Advanced Emergency Braking with Sensor Fusion

Jacob Buckelew, Yunuo Zhang

I. Project Description

This is the intermediate report for CS6376 Final Project. Our plan is to build an autonomous emergency braking (AEB) system using Simulink, Automated Driving Toolbox, Model Predictive Control Toolbox, and Optimization Toolbox.This system will incorporate sensor fusion and simulate the process of collision avoidance. The final product will be a model of an AEB controller that incorporates the sensors, environment, and vehicle dynamics of the system. We'll then simulate the system with different scenarios to assess safety and liveness properties.

II. Progress

1. We first model the AEB controller.

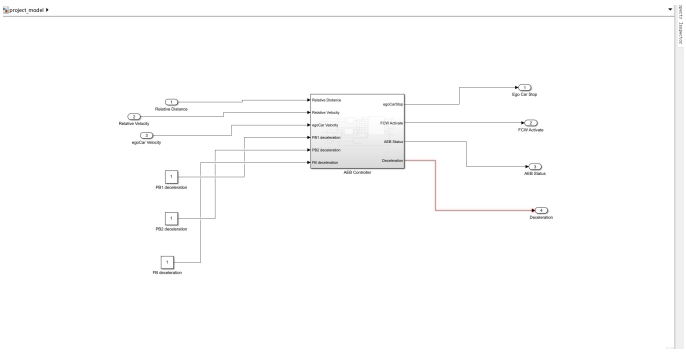

Fig. 1. AEB controller

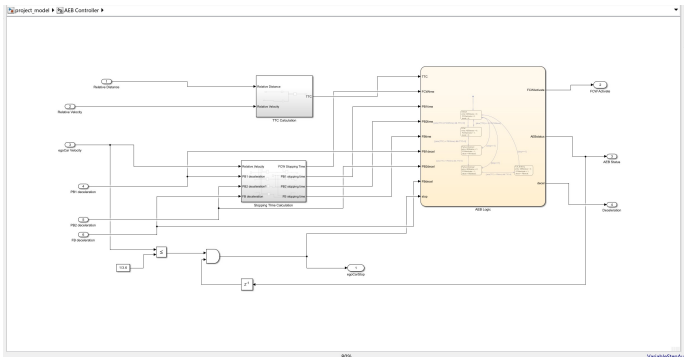

Fig. 2. Inside the AEB controller

As figure 1 and figure 2 show or as the attached model "$AEB_controller.slx$" shows, the AEB controller calculate time-to-collision (TTC), and calculate the stopping time for different braking situations, and the decision logic of AEB controller implemented by Stateflow.

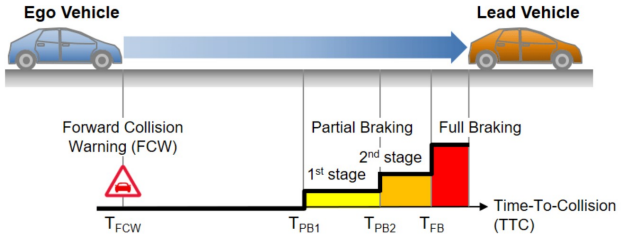

Fig. 3. FCW alert system

A. TTC calculation and stopping time calculation

Time-to-collision(TTC) is defined as the time estimated that will collide with vehicle ahead. As figure 3 shows, we define the time needed for Forward Collision Warning (FCW) begin to alert the driver as the sum of time needed for the driver to react when he/she hear the warning and the time needed for the car to completely stop when the driver applying the brake. We denote the former time as T_{react} and the latter as T_{stop}, then $T_{FCW} = T_{react} + T_{stop}$. We define the controller logic in a way that the FCW alert activates if TTC with the lead vehicle is less than T_{FCW}. And if the driver fails to apply the brake in time, such as due to distraction, the AEB system acts independently of the driver to avoid or mitigate the collision. The AEB system is designed to apply cascaded braking i.e it will receive the signal from the AEB decision logic to decide whether it should apply for 1st stage partial braking or 2nd stage partial braking or full braking. We implemented the AEB automoatically braking in such cascaded way so that we can not only ensure safety by avoiding collision but also providing a best braking experience to avoid "suddenly stop" as much as possible. Figure 4 shows the designing model to calculate stopping time for different types of braking and figure 5 shows the Stateflow Chart that input these information and the chart includes the logic for when should apply which type of braking.

III. Problems

Since we already implemented AEB controller to simulate the FCW warning and AEB braking system with different levels of partial braking and full braking. Next step, we will simulate the AEB model using bird-Eye Scope, and if we have time, we will also extend our model: 1. we will build system to simulate the sensors and environment, for a simulated environment, the system will output the sensors' values such as vision, radar etc. Then we will build system that combines the input of vision and radar detections onto one single output. For the safety and liveness conditions, we will check collision

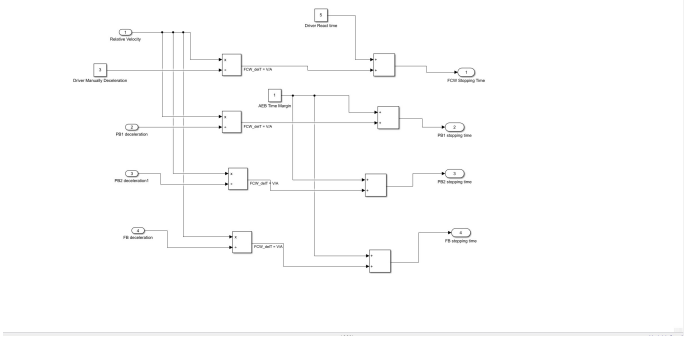

Fig. 4. Stopping time collision

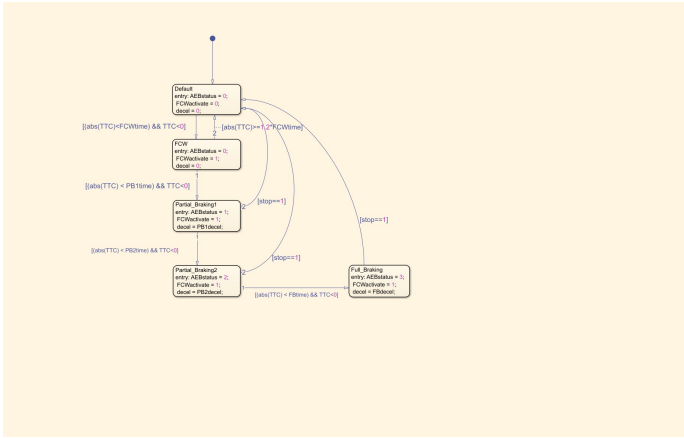

Fig. 5. AEB decision logic

and safety. We will define the safety and liveness conditions as thresholds and verify the ego vehicle is within the threshold throughout the simulation. We are also going to check whether the ego vehicle collide with the target actor at any point during the simulation.

REFERENCES