A PRE-CRASH SIMULATOR TO EVALUATE VEHICLE COLLISION PREDICTION ALGORITHMS

Dana Desrosiers, Charles Birdsong, Ph.D., Peter Schuster, Ph.D.

California Polytechnic State University, San Luis Obispo, California

Abstract: This paper describes a software simulator for pre-crash collision predictions. The simulator is a surrogate test bed for evaluating the performance of proposed pre-crash algorithms. It reads data from a file, transfers distance and angular position of a target to a test algorithm, and then records the algorithm's predictions. To illustrate the simulator functionality, a simplified test algorithm is also described. This algorithm predicts collision risks based on assumptions about the size and acceleration of a target object, and the turning and braking limits of the host vehicle. The test algorithm is shown to be effective for cases where both the vehicle and the target move along straight lines but less effective for curved paths. This result is typical of the difficulty in predicting the future position of another vehicle when its motion may change suddenly in the short time before a crash event.

Keywords: algorithms, computer simulation, automotive control, impact, sensor systems, pre-crash, automotive safety.

1. BACKGROUND

Motor vehicle crashes are the leading cause of death for persons of every age from 2 through 33. Since the 1960s, introduction of passive safety equipment (e.g. seat belts, air bags, crush structures) has dramatically reduced accident rates, injury severity and the number of fatalities; however the absolute number of deaths and injuries remains high. Since 1993, every year nearly 6 million motor vehicle crashes have consistently resulted in over 40,000 deaths in the US alone (NHTSA, 2005). Certain conditions (weather, lighting, impairment, distraction) limit drivers' effectiveness at recognizing and responding to dangerous situations. For example, 50% of fatal accidents occur outside of daylight hours and 12% during inclement weather. Driver distraction is cited as a contributing cause in half of all accidents.

In order to significantly reduce accident severity and occurrence, future safety technologies must move beyond 'passive.' To support this, vehicles will require new exterior pre-crash sensors to create an electronic awareness of the traffic situation (Hover, *et al*, 2006). Pre-crash sensing may well have the most impact in reducing injuries from nighttime accidents involving impaired drivers. However, the advanced safety features enabled by pre-crash sensing will provide a significant benefit in many situations, including poor lighting, bad weather, or driver distraction. Figure 1 illustrates some near-term safety benefits of pre-crash sensing.

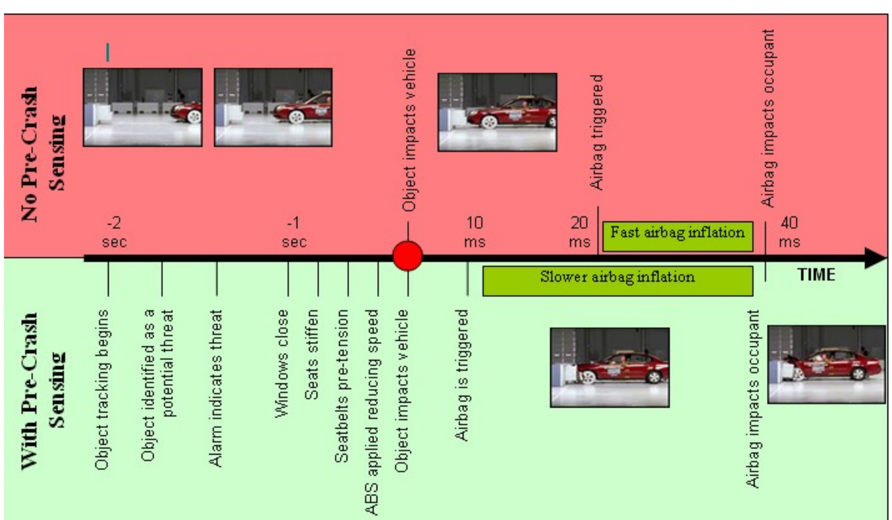

Fig. 1. Timelines for collisions with and without pre-crash sensing.

Current vehicles (top half of the figure) do not have any means of anticipating a crash. In the short time frame (approximately 10-20 ms) after a crash is detected by acceleration-based sensors the options for deploying safety technologies is limited. Currently airbags are deployed approximately 10-20 ms after impact and must be inflated rapidly so that they are in place to protect the passenger. If the crash could be anticipated, additional time would be available to deploy new safety technologies such as audible alarms, seatbelt pre-tensioners, automatic door locks, seat stiffeners, seat position control, window closing, slower airbag inflation rates, and pre-crash braking (Lyons & Taskin, 2000; Spies, 2002; Knoll, *et al*, 2004). The result would be increased vehicle crash survival rates. In addition, pre-crash detection will reduce the incidence of unnecessary airbag deployment. Studies show that unnecessary airbag deployment can cause greater injuries than a minor crash would cause (Jones, 2002).

Beyond the passive safety technologies shown in Figure 1, an advanced pre-crash sensing system will also be capable of directing accident-avoidance technologies. For example, an automated braking system could augment a driver's braking force if the sensor determines more deceleration is necessary to stop the vehicle before impact. With increased sensor robustness, this system could be used to automatically apply the brakes when an imminent crash is predicted; regardless of whether braking is already applied.

2. INTRODUCTION

This work is part of a larger research project at California Polytechnic State University – San Luis Obispo to create and test a complete collision detection system. Prior work (Birdsong, *et al*, 2006; Carlin, *et al*, 2005) includes evaluation and testing of various types of sensors to determine which sensor or combination of sensors is most appropriate for pre crash sensing. The current work is part of a parallel project to develop an algorithm that can process data from multiple sensors to predict automobile collisions.

This paper describes a software simulator developed for use as a surrogate test bed for evaluating the performance of proposed pre-crash algorithms. It reads data from a file, transfers distance and angular position of a target to a test algorithm, and then records the algorithm's predictions. Simulator capabilities are evaluated by testing a simple pre-crash algorithm that makes collision risk predictions based on assumptions about the size and acceleration of a target object, and the turning and braking limits of the host vehicle.

For the purposes of this paper, the host vehicle that is equipped with collision sensing equipment is referred to as "the vehicle." The object that the system is tracking as a possible threat is referred to as "the target."

3. THE PRE-CRASH SIMULATOR

The pre-crash simulator consists of a graphical user interface coupled with a set of simple calculation tools. The user enters fixed parameters, chooses a dataset that represents the motion of the vehicle and target, and then observes a graphical display of vehicle positions and test algorithm outputs for each time in the dataset. When a test algorithm determines that if the target and vehicle remain on their current paths a collision will eventually occur, it is said that the target is on a "collision course". In this case it may or may not be possible to avoid the collision. If the vehicle cannot avoid a collision by turning or braking, it is said that a "collision is imminent."

3.1 Simulator Interface

The simulator interface developed in this work is a simple but efficient tool for visualizing the motion of the vehicles and the output of the algorithm being tested. The interface (Figure 2) allows the user to input fixed parameters:

- Width of the vehicle (m)
- Width of the target object (m)
- Vehicle turning radius (m)
- Vehicle maximum braking deceleration (g)
- Sampling frequency of the sensors (Hz)

The user then selects a data file to analyze from the File menu. The data file is a list of absolute X and Y positions for both the target and vehicle, in meters. Velocity is calculated by multiplying the change in position by the sensor frequency. The user may then view the motion in absolute mode (relative to a fixed observer), or relative mode (relative to the host vehicle). This selection is made from the View menu. The user then scrolls through the data file by dragging the scroll bar at the bottom of the display. The user may also click on the Next button to move to the next data point or click on the Go button to automatically move through the data file at a fixed rate to animate the motion. The position of the vehicles and the simulator output, distance, speed, time to collision, etc. are updated for each data point.

3.2 Data Passed to Algorithm

The first step in pre-crash prediction is locating a target. In an actual vehicle, the physical sensors perform this function. The simulator assumes this has taken place, and reads rectangular (X, Y) coordinates representing the position of the vehicle and one target from a data file. This data is used to define the distance and angular position of the target relative to the vehicle. The test algorithm is then passed the relative target position and absolute

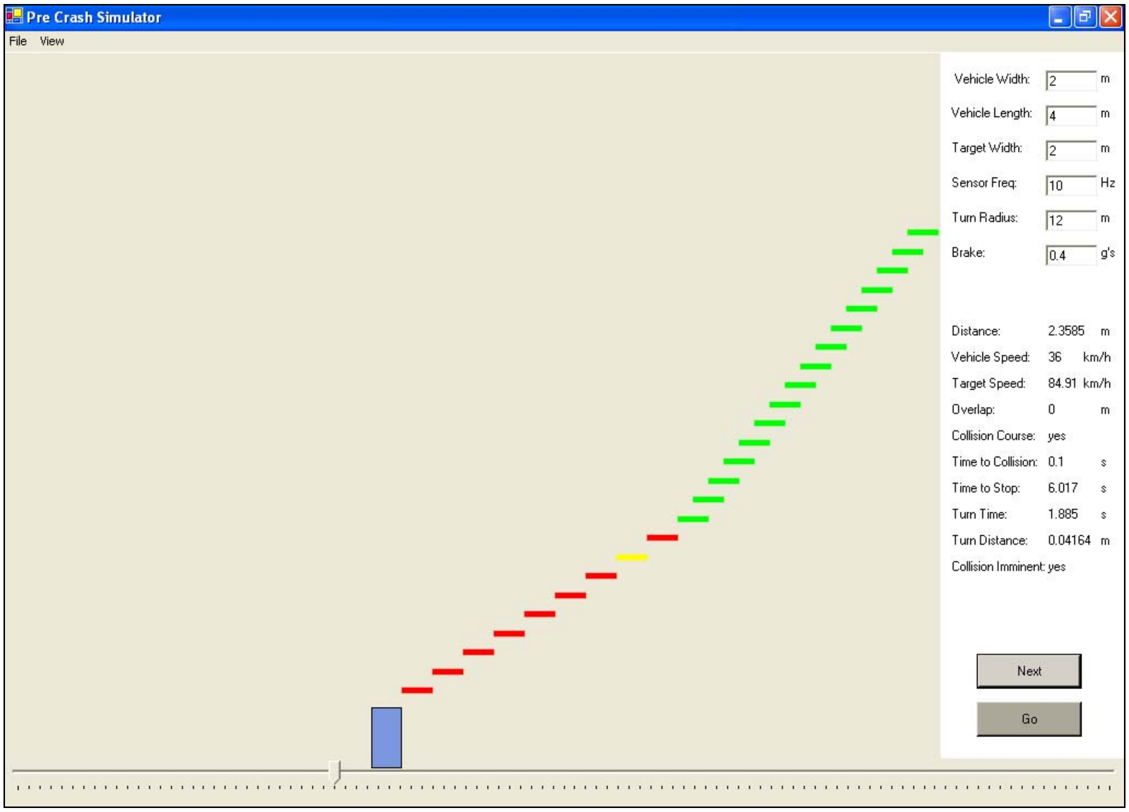

Fig. 2. The simulator software interface allows the user to define fixed parameters, read a data file and visualize the motion of the vehicles and the pre-crash algorithm output

vehicle position data for analysis. In an actual vehicle, the data gathered from a sensor would provide the relative distance and angular position while vehicle speed and position would be provided by other on-board sensors. By using absolute positions in the input data file, the simulator data is easier to generate externally.

3.3 Algorithm prediction of the target path

The simple test algorithm used for illustration purposes in this paper calculates the distance the target has traveled during one time step based on one current and one historical data point. With knowledge of the sensor sampling frequency, a speed is calculated for the target. One historical speed value is kept to calculate target acceleration.

If the target is moving towards the vehicle (relative distance decreasing), a future path line equation is calculated in slope/intercept form using the two most recent data points (Figure 3). The y-axis defines the centerline path of the vehicle, while the x-axis is the plane at the front of the vehicle. The x-intercept of the target's path can be calculated from the predicted travel path. The algorithm assumes that the sensor is in the middle of the front of the vehicle and then checks whether the x-intercept is within one-half the vehicle width. A collision course is predicted if this is true.

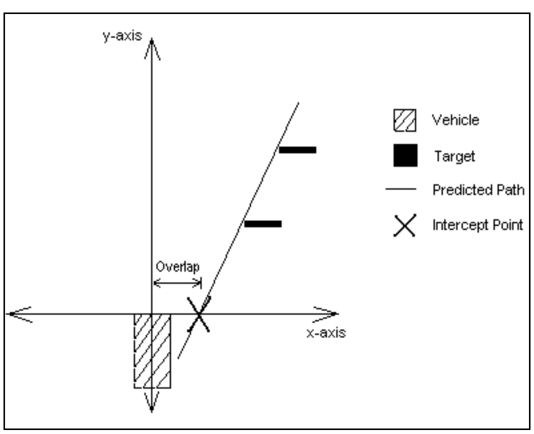

Fig. 3. Collision Course Prediction. The target is not on a collision course because it will not impact the front of the vehicle on its current

3.4 Algorithm crash avoidance prediction

If it is determined that the target is on a collision course, the predicted time to collision (TTC) is calculated using the distance to the current position, the intercept point, the current speed, and the relative acceleration of the target (Kohler , 2004). TTC represents the time that the vehicle has to either brake or turn out of the way before an impact occurs.

The first avoidance alternative considered is vehicle turning. The simplified test algorithm neglects vehicle stability by assuming a turn angle of 90°.

While other turn angles may be more efficient, this assumption provides a simple standard for the purposes of crash prediction. Therefore, the worst case scenario is when the target is coming straight down the y-axis. In this case the vehicle can maximize the x-intercept distance by turning 90°, then driving straight as shown in Figure 4.

Using the speed and turning radius of the vehicle, the time it takes to complete a 90° turn is calculated as

$$t_{turn} = (\pi/2)*R_{turn}/V_v \quad (1)$$

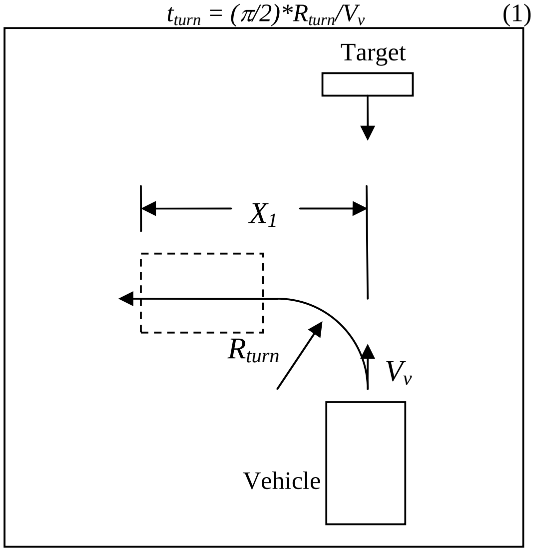

Fig. 4. Algorithm assumes escape path is a 90° turn away from the target to determine if collision avoidance is possible

In this equation, R_{turn} is the known turning radius of the vehicle and V_v is the vehicle velocity. If t_{turn} is less than TTC, the vehicle will be able to translate laterally the distance of its turning radius, plus an additional distance along a straight line as shown in Figure 5. The total lateral displacement possible in TTC seconds in this case is

$$X_1 = R_{turn} + (TTC - t_{turn})*V_v \quad (2)$$

If t_{turn} is greater than TTC, the algorithm calculates how much of the turn is possible before the target reaches the x-axis. The distance that the vehicle can travel in this time is

$$D = TTC * V_v \quad (3)$$

Using this distance as the arc length of the circular path the vehicle takes, the turn angle is

$$\theta = D / R_{turn} \quad (4)$$

With this angle, the lateral distance traveled by the vehicle is

$$X_2 = R_{turn}*(1 - \cos(\theta)) \quad (5)$$

This X_2 position is compared with the predicted target intercept point to determine whether a collision may be avoided by this turn maneuver.

Note that in addition to vehicle dynamics, risk of other collision types is left out in this simple turn algorithm. As a vehicle turns, a larger surface area is exposed to a side impact. Rather than calculating the decrease in frontal area and increase in side area exposed, the algorithm simply assumes the vehicle width is always measured along the x-axis. This is not a realistic calculation, but is reasonable for the purposes of illustrating the simulator tool.

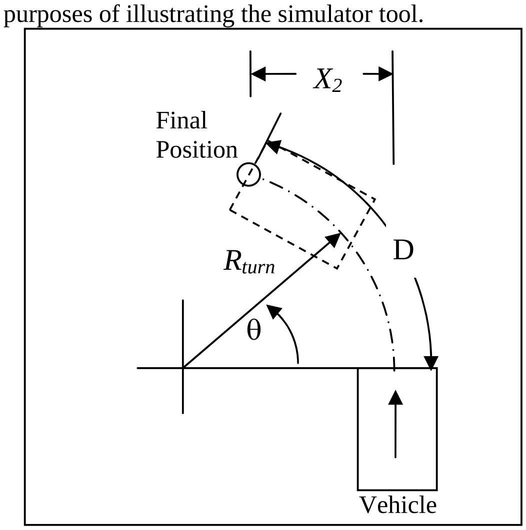

Fig. 5. Algorithm determines if collision avoidance is possible by turning with a constant turn radius

The second avoidance alternative considered is vehicle braking. Specifically, the time it would take the vehicle to stop at maximum deceleration is calculated. If the target is moving backward relative to the vehicle, it is determined that braking will not avoid a collision. If there is not enough time to stop or turn before the target reaches the vehicle, it is determined that a collision is imminent.

3.5 Algorithm Output

The software simulator is capable of displaying the key output statistics expected from a test algorithm:

<u>Distance:</u> The relative distance in meters from the vehicle to the target.
<u>Vehicle Speed</u> (V_v): The actual speed of the vehicle in km/h.
<u>Target Speed:</u> The relative speed of the target in kilometers per hour.
<u>Overlap:</u> The distance in meters from the front center of the vehicle to the point where the target is projected to cross the plane that extends from the front of the vehicle.
<u>Collision Course:</u> "yes" if a collision will occur if current vehicle and target trajectories continue. "no" otherwise.

In addition, when a collision course is predicted, the following data are reported:

Time to Collision (TTC): Predicted time to impact in seconds.

Time to Stop: Time in seconds that it would take for the vehicle to stop at maximum deceleration.

Turn Time (t_{turn}): The time in seconds for the vehicle to complete a 90° turn given its speed and turning radius.

Turn Distance (D): The lateral displacement in meters that the vehicle is capable of achieving using the above escape maneuver given the TTC, turn radius and speed (X in the above equations).

Collision Imminent: "yes" if the vehicle and the target are on a collision course and it is determined that it is not possible to avoid the collision by turning or braking.

4. TEST SCENARIOS

Eleven simulated test scenarios were developed for this project. These are shown in relative mode in Figure 6. They represent basic tests that account for different collision and near miss driving situations. Color-coding is used to indicate the collision risk predicted by the pre-crash test algorithm: green represents no collision course (NC), yellow represents collision course (CC), and red represents collision imminent (CI). The color-coding illustrates the effect of the algorithm at different positions in each crash scenario.

Test 1 The vehicle is stationary and the target is moving toward the vehicle. A collision is imminent at all times because the stationary vehicle can not move to avoid collision. The result is appropriate, but illustrates a potential annoyance condition for a pre-crash warning system. Imagine a driver at a stop light receiving a warning whenever a turning vehicle's instantaneous path intersected the stationary vehicle. More advanced algorithms need to consider how to deal with this scenario.

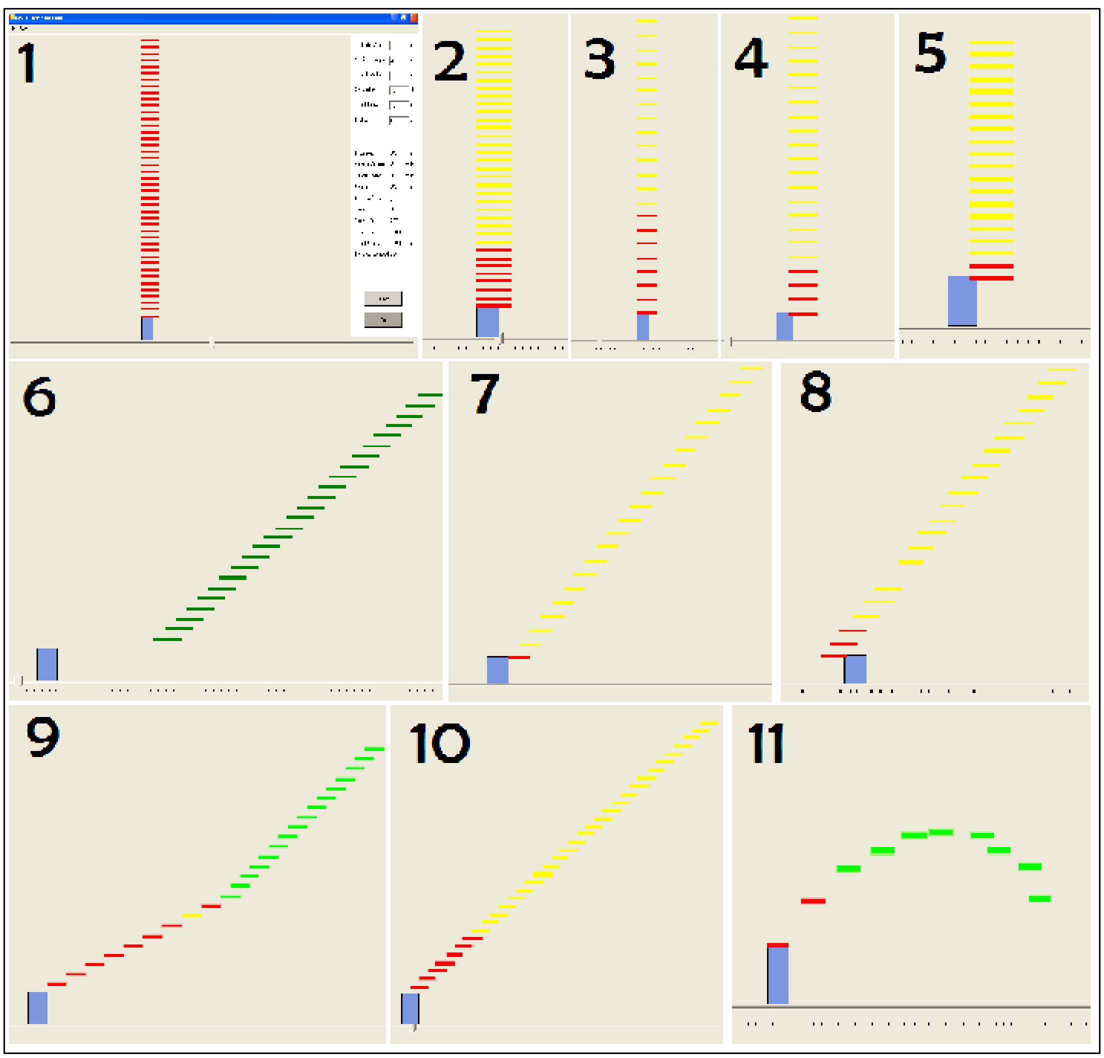

Fig. 6. Test Scenarios in relative view mode. The vehicle is a blue box. The target is green if the target is not on a collision course, yellow if target is on a collision course but a collision can still be avoided, and red if collision is imminent.

Test 2 The vehicle moves forward towards a stationary target. The vehicle is on a collision course and its status changes to collision imminent when neither turning nor braking may be used to avoid a collision. This result is appropriate.

Test 3 The vehicle and the target move toward each other on a collision course. The TTC is shorter than in Test 2 because the relative velocity is higher. This result is appropriate.

Test 4 The vehicle and the target move toward each other on a near-miss course. The simple pre-crash algorithm incorrectly predicts a collision course. This test represents a common situation on city streets, since oncoming vehicles pass nearby. Future pre-crash algorithms need to have an approach to handle this scenario.

Test 5 The vehicle and the target are moving forward, but vehicle is moving twice as fast and overtakes the target. Varying target width shows a near miss or a collision. The simple pre-crash algorithm correctly predicts collision course when the target and vehicle widths overlap. However, when the sizes are close, the algorithm has difficulty (similar to Test 4). This is a common scenario when overtaking another vehicle.

Test 6 The vehicle moves forward as the target moves toward the vehicle at an angle from the top right corner. The target easily clears the vehicle. The pre-crash algorithm correctly predicts no collision.

Test 7 This is another angled approach as in Test 6. Depending on the target width parameter, this shows a near miss or a collision. The target will hit the side of the vehicle, but the sample test algorithm considers only frontal collisions. Future algorithms will need to consider vehicle length and side impact as well as frontal.

Test 8 This test is similar to Test 7, but the target intersects on the left side of the vehicle. The pre-crash test algorithm correctly predicts a collision. Collision imminent is only predicted close to the impact because the vehicle could easily avoid the impact by braking or turning to the right.

Test 9 The vehicle moves forward as the target moves toward it at an angle (as in Test 7), but then abruptly changes direction to a collision course. The sudden change in direction causes a high predicted acceleration. At this instant the algorithm assumes the target continues at this acceleration and therefore predicts a crash is imminent until the next data point is read and the velocity and acceleration are updated. So, the collision course is correctly predicted, but collision imminent is predicted too early. This scenario highlights the difficulty of computing velocity and acceleration from distance measurements.

Test 10 This test illustrates an intersection scenario. The vehicle moves forward while the target moves in tangentially from the right. Collision is correctly predicted.

Test 11 The vehicle and the target are moving forward as in adjacent lanes of traffic and the target swerves in front of the vehicle on a parabolic path. Due to the linear path prediction model used, the target comes quite close to the vehicle before a collision imminent determination is achieved. This case highlights the limitations of the linear path assumption. This situation is a common occurrence on the road, suggesting that more advanced curve prediction may be required.

5. DISCUSSION

5.1 Role of Simulator in Pre-Crash Research

The Pre-Crash Simulator presented in this paper is a valuable tool in developing a pre-crash algorithm. In the early stages of development, proposed algorithms must be tested for a variety of potential crash scenarios in order to determine the algorithms' effectiveness. This evaluation is a complex process that can include both qualitative and quantitative judgments. These judgments are difficult to make without the ability to visualize the potential crash scenario, the behavior of the pre-crash algorithm, and the effects of algorithm changes. The simulator provides this visualization tool. It allows the developer to see the effects of changes in an algorithm and to communicate these effects visually. Ideally the simulator would be modular and allow modifications to the pre-crash algorithm so that proposed changes in the algorithm can then be quickly tested against a standard set of scenarios. The use of the simulator is demonstrated in this paper for a single rudimentary sample algorithm.

5.2 Sample Algorithm Limitations

The sample test algorithm used to demonstrate the simulator is able to predict crash events in many cases but also is limited due to its simplicity. Developing a more advanced pre-crash algorithm is an ongoing process and not the focus of the present paper. The sample test algorithm uses only two data points to calculate a predicted linear path for the target. However, real targets may follow nonlinear relative paths resulting from either target or vehicle motion. This straight-line assumption also ignores any accident avoidance maneuvers by the vehicle or target. This behavior is difficult to predict and has not been considered in the algorithm. To improve accuracy, more historical points could be used to project a curved path. However, without any information about the planned behavior of the target or the vehicle's driver, this additional information may actually result in lower accuracy.

Turning effectively reduces the vehicle's forward velocity. This means that the TTC has increased and the vehicle has more time to get out of the way. The sample test algorithm currently does not account for this. Braking and turning at the same time is a more complex situation that has also not been considered.

Complex vehicle dynamics have been reduced to a simple assumption of a fixed turning radius at any speed. Significant testing or vehicle dynamics modeling would be required to determine the actual characteristics of a given vehicle. For example, if the vehicle is moving 100 km/h it will not be able to turn as sharply as it could at 25 km/h. A fast moving vehicle may not be able to turn very sharply without skidding or rollover. The sample test algorithm assumes a best-case scenario that may be over-simplified. This means that the vehicle cannot turn as quickly as the program assumes and a collision may be imminent earlier than indicated.

In one of the test scenarios (test 9), an imminent collision is predicted when the target changes course. This is because the abrupt change in speed causes a large acceleration that dramatically reduces the time to collision. It is determined that a collision is imminent at the moment of the turn, then changes this conclusion in the next data point when the acceleration returns to zero. This scenario highlights the fact that using distance data to predict velocity and acceleration may produce unacceptable results. A possible solution is to require more than one collision imminent determinations in a row before taking any action or use a sensor that measures velocity directly. This will be more appropriate in real life situations where the update frequencies are much higher than the simulated test data produced thus far.

The Pre-Crash Simulator is currently designed to follow a single object. To be used in real world situations, the algorithm must be extended to locate and track as many objects as necessary. It would at least have to identify and track the most threatening object in the scene. This would require that the program distinguish between objects and constantly re-evaluate which object to track.

6. CONCLUSIONS

Pre-crash detection is a difficult challenge due to the unpredictable nature of human reactions. The algorithm must take into account the vehicle dynamics such as turn radius and maximum braking ability to determine if a collision can be avoided. In addition, the algorithm must also consider the turning and braking ability of the target it is tracking (usually another vehicle). The characteristics of the vehicle can be determined with significant testing and a more advanced dynamic model, but it is almost impossible to know such information about the target.

In the sample test algorithm, the target is assumed to be on a linear course for a number of reasons. First, it is easier to implement, making the algorithm faster and simpler. Second, in a real system, the sensors would update quickly and provide new course data when the target changes direction. And third, the target is equally as capable of turning the opposite direction as it is of continuing on its current curved path. It may be more likely to continue on its current course, but this may not be a good assumption just prior to a collision. Considering all of these points, it is suggested that a linear path will provide a good average of all possible paths.

The algorithm is most accurate when both vehicles are moving in straight lines, such as in a rear-end, straight crossing at an intersection, or head-on collision. These scenarios make up about a third of all collisions in the U.S. (about 2 million incidents) (Sen, 2002). This is a significant portion of all collisions that can be avoided or reduced in severity with the aid of pre-crash warnings and automatic assistance.

REFERENCES

Birdsong, C., P. Schuster, J. Carlin, D. Kawano, W. Thompson and J. Kempenaar (2006). "Test methods and results for sensors in a pre-crash detection system." SAE World Congress, Detroit, USA, March 2006.

Carlin, J., C. Birdsong, P. Schuster, W. Thompson and D. Kawano (2005). "Evaluation of cost effective sensor combinations for a vehicle pre-crash detection system." SAE Commercial Vehicle Engineering Congress, Chicago, USA, November 2005.

Hover, N., B. Lichte and S. Lietaert (2006). "Multi-beam LIDAR for active safety applications." SAE World Congress, Detroit, USA, March 2006.

Jones, R.C. (2002). "Technologies for static airbag suppression systems." 6th International Symposium on Sophisticated Car Occupant Safety Systems, Karlsruhe, Germany, December 2002.

Knoll, P., B.J. Schaefer, H. Guettler, M. Bunse, R. Kallenbach (2004). "Predictive safety systems – steps towards collision mitigation." SAE World Congress, Detroit, USA.

Kohler, M. (2004). "Accurate pre-crash detection." ITS World Congress 2004, Nagoya, Japan.

Lyons, C.T. and I. Taskin (2000). "A low-cost MMIC based radar sensor for frontal, side or rear automotive anticipatory pre-crash sensing applications," IEEE Intelligent Vehicles Symposium, Dearborn, USA.

NHTSA (2005). "Traffic Safety Facts 2003." DOT HS 809 775, www.nhtsa.dot.gov.

Sen, B., B.N. Campbell, J.D. Smith and W.G. Najm (2002). “Analysis of light vehicle crashes and pre-crash scenarios based on the 2000 general estimates system,” DOT HS 809 573, NHTSA, Washington DC.

Spies, H.D. (2002). “What is achievable today and in the near future? - Overview on technologies: Radar, Video, IR.” 6th International Symposium on Sophisticated Car Occupant Safety Systems, Karlsruhe, Germany, December 2002.