in Fig. 1. A dSpace microautobox electronic control unit is used as the low level controller for calculating and sending throttle, brake and steering commands to the vehicle actuators. The dSpace microautobox collects data from the in-vehicle sensors (steering wheel position sensor, acceleration/yaw rate sensor cluster, wheel speed sensors) using its CAN bus interface. A low footprint, rugged PC running real time Linux is used as a data collection, processing and distribution unit for environment perception and situational awareness sensors (Lidar, camera, radar) and for higher level decision making. The experimental vehicle is equipped with a Delphi ESR radar with simultaneous short and long range sensing, a Velodyne VLP-16 16 channel Lidar, a Mobileye camera development kit and an xNAV550 RTK GPS with IMU and INS integration from OXTS (with a real time positioning accuracy of 2 cm) and a Denso DSRC radio for V2X communication running the basic safety message (BSM) set. Four side looking radars are in the process of being integrated into this autonomous vehicle. The computing, sensing, communication and actuation architecture of this experimental autonomous vehicle is shown in Fig. 2.

The high accuracy differential GPS is used for localization and to follow a route of GPS waypoints which are either pre-recorded or are taken automatically from OpenStreetMap [16]. A segmentation and smoothing algorithm runs in the background to convert the GPS waypoints to a smooth route [17]. A cruise controller with car following capability (adaptive cruise control and cooperative adaptive cruise control) runs in the longitudinal direction while a coordinated steering controller is used for path tracking in the lateral direction. The Lidar, Mobileye camera and radar are environmental perception sensors that are used for situational awareness. The 16 channel Velodyne Lidar used here outputs raw point cloud data over an Ethernet connection. Processing of the point cloud data takes place in the computer in Fig. 2.

The Mobileye camera development system and the radar have built-in processors and software which determine and classify objects in their field of view. All three perception sensors can determine the relative location of pedestrians. The camera is the easiest to use under normal weather conditions. This paper concentrates on the collision avoidance algorithm once the obstacles (pedestrians here) are detected as all three perception sensors in the autonomous vehicle in Fig. 2 are able to detect and track pedestrians.

Fig.1. Ford Fusion Hybrid automated driving vehicle.

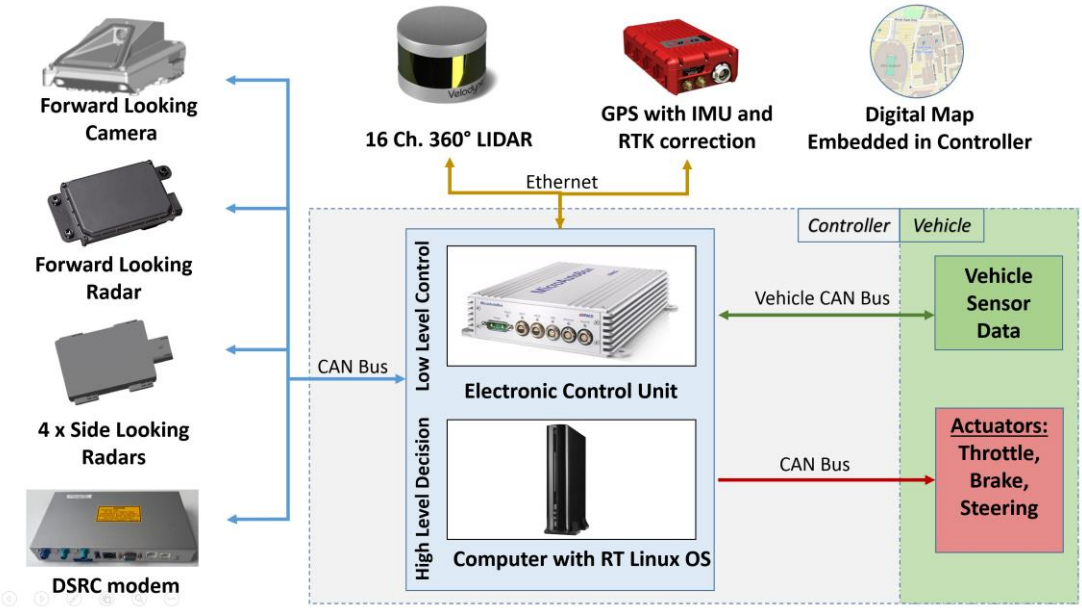

Fig. 2. Computing, sensing, communication and actuation architecture of experimental

autonomous vehicle used.

3. Vehicle modeling and validation

A path following model built on the linearized bicycle model and illustrated in Fig. 3 and a CarSim model are used as the two vehicle models in this paper. The linearized bicycle model based path following model is also used for steering controller design later in Section 6. Both models are used in simulations. The bicycle model of the vehicle and the path to be followed are shown in Fig. 3 along with the variables used in the model equations. The definitions of these variables are given in Table 1. The state space equations of this model are given by

$$\frac{d}{dt}\begin{bmatrix}\beta\\ r\\ \Delta\psi_p\\ e_y\end{bmatrix}=\begin{bmatrix}a_{11} & a_{12} & 0 & 0\\ a_{21} & a_{22} & 0 & 0\\ 0 & 1 & 0 & 0\\ V & l_s & V & 0\end{bmatrix}\begin{bmatrix}\beta\\ r\\ \Delta\psi_p\\ e_y\end{bmatrix}+\begin{bmatrix}b_1\\ b_2\\ 0\\ 0\end{bmatrix}\delta_f+\begin{bmatrix}0\\ 0\\ -V\\ 0\end{bmatrix}\rho_{ref}\,, \tag{1}$$

where

$$a_{11}=-\left(C_r+C_f\right)/mV\,,\ a_{12}=-1+\left(C_r l_r-C_f l_f\right)/mV^2$$
$$a_{21}=\left(C_r l_r-C_f l_f\right)/J\,,\ a_{22}=-\left(C_r l_r^2+C_f l_f^2\right)/JV\,,\ b_1=C_f/mV\,,\ b_2=C_f l_f/J\,.$$

The above single track vehicle based path following model is implemented in Simulink and will be called the Simulink model of the vehicle. It will also be called the single track model of the vehicle in the vehicle dynamics model validation part.

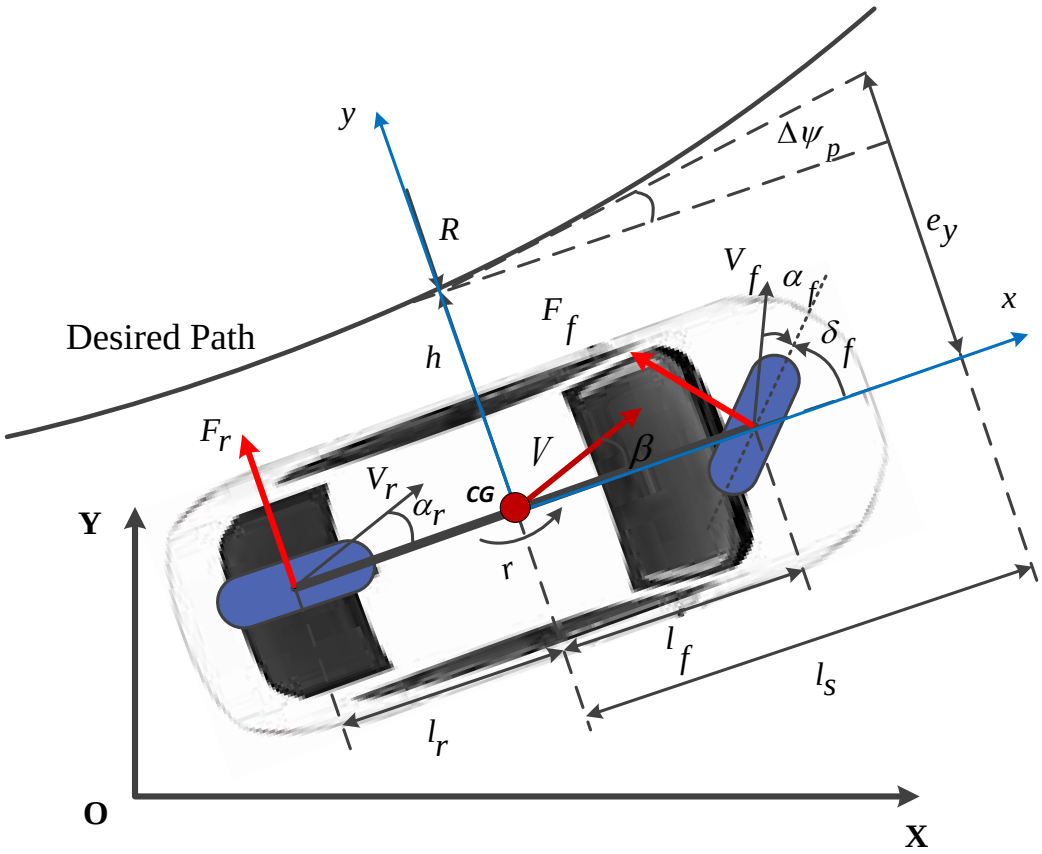

Fig. 3. Diagram illustrating the vehicle model.

Table 1
Parameters of the vehicle model

β	vehicle side slip angle [rad]	r	vehicle yaw rate [rad/s]
subscript f	front tires	subscript r	rear tires
V	vehicle velocity [m/s]	$\Delta\psi_p$	yaw orientation error with respect to path [rad]
l_s	preview distance [m]	e_y	lateral deviation [m]
δ_f	front wheel steering angle [rad]	δ_r	rear wheel steering angle [rad]
J	Yaw moment of inertia [3728 kgm^2]	C_f	front cornering stiffness [195,000 N/rad]
C_r	rear cornering stiffness [50,000 N/rad]	m	vehicle mass [1997.6 kg]
l_f	distance from CG to front axle [1.3008 m]	L_r	distance from CG to rear axle [1.5453 m]
λ	distance parameter along path	$\rho_{ref}=1/R$	curvature of path [1/m]

Path curvature ρ_{ref} enters the model (1) as a disturbance to be rejected by the steering controller. This formulation makes it easier to enter a feedforward disturbance rejection steering input, calculated based on the single track vehicle kinematics which will be

like a human driver. The lateral deviation from the desired path at the preview distance l_s in Figure 1 is e_y and is given by

$$e_y = h + l_s \sin(\Delta\psi_p), \tag{2}$$

where h is the lateral deviation from the desired path at the center of gravity of the vehicle and $\Delta\psi_p$ is the orientation error between the vehicle x axis and the desired path tangent line.

The path tracking steering controller aim is to determine and apply steering input δ_f that will result in zero tracking error e_y in the presence of road curvature ρ_{ref}. Since the reference path is approximated by a sequence of third order polynomial function fits of the GPS waypoints [17], the road curvature can be analytically calculated using

$$\rho_{ref}(\lambda) = \frac{\left| \frac{dx_p(\lambda)}{d\lambda} \frac{d^2 y_p(\lambda)}{d\lambda^2} - \frac{dy_p(\lambda)}{d\lambda} \frac{d^2 x_p(\lambda)}{d\lambda^2} \right|}{\left(\left(\frac{dx_p(\lambda)}{d\lambda} \right)^2 + \left(\frac{dy_p(\lambda)}{d\lambda} \right)^2 \right)^{\frac{3}{2}}}, \tag{3}$$

where $x_p(\lambda)$ and $y_p(\lambda)$ are the polynomial fit points on the reference path to be followed as a function of λ which is non-dimensional distance along the path [17]. As the path to be followed is a sequence of third order polynomial segments and the first and second derivatives are made to match at the intersections of segments, calculation of the curvature disturbance input does not pose any numerical discontinuity problems. The closed path to be followed by the vehicle if no obstacles are encountered and the corresponding curvature calculated using Equation (3) are shown in Fig. 4 and Fig. 5, respectively. The Vehicle Dynamics Area (VDA) used in the experimental implementation in the TRC proving ground and the layout of the closed path are also shown in Fig. 4.

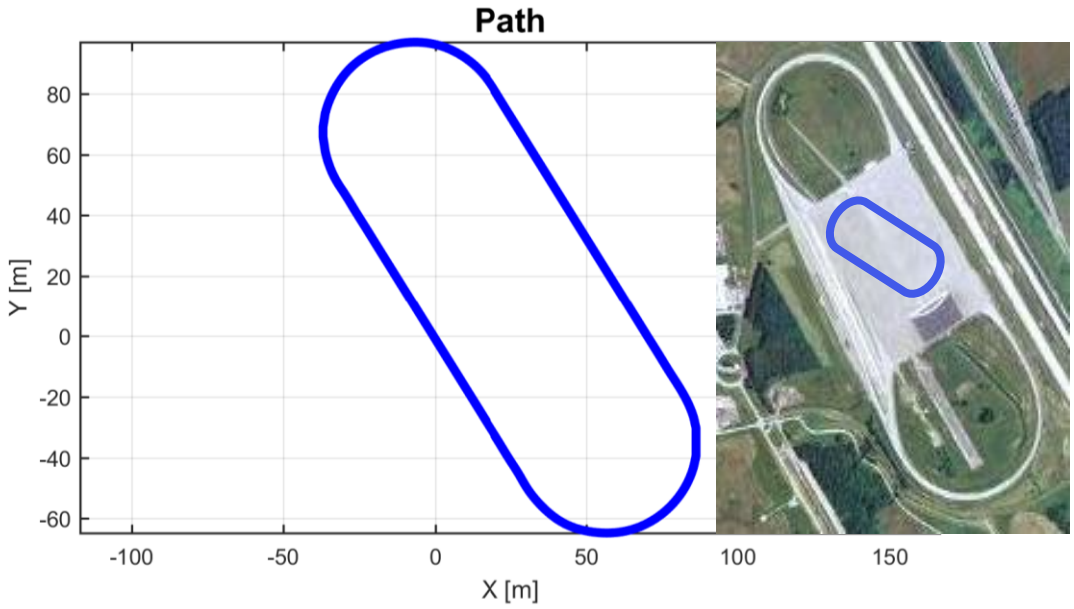

Fig. 4. Closed path used in the experimental evaluation.

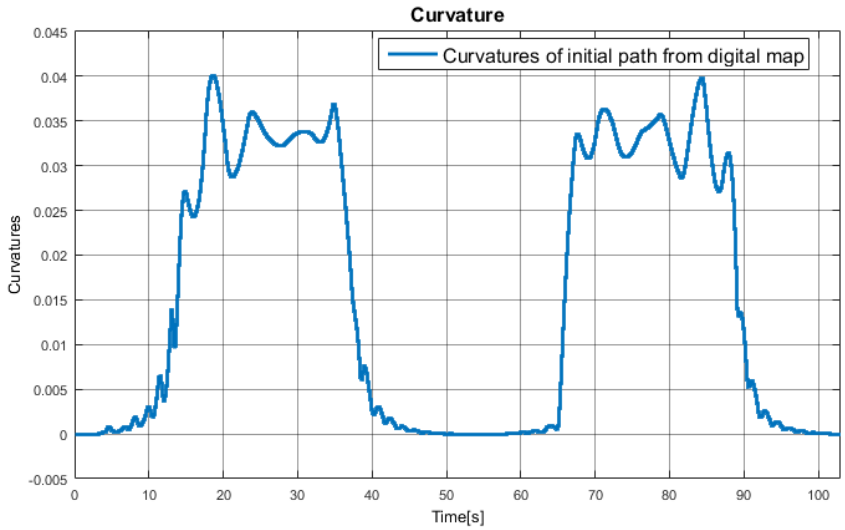

Fig. 5. Curvature of the closed path used in the experimental evaluation.

CarSim is a commercially available, high fidelity vehicle dynamics modeling program. A CarSim model of the Ford Fusion Hybrid autonomous vehicle used in this paper was developed in our lab and is used as the second vehicle model. A snapshop of a main CarSim graphical user interface page is shown in Fig. 6. The CarSim vehicle model is used as the vehicle in our Simulink co-simulations with all control and decision making actions being taken inside Simulink.

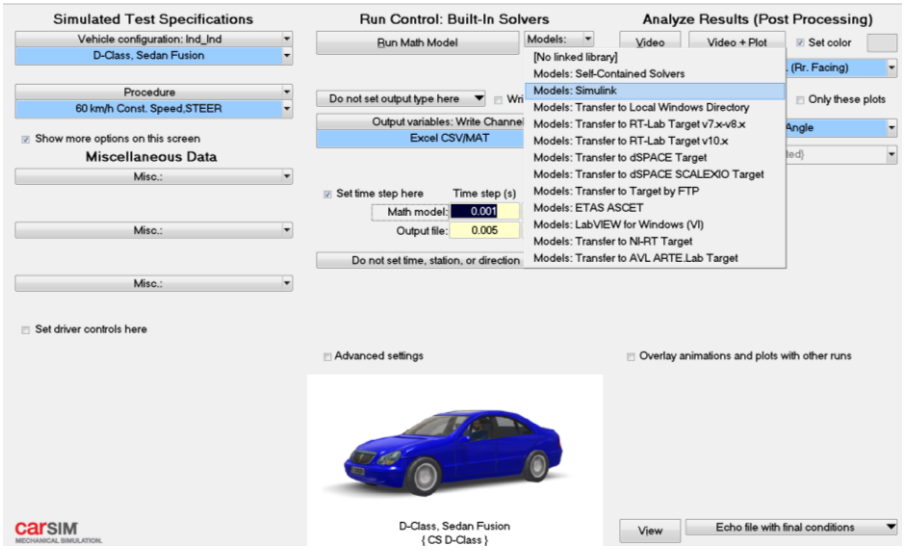

Fig. 6. CarSim user input screen.

The parameters of the autonomous vehicle in Fig. 1 that is used in the experiments here were determined for the linear path tracking model in Equation (1) and its higher fidelity CarSim model using tests on vehicle parameter testing machines and vehicle dynamics tests in the Transportation Research Center proving ground. Some of the geometric distance parameters were available in the literature. The ramp steer maneuver results will be presented in this section as part of the validation of the model. In the ramp steer maneuver, the vehicle speed is brought to a constant cruise speed (30 km/hr here) and the steering wheel angle is increased in a ramp with a slope of 14°/sec until 400° is reached and the vehicle is stopped when lateral acceleration saturates. Fig. 7 shows the steering wheel input used and the vehicle speed during the ramp steer test for the single track Simulink and CarSim models and the experimental results. Fig. 8 shows the lateral acceleration, yaw rate and side slip angle values obtained during the ramp steer test in comparison to the corresponding responses of the single track Simulink and CarSim vehicle models. It is seen that both the single track Simulink and the CarSim models of the vehicle used very closely resemble the dynamics of the vehicle up to about 5 m/s^2 of lateral acceleration.

Conti et al in [21] had stated the importance of different adhesion conditions on vehicle dynamics and propose a new approach for the reproduction on the roller-rig of a generic wheel-rail adhesion pattern and, in particular, of degraded adhesion conditions. The parameter space based controller design procedure introduced later in Section 6 for path tracking control can handle parametric uncertainty in the physical parameters of the model. The modeling approach used here is compared with that in references [18-21] for uncertainty modeling. It should be noted that references [18-19] are on yaw dynamics stabilization and use a bicycle model of the vehicle and not a path following model as in this paper. Concentrating only on the bicycle dynamics part of the path following model and on how uncertainty is modeled, the following observations were made. In [18-20], a state space yaw dynamic model of the vehicle is used and uncertainty is entered as perturbations in the state space model matrices. In the present

paper, the modeling is based on the transfer function approach and the model uncertainty is parametric. In [18-20], the front and rear tire cornering stiffnesses are assumed to be uncertain and that uncertainty is carried over to the perturbations in the state space matrices. In reference [19], the authors also take a look at uncertainty in nonlinear functions of vehicle speed $1/V$ and $1/V^2$ as other uncertainties meanwhile in [20] the vehicle system is parameterized for variation in speed and independent variation of front and rear effective lateral tire stiffness's for guaranteeing robust controller stability. In both references [18-19], vertices of rectangular polytopes of uncertain parameters are used in the analysis. The uncertainty boxes we use and the multi-model design procedure we typically use in D-stability controller design are similar to [18-19].

4. Hardware-in-the-Loop simulator

Hardware-In-the-Loop simulation is an effective way of testing vehicle dynamics controllers and advanced driver assistance systems [22-23]. A validated and high fidelity model of the vehicle runs in real time and sends the vehicle state to the actual electronic control unit which believes that it is connected to an actual road vehicle. This approach allows safe and realistic testing of a large variety of possibilities in a lab setting before the actual road testing stage.

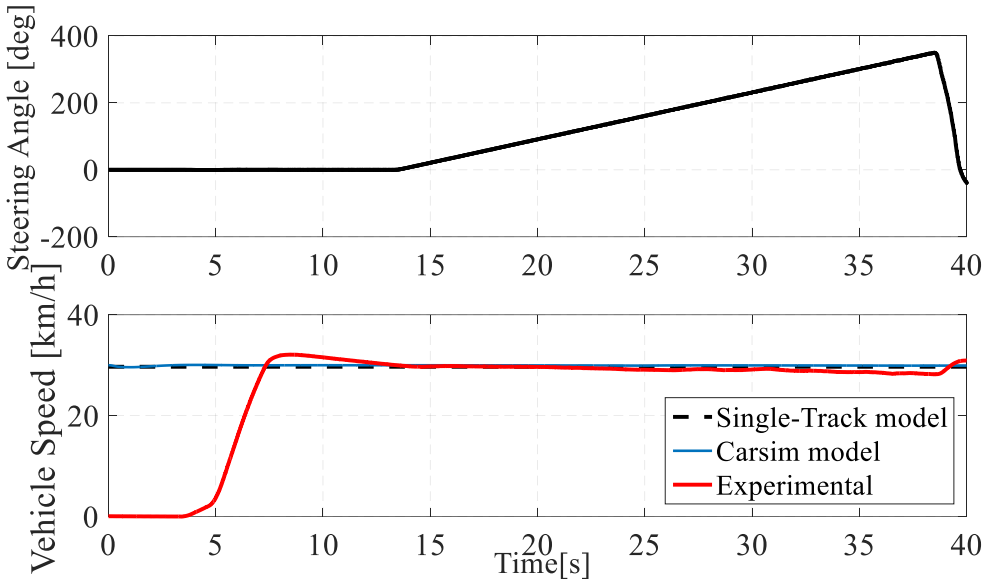

Fig. 7. Steering wheel angle and vehicle speed during ramp steer maneuver.

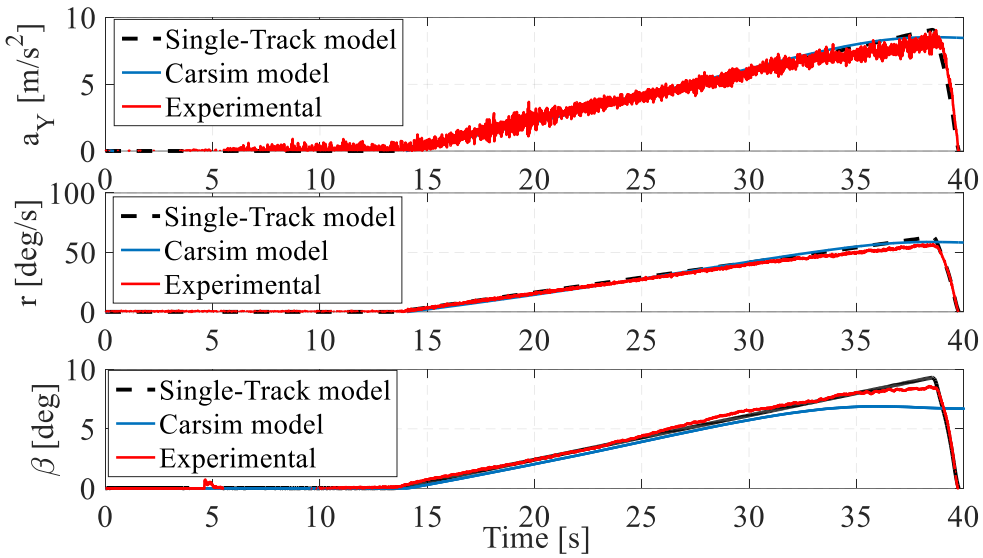

Fig. 8. Comparison of experimental and simulated maneuvers for the ramp steer test.

Along with extensive model-in-the loop simulations in Simulink and CarSim, Hardware-In-the-Loop simulation was also used to make sure that the elastic band based socially acceptable collision avoidance method of this paper worked without problems in a real time implementation. The Automated Driving Lab's state-of-the-art Connected and Autonomous Driving Hardware-In-the-Loop (HiL) Simulator was used with the validated CarSim model of the Ford Fusion Hybrid autonomous vehicle in Fig. 1 for this purpose. The HiL simulator shown in Fig. 9 consists of a dSpace Scalexio system which runs CarSim Real Time with Traffic and Sensors and is connected to a dSpace microautobox control unit and two DSRC radios for V2X communication. Traffic is added as kinematic objects. There are soft front looking (radar, camera, Lidar) and side (radar, Lidar) sensors and a soft GPS sensor. One of the DSRC radios is connected to the ego vehicle during real time simulations and the other DSRC radio represents the rest of the communication from the infrastructure and other vehicles. We built the test track we used in the Vehicle Dynamics Area (VDA) of the Transportation Research Center (TRC) proving ground shown in Fig. 10 in the HiL simulator. The microautobox electronic control unit in the HiL simulator runs exactly the same code as our Ford Fusion Hybrid autonomous vehicle and follows the corresponding route of smoothed GPS waypoints very accurately.

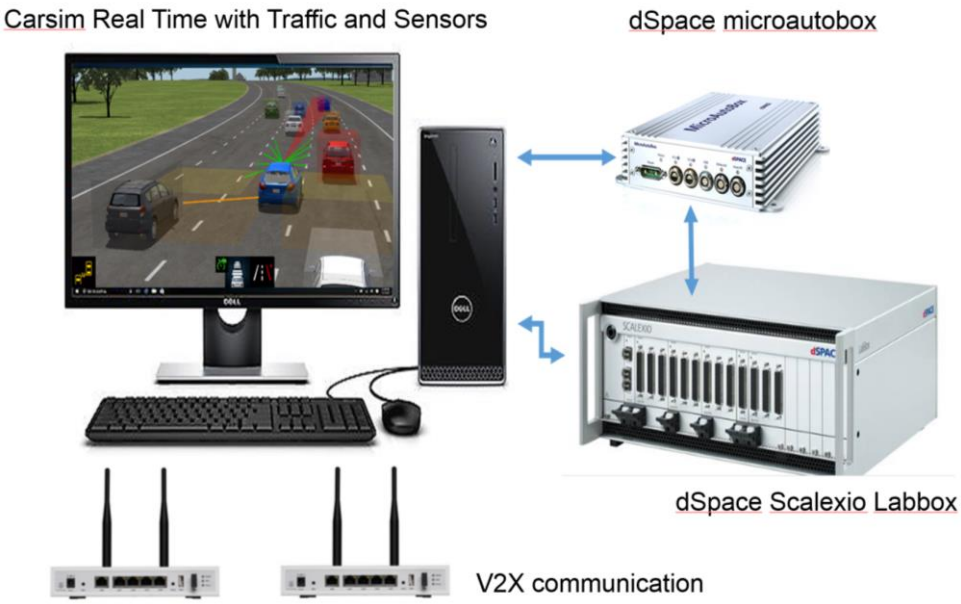

Fig. 9. Hardware-In-the-Loop simulator.

5. Socially acceptable elastic band collision avoidance

Our collision avoidance algorithm is based on the elastic band method [8-9] where socially acceptable distance is considered in modifying the deformed path. The path in low speed autonomous shuttles is formed in a high level planner that extracts GPS waypoints to be followed from a map like OpenStreetMap. These GPS waypoints are smoothed and fit seamlessly with three dimensional polynomial segments [17] and the route to be followed is obtained. The elastic band method works online during the operation of the autonomous shuttle when pedestrians are detected by locally modifying the path around the pedestrians, resulting in collision free navigation.

In the elastic band method, the local path around the obstacle is deformed by defining and using internal and external forces acting on the band. The band is a sequence of displaceable nodes denoted by N_i in Fig. 11. The nodes initially correspond to the original local path of the autonomous shuttle in the vicinity of the detected obstacle. The initial positions of the nodes N_i with respect to the obstacle are shown by position vectors $\mathbf{r_i}$. Internal forces are formulated by adding springs with

Fig. 10. Vehicle dynamics area in the TRC proving ground and the path in the simulator.

stiffness k_s and spring force $\mathbf{F}_{i,j}^{int}$ acting on node i due to the adjacent nodes N_j with $j=i+1$ or $j=i-1$ for $i=1,2,\ldots,n$. The function of internal forces is to hold the nodes or the local path together as a displaceable part of the route of the autonomous shuttle as shown in Fig. 11. External forces $\mathbf{F}_i^{ext}$ acting on node N_i with $i=1,2,\ldots,n$ are defined once an obstacle is detected to deform the band and hence the local path away from the obstacle like artificial potential field forces. The external forces keep deforming the local path around the obstacle which may be moving while the internal forces keep the nodes together in the form of a collision free path to be followed. $\mathbf{u}_i$ for $i=1,2,\ldots,n$ are the deformations of the nodes under the action of external and internal forces when an obstacle is detected. The internal forces $\mathbf{F}_{i,j}^{int}$ become $\mathbf{F}_{i,j}^{int^*}$ after the deformation of the local path. After an obstacle is detected, external forces are applied and the nodes of the deformed path become the new positions $\mathbf{r}_i+\mathbf{u}_i$ for $i=1,2,\ldots,n$ as determined by the balance of internal and external forces acting on the nodes. This method can be applied locally to the path in the vicinity of static or dynamic pedestrian(s) (obstacle) and can be implemented as a real time collision avoidance method.

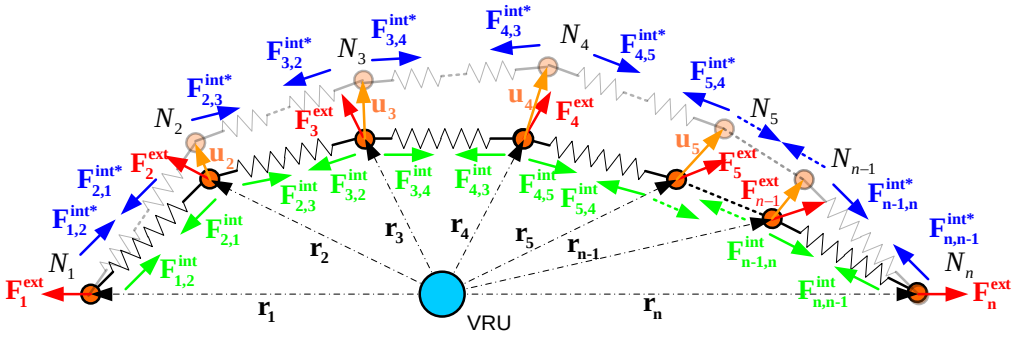

Fig. 11. An initial path deformed by internal and external forces by a vulnerable road user obstacle.

The static balance of internal forces acting on node N_i in Fig. 11 before an obstacle is detected are

$$\mathbf{F}_{\mathbf{i,i-1}}^{\mathbf{int}}+\mathbf{F}_{\mathbf{i,i+1}}^{\mathbf{int}}=k_s\left(\mathbf{r}_{\mathbf{i-1}}-\mathbf{r}_i\right)+k_s\left(\mathbf{r}_{\mathbf{i+1}}-\mathbf{r}_{\mathbf{i}}\right)=0 \tag{4}$$

After the obstacle is detected and external forces are applied, the static balance of internal and external forces acting on node N_i in Fig. 11 are

$$\mathbf{F}_{\mathbf{i,i-1}}^{\mathbf{int*}}+\mathbf{F}_{\mathbf{i,i+1}}^{\mathbf{int*}}+\mathbf{F}_{\mathbf{i}}^{\mathbf{ext}}=k_s\left(\mathbf{r}_{\mathbf{i-1}}+\mathbf{u}_{\mathbf{i-1}}-\left(\mathbf{r}_i+\mathbf{u}_{\mathbf{i}}\right)\right)+k_s\left(\mathbf{r}_{\mathbf{i+1}}+\mathbf{u}_{\mathbf{i+1}}-\left(\mathbf{r}_{\mathbf{i}}+\mathbf{u}_{\mathbf{i}}\right)\right)+\mathbf{F}_{\mathbf{i}}^{\mathbf{ext}}=0\,, \tag{5}$$

which using the identity in Equation (4) becomes

$$\mathbf{F}_{\mathbf{i}}^{\mathbf{ext}}=-\left[k_s\left(\mathbf{u}_{\mathbf{i-1}}-\mathbf{u}_{\mathbf{i}}\right)+k_s\left(\mathbf{u}_{\mathbf{i+1}}-\mathbf{u}_{\mathbf{i}}\right)\right]=-k_s\left(\mathbf{u}_{\mathbf{i-1}}-2\mathbf{u}_{\mathbf{i}}+\mathbf{u}_{\mathbf{i+1}}\right). \tag{6}$$

The external force $\mathbf{F}_{\mathbf{i}}^{\mathbf{ext}}$ acting on node N_i is calculated as a repulsive force using

$$\mathbf{F}_{\mathbf{i}}^{\mathbf{ext}}=\begin{cases}\left(\mathbf{F}_{\mathbf{i}}^{\mathbf{ext}}\right)_{\max}, & \left|\mathbf{r}_{\mathbf{i}}\right|<d\\ -k_e\left(\left|\mathbf{r}_{\mathbf{i}}\right|-r_{\max}\right)\dfrac{\mathbf{r}_{\mathbf{i}}}{\left|\mathbf{r}_{\mathbf{i}}\right|}, & d\le\left|\mathbf{r}_{\mathbf{i}}\right|\le r_{\max},\\ 0, & \left|\mathbf{r}_{\mathbf{i}}\right|>r_{\max}\end{cases} \tag{7}$$

where $|...|$ denotes the magnitude of the argument. $\left(\mathbf{F}_{\mathbf{i}}^{\mathbf{ext}}\right)_{\max}$ in Equation (7) is used to saturate the repulsive force within $\left|\mathbf{r}_{\mathbf{i}}\right|<d$ so that it does not go to infinity as $\left|\mathbf{r}_{\mathbf{i}}\right|\to 0$. k_e is the stiffness associated with the repulsive force and r_{max} is the range of the repulsive force. Once a pedestrian(s) is/are detected and localized with respect to the path of the vehicle, Equations (6) and (7) are solved to obtain the new coordinates

$\mathbf{r}_i+\mathbf{u}_i$ for i=1,2,…,n of the locally deformed path. In the case of a moving pedestrian(s), the computations are repeated at each time step to continue to locally deform the obstacle avoidance path to be followed.

The distance d in Equation (7) is also used to model the physical dimension of the obstacle and our own vehicle $d_{vehicle}$. Noting that $|\mathbf{r}_i| < d$ is a circular region around the obstacle to be avoided, d is adjusted such the obstacle including any short duration displacement between two obstacle detection sampling instants is enclosed by that circle. In the case of a moving obstacle, the circular region $|\mathbf{r}_i| < d$ keeps moving with the obstacle, requiring the local path modification calculations based on solving Equation (6) and (7) to take place within the steering control sampling time.

In socially acceptable collision avoidance, the obstacle is a pedestrian or group of pedestrians and the circular region $|\mathbf{r}_i| < d$ is increased to also accommodate a socially acceptable distance. When the obstacle to be avoided is a pedestrian or group of pedestrians, d is calculated using

$$d = d_{vehicle} + d_{pedestrians} + d_{social}, \tag{8}$$

where $d_{vehicle}$, $d_{pedestrian}$ and d_{social} account for our own vehicle dimensions, the pedestrian group dimensions including their possible motion between two perception sensor detection sampling instances and the social acceptance distance for pedestrians. The use of Equation (8) is illustrated in Fig. 12. In Equation (8), $d_{pedestrian}$

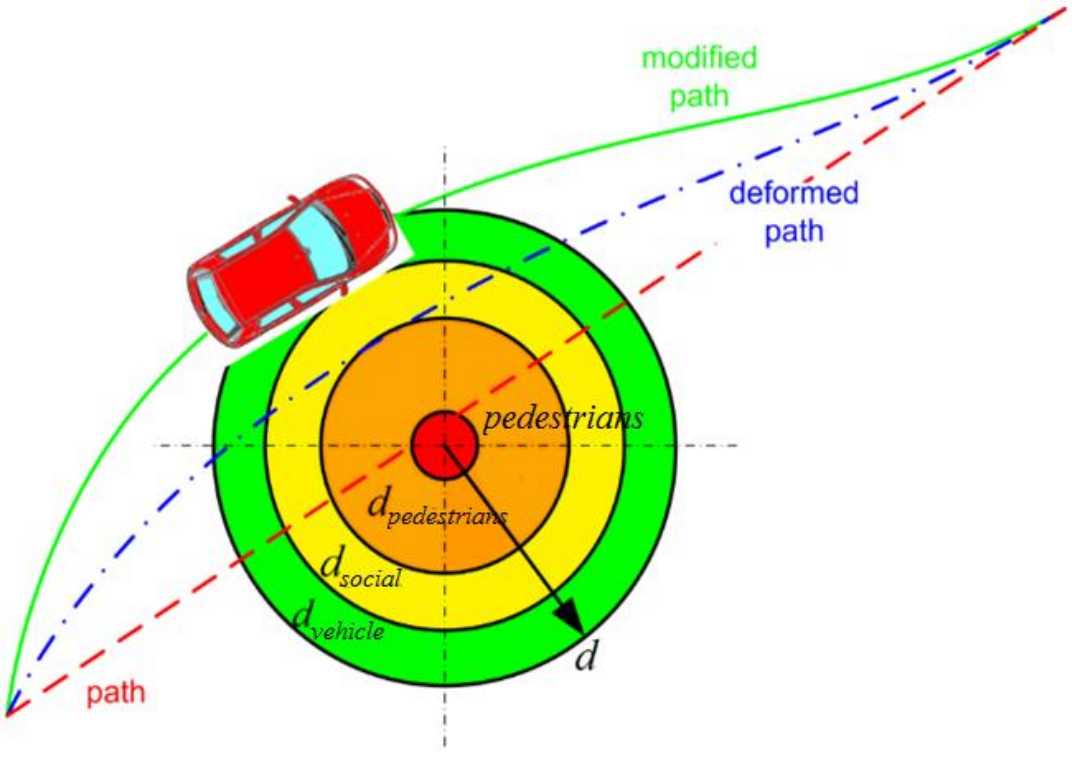

Fig. 12. Safety region around the pedestrians.

stands for the distance that the pedestrian(s) may advance by during the collision avoidance maneuver and is calculated using a conservative time-to-collision (TTC) measure as

$$TTC = \frac{s}{V} = \frac{d_{pedestrian}}{V_{pedestrian}}, \tag{9}$$

where V is the vehicle speed and s is defined as the distance from the vehicle to the the circle of radius $d_{pedestrian}$ around the pedestrian(s). A conservative calculation is used by assuming that the collision condition may occur if the time that the vehicle reaches the pedestrian safety zone boundary is the same as the time it takes the pedestrian(s) to reach the same boundary. This is conservative because both the vehicle and the pedestrian(s) are assumed to be moving directly towards each other along the same line connecting their initial positions. In Equation (9), $d_{pedestrian}$ is calculated as $V_{pedestrian}s/V$ since s is the pedestrian(s) detection distance with respect to the vehicle, V is the current vehicle speed and $V_{pedestrian}$ is estimated based on the literature as having a maximum value of 1.5 m/sec [24].

The social distance d_{social} represents the personal space of the pedestrian(s) that has to be respected by the autonomous shuttle. The idea of social distance is borrowed from the work of Shiomi *et al* [14] who treat a mobile robot operating in a closed mall environment. The current work, however, concentrates on autonomous shuttles and not mobile robots and road environments instead of closed malls. So, there are also big differences in the application being considered. There is a large literature on pedestrian and pedestrian dynamics modeling that is useful as a beginning point. Qian *et al* [25], for example, model social distances within and between groups of pedestrians from both a qualitative and a probabilistic point of view. The social forces model of determination of social distance presented in Was *et al* [26] is more useful for this paper as they also report values like 50-150 cm as personal distance, 1.5 to 3 m as social distance and above 3 m as public distance. This paper treats pedestrian(s) and an autonomous vehicle in an outdoor environment which is a different situation than those in the references but a social distance d_{social} of 1.5 to 3 m is used here as a starting point.

Note that the method presented here works naturally for the case of multiple pedestrian(s). If the pedestrian(s) are close to each other, they are treated as one obstacle group within a safety region as in Fig. 12. Multiple pedestrian(s) that are separated from each other are also easily treated using the elastic band method as they only result in extra external forces in the formulation of Equations (5) and (6). The internal forces in Equation (4) are not affected by pedestrian(s) at different locations. Note that the static model of interconnected springs of the elastic band method can be seen to be similar to the viscoelastic hybrid models used in vehicle crash simulation [27] where crashing vehicles are represented by mass-spring-damper models, at first sight. However, the spring model is used here for deforming a path around an obstacle (pedestrians) and crash/collision is not considered as our main aim is to avoid collisions. In the event that a collision free path could not be found, the method in Pawlus *et al* [27] can be used to model the unavoidable crash of the ego vehicle that is following the deformed elastic band path with the obstacle. Damper forces are not needed in the elastic band deformation

calculations as we are interested in the static deformed path due to the repulsive forces generated by the detected pedestrian(s) (obstacle). This paper is on real time determination and execution of collision free maneuvers by an autonomous shuttle after detecting pedestrian(s) on the path being followed. In the case of a possible crash that cannot be avoided, vehicle materials and structure and side, front positions of the vehicle against the pedestrian(s) will become important in order to reduce the damage due to an eventual crash with the pedestrian(s).

6. Steering controller design

The autonomous shuttle used in the experimental evaluation of the socially acceptable collision avoidance based on the elastic band method presented in the previous section uses a steering controller for path tacking. Fig. 13 shows the low level control architecture of this autonomous vehicle with the collision avoidance algorithm. A cruise controller and a steering controller work in a coordinated fashion for tracking the path that is determined by extracting GPS waypoints from a map. As the path is known in advance, the curvature ρ_{ref} which acts as a disturbance in our model can also be calculated and used in a feedforward controller. Based on the steady state characteristics ($dr/dt=0$, $d\beta/dt=0$) of the single track model, the relation between yaw rate and feedforward steering wheel input denoted as δ_{ff} here is

$$\frac{r}{\delta_{ff}} = \frac{V}{l}\frac{1}{1+KV^2}. \tag{10}$$

Assuming yaw rate r to be given by $r=V/R$ where R is the radius of curvature of the path being followed results in

$$r = \frac{V}{R} = V\rho_{ref} = \frac{V}{l}\frac{1}{1+KV^2}\delta_{ff}, \tag{11}$$

and

$$\delta_{ff} = l\left(1+KV^2\right)\rho_{ref}, \tag{12}$$

where $K = \frac{m}{l_s^2}\left(\frac{l_r}{C_f} - \frac{l_f}{C_r}\right)$ and $l = l_f + l_r$.

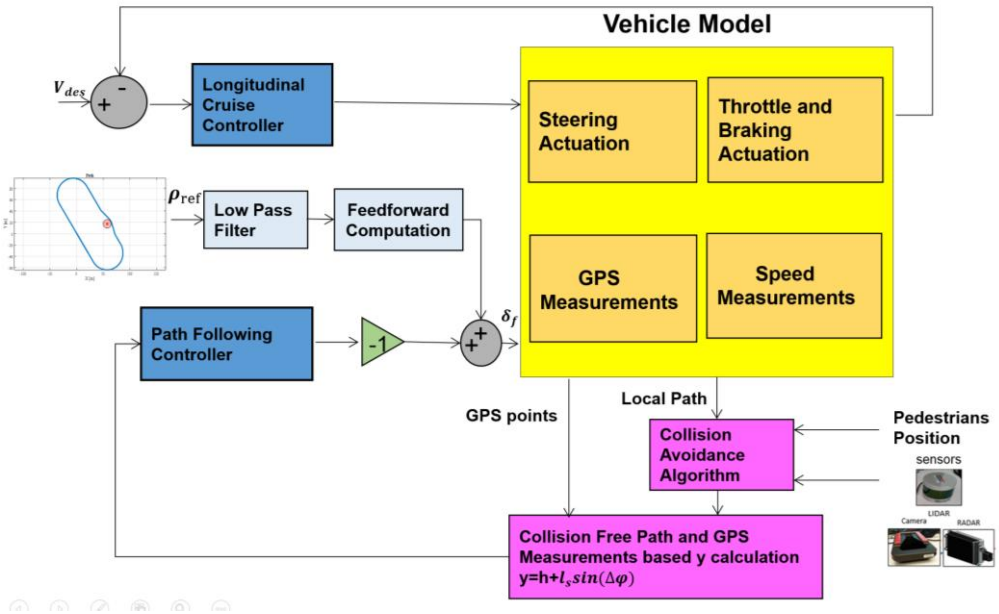

Fig. 13. Low level control architecture with collision avoidance.

As feedforward steering control will not be able to handle uncertainties and other disturbances that may take place during real world driving, it is complemented by a feedback steering controller. The feedback steering controller used is a proportional plus derivative (PD) steering controller designed using parameter space robust controller design for D-stability [10, 28]. While Hurwitz stability requires all closed loop system poles to lie in the left half complex plane, D-stability constrains these poles to lie within a D-shaped closed boundary in the left half plane bounded by a settling time constraint (D1), a damping ratio or overshoot constraint (D2) and a bandwidth constraint (D3) as illustrated in Fig. 14. For purposes of comparison a separate PD controller was tuned using readily available tuning tools available in the Simuink^{TM} PID tuner of Matlab^{TM}. This will be referred to as the tuned PD as opposed to the parameter space PD in the rest of the paper.

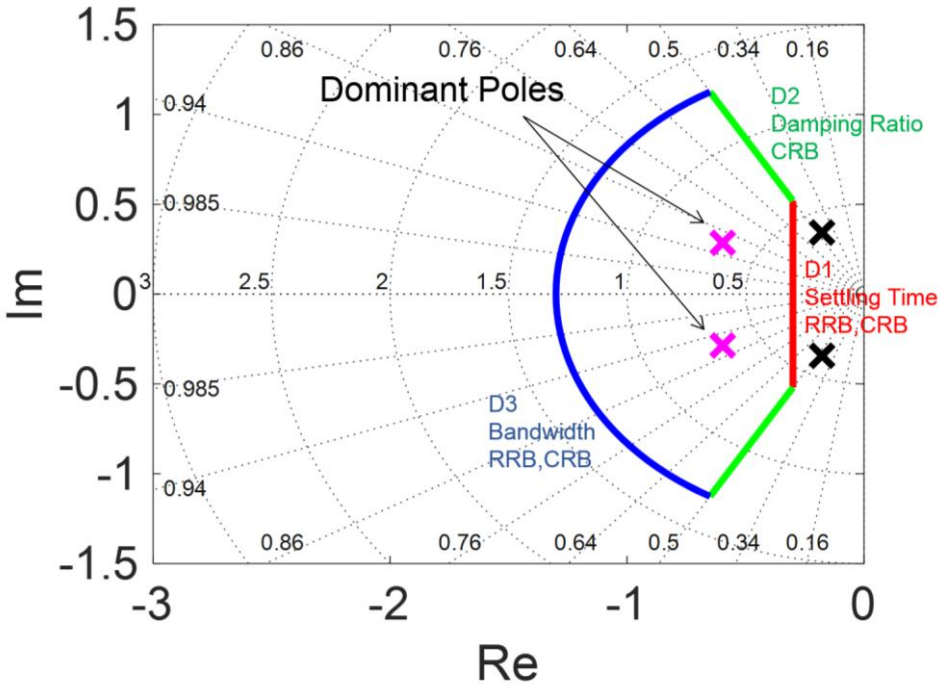

Fig. 14. D-stability boundary.

The Boundary Crossing Theorem [10] is used in calculating the boundaries in the K_P-K_D controller parameter space where Real Root Boundaries (RRB), Complex Root Boundaries (CRB) and Infinite Root Boundaries (IRB) are possible. The Boundary Crossing Theorem states that starting with a D-stable system (all poles inside the D-stability region), changes in controller or plant parameters can result in a D-unstable system if and only if the D-stability region boundaries are crossed as the controller and plant parameter changes are taking place. Determination of which parameter values (controller gains K_P-K_D in this case) result in a crossing of the D-stability boundary and plotting these in the parameter space results in the solution as one (if any) of the closed regions obtained in the parameter space is the solution region.

The actual vehicle path following model used in the feedback controller design is presented next. During the experiments, it was seen that the steer-by-wire system introduced a "delayed" behavior. This delay had to be incorporated into the feedback controller design as its negligence can lead to oscillation of lateral deviations from the desired path. The vehicle path following transfer function model becomes

$$\frac{y}{\delta_f}(s) = G(s)\frac{e^{-\tau_d s}}{\tau s + 1} = G(s)\frac{e^{-0.08s}}{0.2s + 1}, \tag{13}$$

where $G(s)$ is the transfer function obtained from Equation (1) and the steer-by-wire system dynamics consists of a first order system with time constant of 0.2 s and a dead time of 0.08 s. A fourth order Pade approximation is used in the controller design to approximate the dead time by a rational transfer function.

The D-stability boundaries are chosen as settling time constraint of σ=0.3 (D1 in Fig. 14), damping constraint of ζ>0.5 (D2 makes a 30° angle with imaginary axis in Fig. 14) and a bandwidth constraint of R=1.3 rad/sec (D3 in Fig. 14). The resulting K_P-K_D controller parameter space is shown in Fig. 15. The Hurwitz stable and unstable regions and the D-stability region and the RRB and CRB boundaries (no IRB boundary for D-stability) corresponding to the three boundary constraints are all displayed in Fig. 15. The PD controller gains are chosen within the D-stable solution region as K_P=0.1 and K_D=0.15 which is marked with a large black dot in Fig. 15.

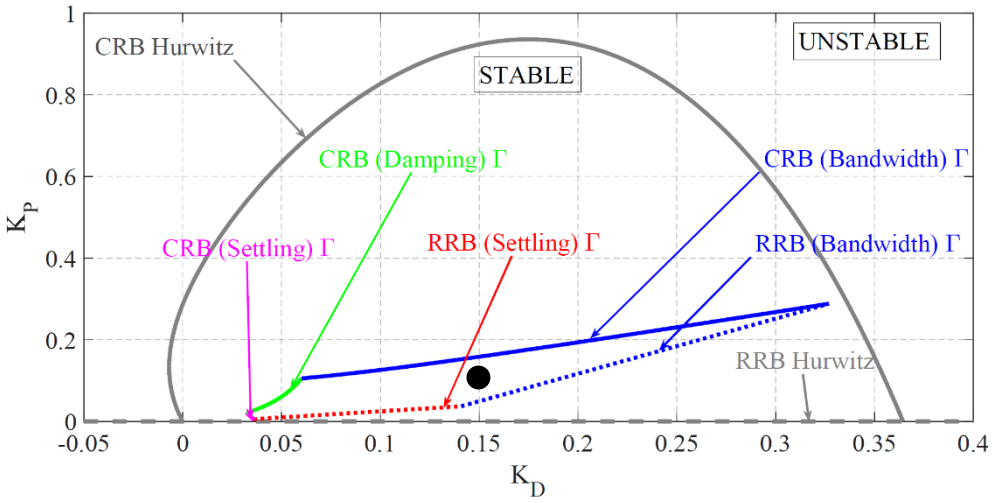

Fig. 15. D-stability region in complex plane and D-stability solution region.

As compared to references [18-20], our D-stability region shape is very different and control system performance oriented. We are also able to easily do multi-objective design in our parameter space approach using other requirements like stability margin or/and mixed sensitivity bounds. The bicycle model, described in the "Vehicle modeling and validation" section, takes into account slip-angles (α_f and α_r) between wheels and ground and their influence, represented by front and rear cornering stiffness (C_f and C_r), in the generation of tires lateral forces. C_f and C_r are highly dependent on load transfer, friction coefficients and suspension kinematics and they are conventionally kept equal to their nominal values for linear controller design. Similar to references [18-20], uncertainty in the front and rear tire cornering stiffnesses was considered according to the uncertainty box in Fig. 16. The parameter space computations for D-stability were computed for the four vertices in Fig. 16 and superimposed graphically as shown in Fig. 17 which demonstrates that the PD controller chosen in Fig. 15 (marked with a red dot) satisfies D-stability for all of those extremal uncertainty combinations in the cornering stiffnesses. The tuned PD mentioned previously in this section has gains of K_p=0.142 and K_d=0.0125 and is marked with a black diamond shape in Fig. 17. The closed loop characteristic equation roots corresponding to the tuned PD are shown with black **X** signs in Fig. 14 and are seen to be outside the chosen D-stability region. This is a disadvantage of using tuning rules in which case there is no direct control over the design specifications.

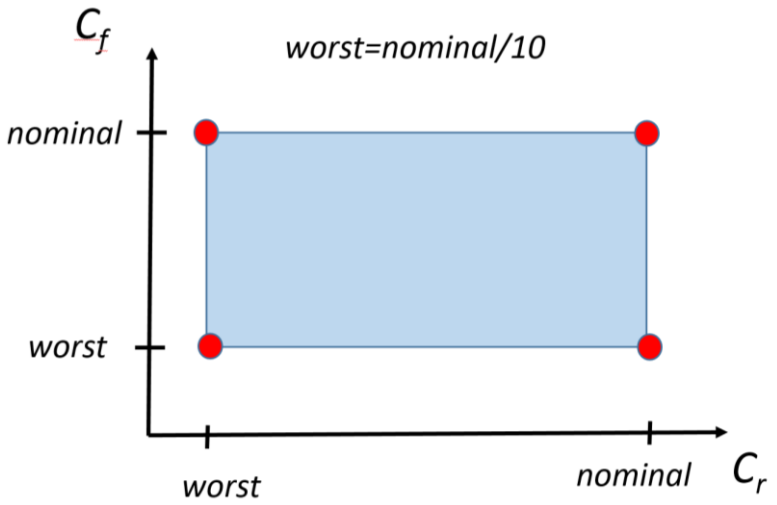

Fig. 16. Uncertainty box of tire cornering stiffnesses.

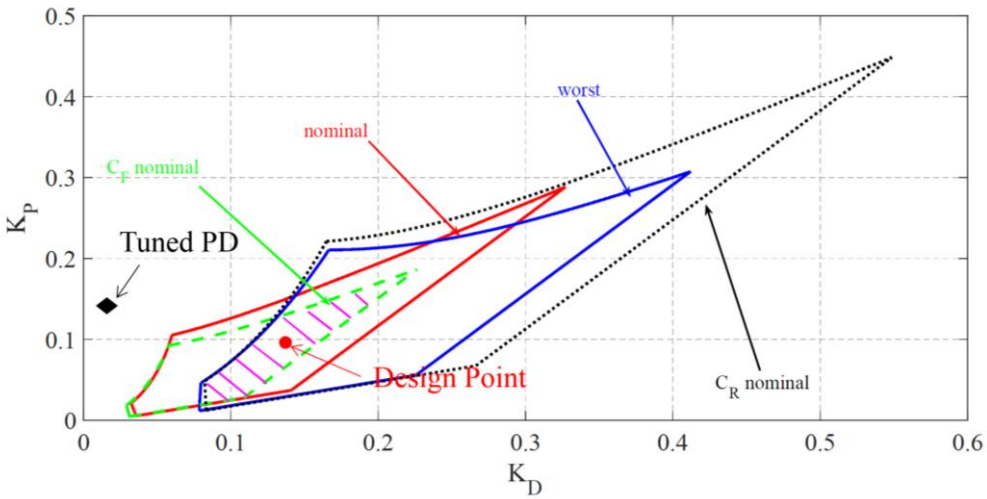

Fig. 17. Overall D-stability region for the four extremal uncertainty combinations in cornering stiffness.

7. Simulation and experimental results

After model-in-the-loop and Hardware-In-the-Loop simulations using our validated Simulink and CarSim models, the proposed elastic band based socially acceptable collision avoidance algorithm was applied to our Ford Fusion autonomous vehicle and tested in the Vehicle Dynamics Area of the TRC proving ground. The safety region of Fig. 12 is placed around the pedestrian(s) that are detected on the pre-determined elliptical route shown earlier in Fig.s 4 and 10 and marked as obstacle in Fig.s 18 and 19. The control architecture in Fig. 13 detects the pedestrian(s) (obstacle) and uses the socially acceptable collision avoidance algorithm by adding an elastic band to the local path around the pedestrian(s) (obstacle) and deforming it for a collision avoidance maneuver during the test.

Fig. 18 shows the planned route and the locally modified path corresponding to simulation and experimental testing of the collision avoidance maneuvering described

above for the PD feedback steering controller alone. Fig. 19 shows the same for the PD plus feedforward steering controller. The first observation is that simulated and experimental path tracking and collision avoidance maneuver responses are very close to each other. It is seen that the collision avoidance algorithm is able to maintain the desired socially acceptable distance and the collision avoidance maneuver is smooth, not causing oscillating behavior of the lateral deviation of the autonomous vehicle.

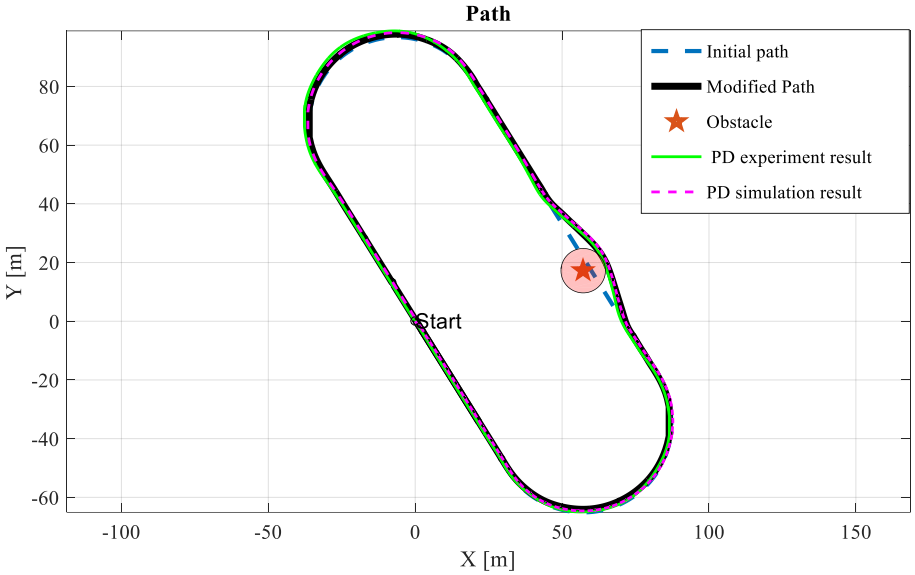

Fig. 18. Simulation and experimental collision avoidance results with PD feedback steering controller.

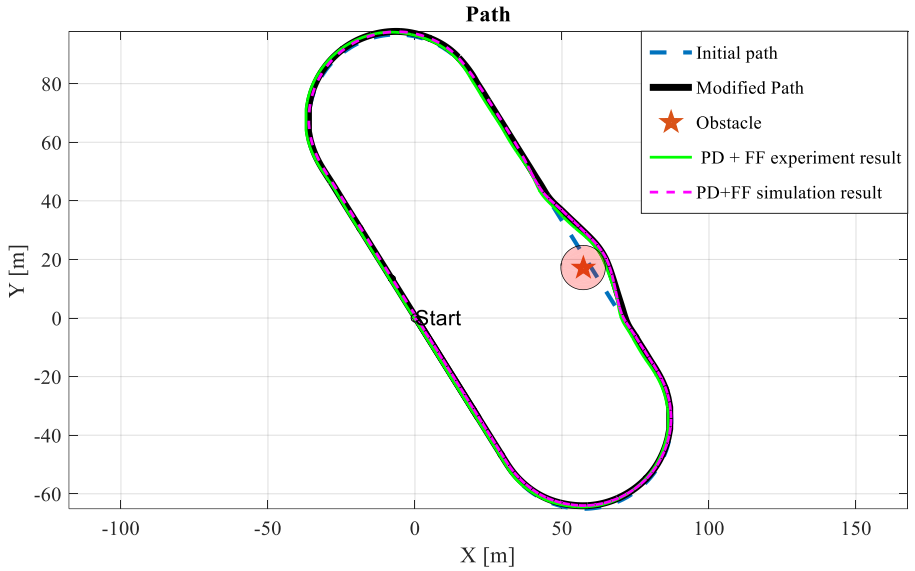

Fig. 19. Simulation and experimental collision avoidance results with PD feedback plus feedforward steering controller.

The vehicle speed was kept at a low value of 5 km/hr in the tests of Fig.s 18 and 19 during the pedestrian(s) avoidance maneuver. Speeds of up to 15 km/hr were used in the rest of the path tracking. The steering actuator command is sent every 10 ms due to CAN bus communication timing and the steering controller sampling time was

therefore 100 Hz. The elastic band collision avoidance algorithm can run at 1 kHz sampling time without any problems. In some of the tests that were not reported in this paper, the road surface was made wet by dumping water from a water truck on the test area used. No appreciable difference in path tracking accuracy was observed at the low vehicle speeds used here.

The simulation results of the vehicle front wheel steering angle δ_f and the lateral deviation y from the desired path at the preview distance are shown, respectively, in Fig. 20 and Fig. 21. The tuned PD results are also shown in Fig. 21 for comparison with the parameter space (PS) robust control results. All simulations were carried out with and without steering feedforward. The first column of Table 2 displays the maximum lateral deviation errors for each of these cases. Analysis of the simulations of Fig. 21 and the simulation data in Table 2 show that the robust PS design gives the best results with and without steering feedforward. The rest of the discussion and the experimental results are, therefore, based on this controller.

Steering angles are at acceptable values and quite similar with both the PD feedback and PD feedback plus feedforward steering controllers in Fig. 18. However, it is seen that the PD feedback plus feedforward steering controller reduces lateral deviation quite significantly as compared to the use of PD feedback alone. The corresponding experimental responses are displayed in Fig. 22 and Fig. 23. The experimental steering angle responses are acceptable and similar for both the PD feedback and PD feedback plus feedforward controllers. The lateral deviation achieved with PD feedback plus feedforward steering control is significantly better than that obtained using PD feedback steering control alone. Table 2 presents a comparison of maximum path tracking errors between the two controllers. It can be concluded that PD feedback plus feedforward controller implementation has better path following performance. While all the general trends are similar between simulated and experimental responses, the individual steering wheel and lateral deviation responses are not too close when simulated and experimental results are compared. From a performance perspective, both simulated and experimental responses are highly satisfactory in collision avoidance maneuvering and in keeping the socially acceptable distance such that this discrepancy is not viewed as a significant issue.

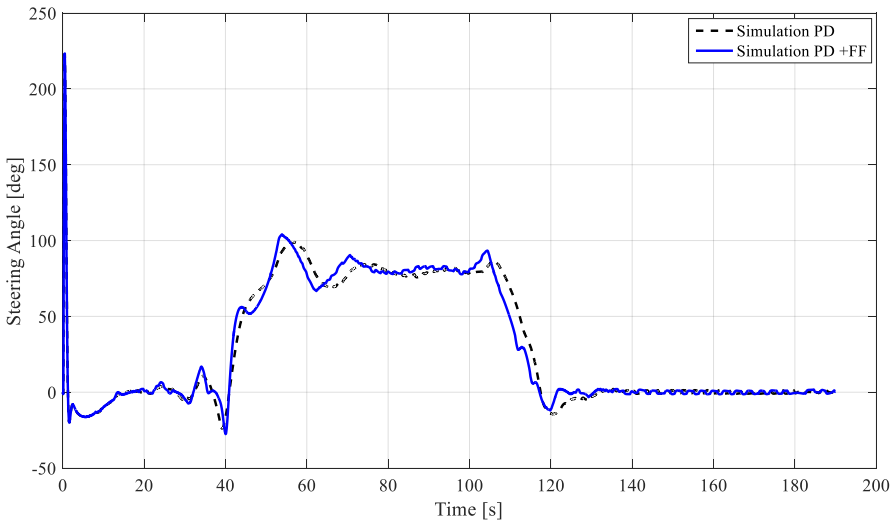

Fig. 20. Simulation results comparison of vehicle steering angle.

Table 2
Comparison of maximum path tracking errors

Steering controller	Simulated maximum path tracking error [m]	Experimental maximum. path tracking error [m]
Robust PS PD feedback	0.748	1.158
Robust PS PD feedback plus steering feedforward	0.271	0.570
Tuned PD feedback	0.807	--
Tuned PD feedback plus steering feedforward	0.529	--

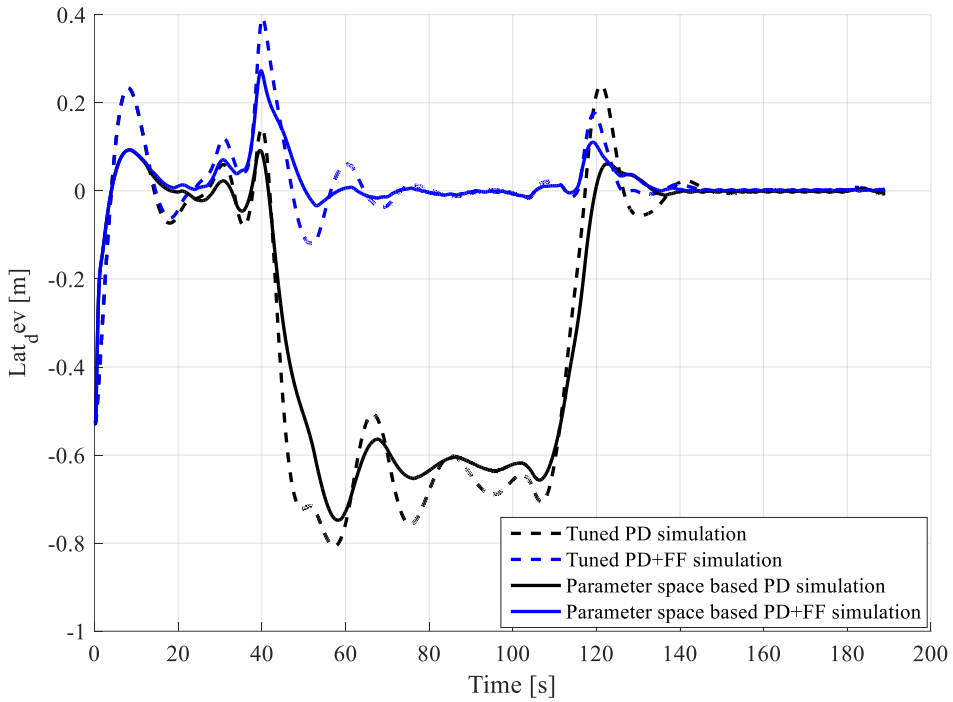

Fig. 21. Simulation results comparison of vehicle lateral deviation.

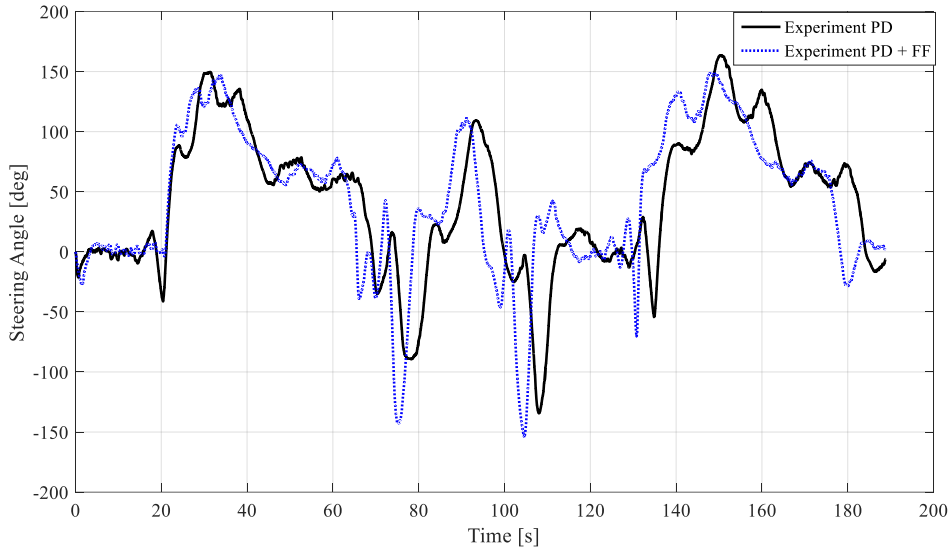

Fig. 22. Experimental results comparison of vehicle steering angle.

The feedforward steering action is seen to be very effective in reducing the lateral path following error in both simulation (Table 2 and Fig. 21) and experiment (Fig. 23). While the steering angle plots with and without feedforward control look similar in Fig. 20 (simulation) and Fig. 22 (experiment), a closer analysis shows that the steering action happens earlier in the case of feedforward plus feedback as opposed to a delayed response in the case of feedback action alone which is the main reason for the difference in path tracking performance.

Note that sensor fault diagnostics and fault tolerant control in the possibility of sensor faults and actuator saturation is an important topic that has not yet been fully worked out for autonomous vehicles. Aouaouda *et al* [29], for example, treat a simulation based study of fault tolerant control of vehicle lateral dynamics control. In this paper, we take steering actuator saturation into account by not using very aggressive steering controllers and checking the steering actuator output during simulations and experiments in order not to saturate it. All of the sensors we use including the steering actuator sensors have their own built-in fault diagnosis and we rely on that. Other than that fault diagnostics and fault tolerant control are outside the scope of this paper.

8. Conclusions

Socially acceptable collision avoidance based on the elastic band method was formulated in this paper for low speed autonomous shuttles operating in high pedestrian density areas. This paper focused on demonstrating the real world applicability of this method. Validated single track based Simulink and CarSim models of automated path following were introduced for model-in-the-loop and Hardware-In-the-Loop simulations which were used to detect and correct any problems in the proposed collision avoidance algorithm before experimental testing. Experimental results conducted in the TRC proving ground demonstrated successful real time application of

smooth collision avoidance maneuvering while keeping the desired social acceptance distance.

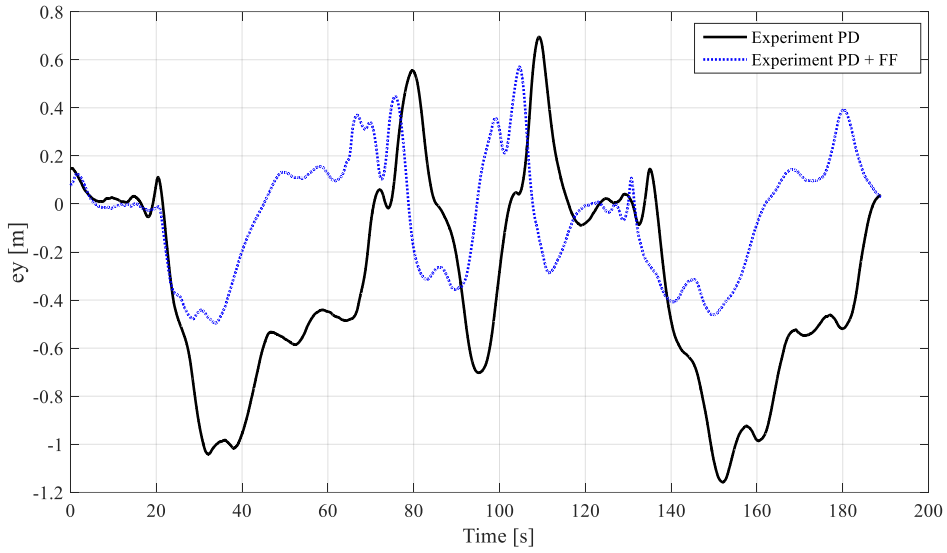

Fig. 23. Experimental results comparison of vehicle lateral deviation.

Acknowledgments

This paper is based upon work supported by the National Science Foundation under Grant No. 1528489. The authors would like to thank the Transportation Research Center for letting them use the Vehicle Dynamics Area for testing and their help during the tests. The authors would like to thank SEA Inc. for letting us use their vehicle testing machines and their help with CarSim model parameter determination. The authors would like to thank Dr. Hongliang Zhou for his help with the CarSim simulations and graduate students Santhosh Tamilarasan and Nitish Chandramouli for their help during the experimental evaluation.

References

[1] Guvenc, L., Aksun Guvenc, B., Emirler, M.T. Connected and Autonomous Vehicles," Chapter 35 in Internet of Things/Cyber-Physical Systems/Data Analytics Handbook, Editor: H. Geng, Wiley, ISBN: 978-1119173649, 2017.

[2] Aufrere, R., Gowdy, J., Mertz, C., Thorpe, C., Wang, C.-C., Yata, T. Perception for collision avoidance and autonomous driving. Mechatronics 2003; 1149–1161.

[3] Ferrara, A., Vecchio, C. Second order sliding mode control of vehicles with distributed collision avoidance capabilities. Mechatronics 2009; 471–477.

[4] Ji, J., Khajepour, A., Melek, W., & Huang, Y. Path planning and tracking for vehicle collision avoidance based on model predictive control with multi-constraints. IEEE Transactions on Vehicular Technology 2016, PP (99).

[5] Fu, M., Zhang, K., Yang, Y., Zhu, H., Wang, M. Collision-free and kinematically feasible path planning along a reference path for autonomous vehicle. IEEE Intelligent Vehicles Symposium, 2015, p. 907-912.

[6] Chu, K., Lee, M., Sunwoo, M. Local path planning for off-road autonomous

driving with avoidance of static obstacles. IEEE Transactions on Intelligent Transportation Systems 2012; 13, 1599-1616.

[7] Quinlan, S., Khatib, O. Elastic bands: connecting path planning and control. Proceedings of the IEEE International Conference on Robotics and Automation, 1993, (2), p. 802-807.

[8] Ararat, O, Aksun-Guvenc, B. Development of a collision avoidance algorithm using elastic band theory. Proceedings of the 17th IFAC World Congress, 2008, p. 8520-8525.

[9] Emirler, M.T., Wang, H., Aksun-Guvenc, B. Socially acceptable collision avoidance system for vulnerable road users. IFAC Control in Transportation Systems, Istanbul, Turkey, May 18-20, 2016.

[10] Ackermann, J., Blue, P., Bunte, T., Guvenc, L., Kaesbauer, D., Kordt, M., Muhler, M., Odenthal, D. Robust control: the parameter space approach. London, U.K., Springer-Verlag, 2002.

[11] Guvenc L., Ackermann, J. Links between the parameter space and frequency domain methods of robust control. International Journal of Robust and Nonlinear Control 2001. 11, 1435-1453.

[12] Aksun-Guvenc, B., Guvenc, L. Robust two degree of freedom add-on controller design for automatic steering. IEEE Transactions on Control Systems Technology 2002, 10 (1), p137-148.

[13] Chan, W.P., Kumagai, I., Nozawa, S., Kakiuchi, Y., Okada, K., Inaba, M. Creating socially acceptable robots: learning grasp configurations for object handovers from demonstrations. In: IEEE Workshop on Advanced Robotics and its Social Impacts, 2013, p. 94-99.

[14] Shiomi,M., Zanlungo, F., Hayashi, K., Kanda, T. Towards a socially acceptable collision avoidance for a mobile robot navigating among pedestrians using a pedestrian model. International Journal of Social Robotics 2014, (6), 443-455.

[15] Vasconcelos, P.A., Pereira, H.N., Macharet, D.G., Nascimento, E.R. Socially acceptable robot navigation in the presence of humans. In: 12th Latin American Robotics Symposium and 3rd Brazilian Symposium on Robotics, 2015, p. 222-227.

[16] Wang, H., Cao, Y., Aksun-Guvenc, B., Guvenc, L. MPC based automated steering of a low speed shuttle for socially acceptable accident avoidance. ASME Dynamic Systems and Control Conference, October 12-14, Minneapolis, Minnesota, U.S., 2016.

[17] Emirler, M.T., Wang, H., Aksun-Guvenc, B., Guvenc, L. Automated robust path following control based on calculation of lateral deviation and yaw angle error. ASME Dynamic Systems and Control Conference (DSCC), Columbus, Ohio, U.S.A., October 28-30, 2015.

[18] Zhang, H., Wang, J. Vehicle Lateral Dynamics Control Through AFS/DYC and Robust Gain-Scheduling Approach. IEEE Transactions on Vehicular Technology, Vol. 65, No. 1, pp. 489-494, Jan. 2016.

[19] Zhang, H., Zhang, X., Wang, J. Robust gain-scheduling energy-to-peak control of vehicle lateral dynamics stabilisation. Vehicle System Dynamics, Vol. 52, No. 3, pp. 309-340, 2014.

[20] Benton, R. E., Smith, D. A static-output-feedback design procedure for robust emergency lateral control of a highway vehicle. IEEE Transactions on control systems technology, Vol. 13, No. 4, pp. 618-623, Jul. 2005.

[21] Conti R., Meli, E., Ridolfi, A., Rindi, A. An innovative hardware in the loop architecture for the analysis of railway braking under degraded adhesion conditions through roller-rigs. Mechatronics, Vol. 24, No. 2, pp. 139-150, Mar. 2014.

[22] Emirler, M.T., Uygan, I.M.C., Gelbal, S.Y., Gozu, M., Boke, T.A., Aksun- Guvenc, B., Guvenc, L. Vehicle dynamics modelling and validation for a hardware-in-the-loop vehicle simulator. International Journal of Vehicle Design Guvenc, 71(1/2/3/4): 191-211.

[23] Unal, K., Guvenc, L. Real-time Hardware-in-the-loop simulation of time to rollover warning for heavy commercial vehicles. International Journal of Heavy Vehicle Systems 2014, 21(2): 105–122.

[24] Knoblauch, R.L., Pietrucha, M., Nitzburg, M. Field studies of pedestrian walking speed and start-up time. Transportation research record. No. 1538, Nov., 1996.

[25] Qian K, Ma X, Dai X, Fang F, Socially acceptable pre-collision safety strategies for human-compliant navigation of service robots. Adv Robot 24(13):1813–1840, 2010.

[26] Was J., Gudowski B., Matuszyk P.J. Social distances model of pedestrian dynamics. In: El Yacoubi Chopard B., Bandini S. (eds.) ACRI 2006 LNCS, vol.4173, pp.492-501. Springer, Heidelberg, 2006.

[27] Pawlus, W., Reza, H., Robbersmyr, K.J. Application of viscoelastic hybrid models to vehicle crash simulation. International Journal of Crashworthiness, Vol. 16, No. 2, pp. 195-205, 2011.

[28] Guvenc, L., Aksun-Guvenc, B., Demirel, B., Emirler, M.T. Control of Mechatronic Systems. the IET, London, 2017.

[29] Aouaouda. S., Karimi, H.R., Chadli, M. Robust static output-feedback controller design against sensor failure for vehicle dynamics. ET Control Theory & Applications, Vol. 8, No 9, pp. 728-737, 2014.