The Pennsylvania State University
The Graduate School
College of Engineering

A UNIFIED METHOD TO CALCULATE SAFETY SURROGATES FOR AN AUTONOMOUS VEHICLE IN WORK ZONES

A Thesis in
Mechanical Engineering
by
Marcus Putz

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Master of Science

August 2023

The thesis of Marcus Putz was reviewed and approved by the following:

Sean Brennan
Professor of Mechanical Engineering
Thesis Advisor

Satadru Dey
Assistant Professor of Mechanical Engineering

Robert Kunz
Professor of Mechanical Engineering
Associate Head of Graduate Program of Mechanical Engineering

Abstract

Autonomous vehicles (AVs) have the potential to provide many benefits not only to individuals but also to the economy at large. The progression of this safety-critical technology requires careful assessment, particularly in work zones, as they are some of the most complex areas that vehicles have to navigate. This thesis quantifies metrics of safety to assess how AVs perform in these edge cases through the implementation of surrogate safety measures (SSMs).

More specifically, the measurement and collection of SSMs seek to determine the likelihood and severity of a collision, AV error, near-miss, lane departure, and similar unexpected events. The inputs to SSM calculations include the parameters of a vehicle, such as size, the trajectory of the vehicle, such as speed, position, and orientation, and the scenario description of the driving environment, including the road geometry and trajectories of surrounding vehicles. This thesis proposes a unified method to calculate common SSMs with a precise technique to simplify each measure's calculation and understanding. Specifically, the thesis shows how time-to-collision (TTC), post-encroachment time (PET), time-to-lane crossing (TLC), deceleration rate to avoid crash (DRAC), relative lane position (RLP), and many other metrics all can be unified in calculation by using spacetime coordinate representations and dilation of the physical artifacts in the environment. By converting to spacetime, the calculation of most SSMs is reduced principally to calculations of lengths along different projections using ray casting in spacetime.

Table of Contents

List of Figures

Acknowledgments

I would like to first acknowledge Dr. Sean Brennan for his help throughout the years. I would also like to thank all the friends that I gained though out the time I was in grad school. I would not have been able to survived it without you all. Lastly, I would like to thank Paige for all her help, and for dealing with me throughout this process.

This thesis is disseminated under the sponsorship of the Pennsylvania Department of Transportation in the interest of information exchange. The PA State Department is not endorsing any manufacturers, products, or services cited herein and any trade name that may appear in the work has been included only because it is essential to the contents of the work.

Acknowledgement of Support
This thesis is based upon work supported by the U.S. Department of Transportation under Grant Award No. NOFO693JJ319NF00001. This material is being produced with the support of HNTB Corporation under Contract for the Automated Driving System Demonstration Grant.

Disclaimer
Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the Author(s) and do not necessarily reflect the view of the Pennsylvania Department of Transportation and/or USDOT.

Chapter 1 Introduction

1.1 Background and Motivation

Autonomous vehicles (AVs), colloquially referred to as self-driving cars, have the potential to revolutionize the safety, economics, and the accessibility of transportation and its related infrastructure. Individuals relatively unfamiliar with AVs may understand these to solely include vehicles capable of operating independently of driver input, but most engineers separate functionality into one of six levels. The levels are divided by a vehicle's capability ranging from zero (no automation) to five (fully autonomous). There is, generally, a positive correlation between the desired benefits and the levels of autonomy. At a fully-autonomous level-five operation, due to their capability of driving without human input, AVs could provide independence and mobility to groups that require assistance with driving, such as children and those who are disadvantaged [9, 10]. Even if they are not level-five, AV algorithms can be used to promote efficient driving practices such as truck platooning, smoother acceleration, and coordinated driving, which help to reduce emissions and alleviate traffic congestion, as demonstrated in studies such as [11, 12]. Finally, AVs have the potential to improve driving safety by removing or reducing the possibility of driver error [13, 14]. The National Highway Traffic Safety Administration cites driver error as being responsible for 94% of all accidents, resulting in around 2 million crashes between 2005 and 2007.

Despite the positive outlook on this technology, the general public remains hesitant toward the widespread adoption of AVs. According to a study published by the Worcester Polytechnic Institute, uncertainty regarding the safety of AVs is the primary source of this reluctance [15]. The widely shared reporting of AV failures, including some depicting AV crashes, exacerbates the growing public hysteria [16]. Jefferson and McDonald, [17] performed a study that used social media data and found that after the reporting of an

AV crash, positive sentiments dropped by half while the negative ones stayed the same. If further developments can demonstrate consistent reliability in the function of AVs, public opinion toward widespread adoption could become more positive.

Work zones, in particular, represent a difficult challenge to both human drivers and autonomous vehicle algorithms. Such challenges include unexpected lane changes, slowdowns (either from the reduction of the speed limit or induced congestion), and visual distractions or disruptions related to the site's operation. Work zones can be treated as an edge case for AVs as work zones are likely the most complex region an AV will have to navigate. All this complexity led to 680 deaths from drivers and passengers, and in 2020 a total of 102k crashes due to work zones [18]. Even with all the precautions taken, work zones are still unsafe for the workers. 74.7% of fatalities for highway workers are attributed to being struck by vehicles or workers driving or riding in a vehicle [19]. With work zones being the most complex problems for AVs, and being the area with the greatest ability to gain in safety, research is needed to understand AV operation in work zones and particularly, to evaluate the safety of AVs relative to human operators in these environments.

In this thesis, a vehicles' safety is determined not by analysis of crashes, but rather by using safety surrogate methods (SSMs). These methods allow one to evaluate safety without relying on statistical safety models or post-crash analysis. Statistical safety models use historical accidents and fitting a statistical model to that data to then estimate the safety. Because they do not rely on a statistical safety model, SSMs provides many benefits and are widely used in academic research - examples of which are briefly discussed in Chapter 2. However, the main benefit of using SSMs for this paper is that there is insufficient historical data on AVs to determine their safety. There are many different SSMs, each valid for separate tasks and each having a separate calculation process. These differences can cause problems when trying to process or compare metrics to each other. To solve this issue, this thesis proposes a unified method to represent common SSMs in spacetime. The results demonstrate a greatly improved ease of calculation.

1.2 Goals

This thesis presents a method to determine safety surrogate methods to assess the safety of an autonomous vehicle in a work zone using a technique that unifies the calculation processes across all SSMs. A goal of this is to create a universal framework to simplify the calculation of SSMs such that verification of a small set of SSMS enables interpretation and verification of comprehensive safety in the vehicle behaviors.

1.3 Organization

The remainder of this thesis is organized as follows:

- Chapter 2 is a literature review including:
  - A presentation of previous work on SSMs, explaining how each measure is calculated, and example usage of measure found in the literature.
- Chapter 3 explains a method of unifying SSM calculations using spacetime, including
  - how to visualize a vehicle trajectory
  - the meaning of geometric measures in spacetime relative to the vehicle trajectory
  - how each key items related to common SSMs are calculated in spacetime.
- Chapter 4 presents the results of the previously described methods applied to test situations, including:
  - how each metric is transformed to function in spacetime
  - testing each metric for test situations
- Chapter 5 is a discussion including conclusions and potential further work.

Chapter 2 Literature Review

This chapter gives an overview of previous research on how and why the safety of a vehicle is determined using SSMs. The chapter next describes each SSM in detail, focusing on how they are calculated and implemented. Lastly, this chapter reviews previous research that uses spacetime representations to simplify computation.

2.1 Surrogate Safety Measures

Surrogate safety measures (SSMs) are the methods used to determine the safety of a vehicle without reliance on accident data. Before SSMs were introduced, road safety policy-makers would use historical accident data to determine the safety of a road section or vehicle features. This historical data would be sourced from police reports, hospital data, insurance companies, federal databases (Fatality Analysis Reporting System, or FARS, for example), and other authorities. From such data policy-makers, engineers, regulators, etc., could then determine whether any trends emerge related to features or changes in infrastructure, vehicle design, operational rules, driver activities, etc. Key issues with this data are that it is a retroactive rather than a proactive analysis. It could take a long time to collect such data before any statistical correlations emerge with significance. Further, the collected data may omit important confounding information, such as how the crash happened as a sequence of failures, events, and operational decisions. While such data could be collected from black-box data collection, this could raise some ethical and legal concerns. However, without systematic data collection, there is a high likelihood of accident under-reporting depending on the severity of the accident. [20, 21] A famous model proposed by Hyden (1987) [22], shown in Figure 2.1, states that with the increasing severity of an accident, there is a decrease in the frequency of the accident, thus making severe accidents harder to observe.

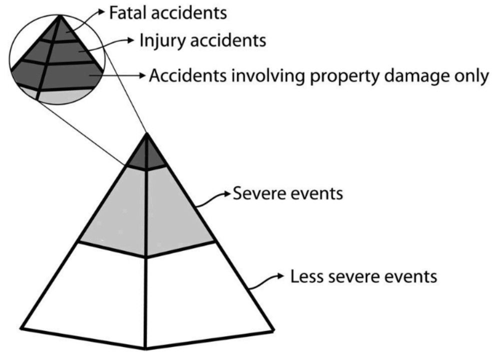

Figure 2.1. Severity levels of traffic events [1]

With these issues regarding historical data and the evidence provided by Hyden, it is clear why SSMs are popular when studying the safety of vehicles - these surrogates infer safety by examining key factors - lane keeping, the proximity of the vehicle to other objects, maneuvering behaviors and limits of the vehicle - that physically and unambiguously relate to the potential for crash events. What makes an SSM, according to, Tarko et al. (2009) [23], is (1) it should be derived from traffic conflicts that are directly linked to crashes, and (2) the relationship between traffic conflicts and the related potential crash frequency and/or severity can be quantified and/or verified by some practical methods. SSMs provide insight into the risks that a vehicle is taking while driving by measuring various properties of the vehicle. Some of these properties are the position relative to other objects and vehicles on the road and the speed or acceleration of the vehicle. One significant benefit that SSMs can provide to AVs is since these properties can be measured in real-time, the riskiness of a vehicle can be quantified. SSMs thus allow the designers of AV algorithms to directly measure SSMs to determine safety measures and trade-offs related to how the AV is driving, thus allowing changes to optimize algorithm performance with direct consideration of safety [24].

Among the earliest use of SSMs in the literature were by Hayward in 1972, with the discussion over time-to-collision (TTC). Since the 1970s, many different types of SSMs have been developed, many of which are simply variants of a few primary SSMs. The primary SSMs seen in literature today include: time-to-collision (TTC), post-encroachment time (PET), deceleration rate to avoid crash (DRAC), speed differential (SD), relative lane position (RLP), time-to-lane departure (TTL), and conflict severity index (CSI). This thesis reviews how the literature defines each of these measures and a seeks clear, unified method for the calculation of these. This chapter organizes the discussion of the surrogate safety measures using the framework created in [24], which divides SSMs into

Time-Based SSMs, Deceleration-Based SSMs, and Conflict-Based SSMs.

2.2 Time-based SSMs

2.2.1 Time-to-collision

Time-to-collision (TTC) is the first of the SSMs, initially defined in 1972 by Hayward [25]. Hayward(1972) defined TTC as "the time that remains until a crash between two vehicles would have occurred if the crash course and speed difference are maintained." The understanding for TTC is that the lower the TTC, the higher the risk of a collision. TTC is one of the most popular SSMs, with many different methods made to calculate it. This section reviews different methods to calculate TTC and why they are used.

In Hayward's study, it is assumed that the vehicles' speeds and directions are maintained during the calculation of TTC. This method also treats the vehicle as a point mass, measuring off the left front fender. To avoid loss of information about the width of the vehicle, a vehicle width assumption is added. The assumption of using the left front fender as the measurement point and adding vehicle width does cause erratic calculations, a known issue that Hayward mentions as a problem. At the same time, the issue with assuming a maintained speed and direction is that it only gives a snapshot of the possibility of a collision, not factoring in any changes.

A mathematical description of Hayward's TTC is as follows:

$$TTC_i(t) = \frac{(X_{i-1}(t) - X_i) - l_i}{v_i(t) - v_{i-1}(t)} \tag{2.1}$$

where X is the vehicle position with subscript i being the following vehicle, $i-1$ indicating the leading vehicle, l is the vehicle length, and v is the vehicle velocity. The calculation of TTC is done using a recorded trajectory; the method could use, for example, videos of an intersection.

A more geometric method of calculating TTC presented in [2] uses an assumption of the vehicles being points. It assumes their initial speeds and directions are known, and uses the following expressions to calculate the intersection point.

$$x_+ = \frac{(y_2 - y_1) - (x_2 \tan\theta_2 - x_1 \tan\theta_1)}{\tan\theta_1 - \tan\theta_2} \tag{2.2}$$

$$y_+ = \frac{(x_2 - x_1) - (y_2 \cot\theta_2 - y_1 \cot\theta_1)}{\cot\theta_1 - \cot\theta_2} \tag{2.3}$$

Figure 2.2 shows the scenario to calculate the intersection.

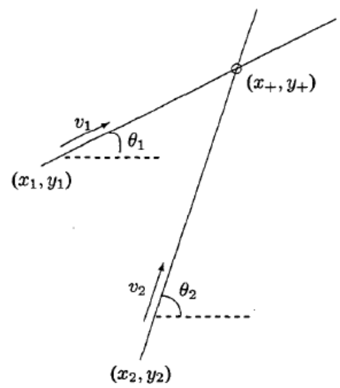

Figure 2.2. Method to calculate intersection [2]

Once the intersection of the two vehicles is found, the time-to-intersection (TTX) is calculated using the equations below.

$$TTX_1 = \frac{|\vec{r}_+ - \vec{r}_1|}{|\vec{v}_1|}\text{sign}((\vec{r}_+ - \vec{r}_1) \cdot \vec{v}_1) \tag{2.4}$$

$$TTX_2 = \frac{|\vec{r}_+ - \vec{r}_2|}{|\vec{v}_2|}\text{sign}((\vec{r}_+ - \vec{r}_2) \cdot \vec{v}_2) \tag{2.5}$$

With: v_1 and v_2 being the velocities of vehicles one and two, $\vec{r_n}$ being the vector representation of coordinate (x_n, y_n), and 'sign' being the sign function. To then find the TTC, there has to be a collision, so when $TTX_1 = TTX_2$, the TTC is equal to TTX. Since all the equations assume the vehicles are a point, a contention criterion is used as shown below.

$$|TTX_1 - TTX_2| < \alpha \tag{2.6}$$

with: α being the contention parameter. Within this contention parameter is where one would take into account the vehicle size and uncertainties on size. The larger α becomes, the more conservative is this algorithm. This method was developed to function with GPS data, thus allowing the calculation in real-time as well as not needing a line of sight, unlike Hayward's method. The method of [2] also detects whether there would be a collision mathematically rather than empirically.

Using a more simplified method of calculating TTC, the work by Jiménez [3] presents a method that reduces collisions down to only ten different configurations. The authors simplify the analysis by characterizing accidents by their initial conditions. Those initial conditions are $\alpha < 90°$ and $\alpha > 90°$, where α is the angle between the motion vectors of the two vehicles. Figure 2.3 shows the two scenarios.

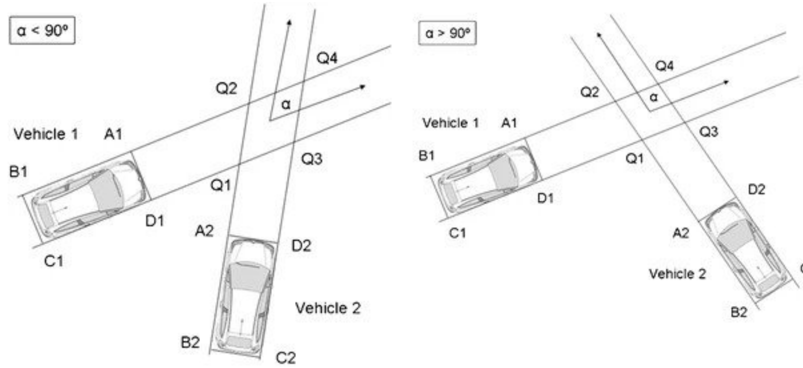

Figure 2.3. Initial conditions [3]

From the initial conditions, the modes wherein accidents can occur are tabulated, including six configurations in the $\alpha < 90°$ situation and four in the $\alpha > 90°$ situation. The TTC is then calculated by matching the condition on the table and finding the tabulated TTC resulting from that configuration. The table shows that the TTC for that condition is equal to when a point reaches the intersection. For example, if the corner of vehicle 2, A2, hits the side of vehicle 1, C1 to D1, then the TTC equals the time it takes A2 to get to Q1.

In a further refinement of the TTC algorithm, [4, 26] proposes different methods to calculate TTC with varying levels of accuracy. In [4], they propose using three different methods to calculate TTC: the circle, the rectangle, and the ellipse-rectangle. These variants are presented to ensure accuracy without significantly increasing computation time. In the circle method, they draw a bounding circle around the vehicle and check for collisions by comparing the radii. The circle method is a quick way to detect if there is a collision but it can create errors, an issue evident by the difference of the areas between the polygons. To increase the accuracy of the TTC calculation, the rectangle model is used instead of the circle method. Here collisions are detected using a simulation method rather than equations. Using the simulation method, any overlaps of rectangles are checked by using a general polygon-clipping algorithm. An issue with this method is that no buffer zone is taken into account, though such a buffer could easily be added by artificially increasing the vehicle dimensions. Lastly, they propose an ellipse-rectangle algorithm, which combines the circle and rectangle algorithms. Their paper introduces a buffer area via this method, which acts as a safety factor, expanding the area one would calculate a collision. They do this because if no buffer zone is added, the vehicle's algorithm would make maneuvers that would not cause a collision but could make the human uncomfortable. For example, Hou's method places an ellipse around the subject vehicle and uses a rectangle on the target vehicle. This method allows the accuracy of the rectangle method but includes the buffer area the circle method provides. To complete their algorithm (as seen in Figure 2.4) for TTC calculation, they use a combo of the circle and ellipse-rectangle algorithm.

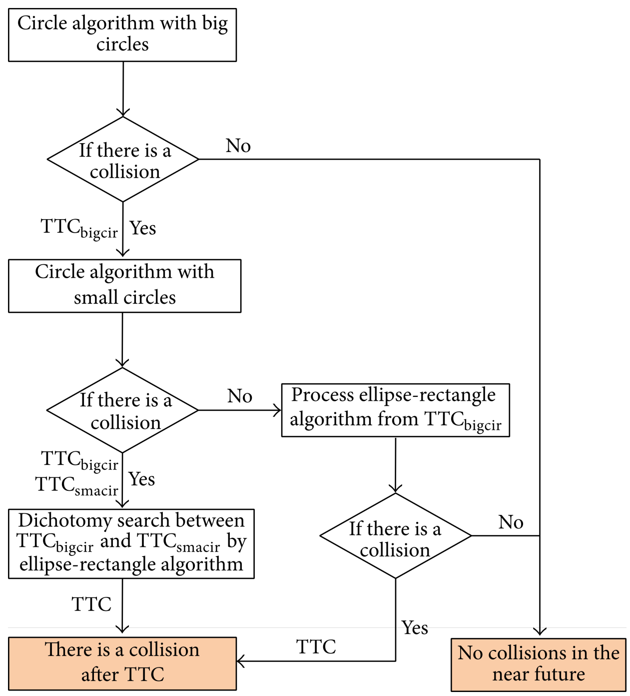

Figure 2.4. Combined algorithm [4]

Using this combined TTC calculation, they found they could reduce the computation time and predict the same TTC result as the longer, more accurate TTC calculation methods.

Lastly, [26] proposes an algorithm similar to [4] that combines faster methods to calculate with higher accuracy. Instead of using the circle method to detect collisions, they use a method similar to [2] that includes a GPS-based model. To increase accuracy, they propose a novel rectangular model that uses ideas from [3].

As shown in Zhou (2020), all these TTC refinements are trying to solve two issues seen with Hayward's first definition [27]. The first issue is that the severity of a conflict is not defined, only whether a collision occurred or not and the second issue is that TTC assumes constant speed, which is not always the situation for moving vehicles. For example, as Nadimi et al.(2019) [28] presents in their work, new indicators other than the original definition of TTC might be better for different collision types but not as a general indicator. Zhou(2020) and Songchitruksa(2004) [27, 29] both argue that these new indicators are data-intensive and require difficult-to-measure derivatives of position. Finally, Zhou(2020) [27] shows no substantial evidence that the modified definitions of TTC outperform the original definition of TTC.

2.2.2 Time-to-collision Variants

One of the variants of TTC that is used in this thesis is time-to-lane departure, also known as time-to-lane crossing (TLC). TLC is defined as the time until the vehicle departs the lane, given its current position, velocity and direction. This measure was first proposed by Godthelp (1984) to determine if there is a connection between driving performance and time-to-lane crossing [5]. In the paper, he uses concepts similar to the TTC measure from Hayward (1972) to calculate the TLC, using the vehicle's lateral lane position, speed, and heading angle. Figure 2.5 shows an example of how Godthelp calculates TLC.

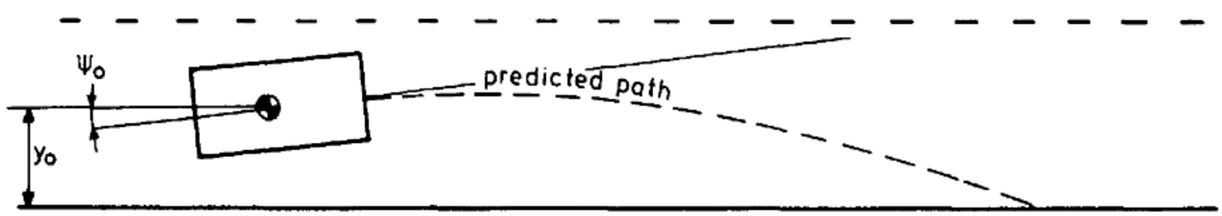

Figure 2.5. Godthelp's way of calculating TLC [5]

For a more accurate calculation of TLC, a trigonometric method is proposed by W. van Winsum (2000) using the fundamental idea that the vehicle never truly drives straight, and thus the trajectory can be represented using a curve that either goes to the left or the right relative to the straight road [6]. From that, the following equation can be derived:

$$TLC = \frac{DLC}{u} \tag{2.7}$$

This equation holds true when u, or the vehicle speed, is greater than 0; otherwise, TLC is undefined. DLC is the distance-to-lane crossing in this equation and it can be calculated with the cosines rule using the following equation:

$$DLC = \alpha R_v \tag{2.8}$$

With α being the angle between the front wheel to the center point of the vehicle trajectory and R_t being the vehicle speed divided by the yaw rate. Figure 2.6 shows how the different lengths relate to each other.

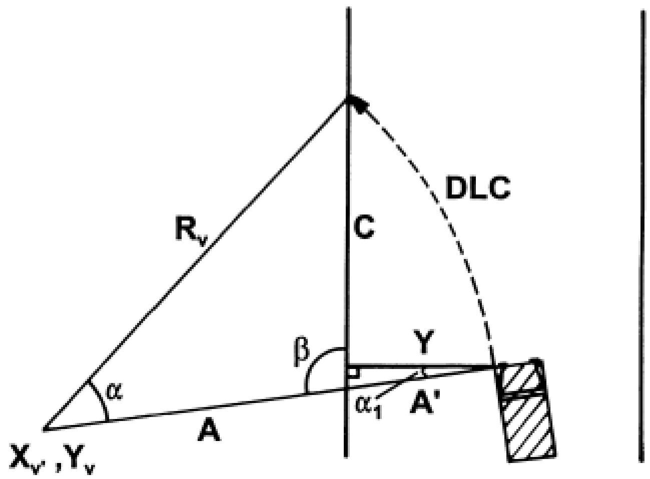

Figure 2.6. Trigonometric length determination on a straight road [6]

The above method is a straightforward way to calculate TLC, but in real situations, the lengths needed within this calculation are not easily available. To address this, van Winsum (2000) [6] proposed an approximation of TLC. The approximation simplifies the calculation to the following equation:

$$TLC = \frac{Y}{Y'} \tag{2.9}$$

where Y is the lateral distance between the front wheel and the lane, and Y' is the lateral velocity.

2.2.3 Relative Lane Position

The relative lane position is defined as the vehicle's location with respect to the center line and the lane edge. It is used in the calculation of TLC, and it has safety implications because relative lane position can provide information about how well a driver maintains the vehicle within the lane or how close a vehicle is to encroaching on another vehicle in an adjacent lane. In general, the further away a vehicle is from the center line, the higher the risk of a collision with objects outside the lane.

TLC and, consequently, relative lane position have been shown to be surrogate measures of how drivers create a safety margin. For example, Godthelp in 1984 [5] shows that drivers generally take corrective actions at a particular TLC regardless of vehicle speed. This was established further in his later work (1996) [30], where it was shown that the TLC minima would remain constant over different curve radii. This finding suggests that drivers are acutely aware of their relative lane position and, generally, regulate this closely throughout different trajectories.

2.2.4 Post-encroachment Time

Post-encroachment time (PET) is defined as the time from when a vehicle leaves a conflict area to the time when another vehicle enters that location [31]. More specifically, PET is measured from the geometric extreme of one vehicle leaving the area to the geometric extreme of the second vehicle entering it. PET can be more generally described as the amount of time that two vehicles miss each other when passing through the same physical area. An example is shown in Figure 2.7, where one vehicle (B) is leaving an intersection area (shown in red) traveling from left to right, and another vehicle (A) is entering the intersection traveling from bottom to top. In this case, the PET is given by $t_2 - t_1$ from the times indicated on the figure.

Geometrically, the lower the PET, the closer the vehicles are to impacting each other, implying a higher risk of a collision. PET was first defined by [31], where he divides gap time into PET and encroachment time (ET). His study [31] found that PET is the most promising indicator of safety in intersections versus the other surrogate safety measures of ET and gap time due to its ease of measurement. Moreover, according to [32], it is shown that for the situation of crossing conflict areas (intersections), PET is better than TTC as an SSM. However, [31] does note some shortcomings of PET; specifically, it does not describe the actions taken by drivers or the initial situation.

2.3 Deceleration-based Measures

2.3.1 Deceleration Rate to Avoid Crash

The deceleration rate to avoid crash (DRAC) is defined as the minimum required deceleration rate a vehicle must initiate at its present position, orientation, and velocity to avoid a crash if the paths and the speeds are maintained. DRAC was first proposed in 1976 by [33] to explain the severity of a collision. Unlike the previously stated SSMs, DRAC is intended to include driver responses to a situation, assuming that deceleration is the first and only action drivers take to evade a collision [22]. To calculate DRAC, an assumption is made that only one vehicle is taking evasive actions while at the same

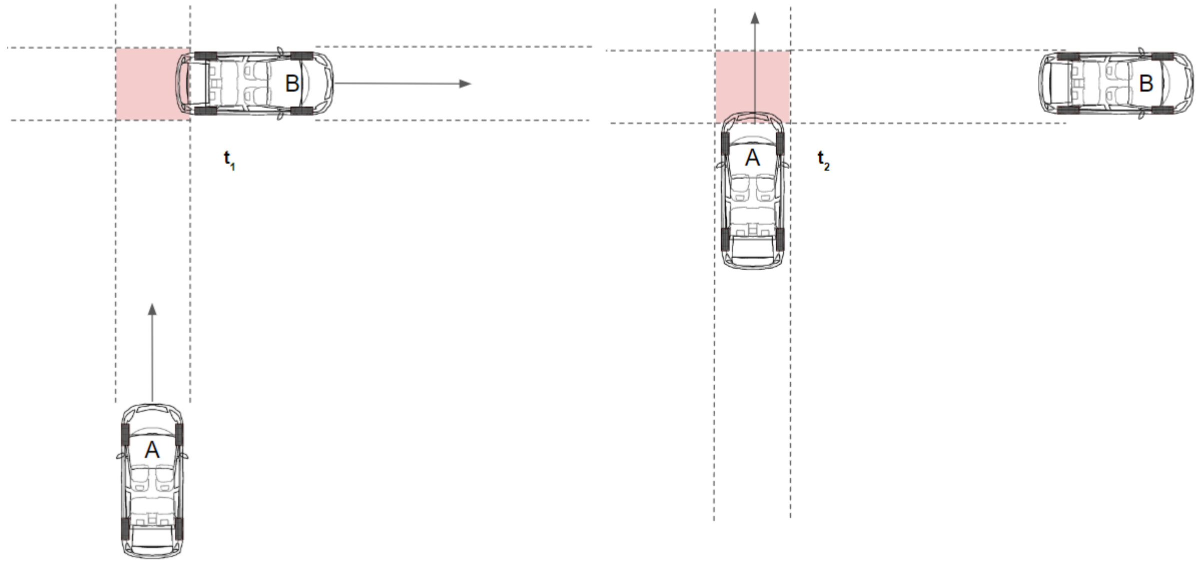

Figure 2.7. Showing PET (t_2-t_1), where vehicle A is approaching where vehicle B was

time the other vehicle is maintaining its speed and direction. Once those assumptions are made, equation 2.10 can be used to calculate DRAC:

$$DRAC_{i,t+1} = \frac{(V_{i,t} - V_{i-2,t})^2}{2[(P_{i-1,t} - P_{i,t}) - L_{i-1}]} \tag{2.10}$$

where the variables are defined as follows: t is the time with (t being the current time and $t+1$ being one time step in the future); P is the position of a vehicle (i = following vehicle (one decelerating), $i-1$ = leading vehicle); L is the length of the leading vehicle l; V is the velocity (i = following vehicle, $i-1$ = leading vehicle). The equation is derived from Newton's equations of motion, with an extra length considered to account for vehicle length. The understanding is that the higher the DRAC value, the higher the chance of a crash occurring. It is implied that a crash will occur if the DRAC exceeds the maximum available deceleration rate.

2.4 Conflict Severity Measures

2.4.1 Speed Differential

The speed differential is calculated by taking the difference in speed between a vehicle and a conflicting entity at the time of a collision. This measure is not an indicator of an impending crash, but rather an indicator of the severity of a crash at the moment of impact. A large body of works shows that collision severity's effects depend on the relative velocity (speed differential) and the vehicles' mass or the properties of kinetic energy [34–37].

2.4.2 Conflict Severity Index

It is possible that other SSMs may have values that seem safe or unsafe due to numeric values, but the situation itself is not indicated by the measures alone. For example, a low-speed vehicle bumping against an obstacle while parking has very different safety implications than the same vehicle bumping up against another moving vehicle nearby at highway speeds, but with precisely the same geometric situation. The conflict severity measure tries to measures the severity of a conflict using a combination of other SSMs. The severity of a conflict is a measure of how likely a collision/conflict occurs and how severe that collision/conflict would be.

In the literature, the conflict severity index is calculated in many ways; one of the first is the Swedish Traffic Conflict Technique (TCT), developed by Hyden in 1987. [22] The Swedish TCT is given as a ratio of time-to-accident (TA) to the conflicting speed (CS). TA is the time that remains to an accident from the moment that one of the road users starts an evasive action if they had continued with unchanged speeds and directions." TA can be understood as how close the accident was to happen. While CS "is the speed of the road user taking evasive action, for whom the TA value is estimated, at the moment just before the start of the evasive action. The CS can be described as how serious the collision would be had it occurred. These ratios can then be graphed, see Figure 2.8, and lines of increasing severity can be graphed, with any severity above 26 considered severe.

Another method developed is called the DOCTOR method [38]. The conflict severity is scored from 1 (least severe) to 5 (most severe). How this score is calculated is a two-part equation. The first part is a subjective measure of how severe the conflict is, and the second part is the probability of the collision and potential injury severity is measured. The second measure is calculated using TTC and PET [38].

Lastly, a conflict index (CI) proposed by Alhajyaseen (2015) blends PET with the speed and masses of the vehicles with the following equation [8]:

$$CI = \frac{\alpha \Delta K_e}{e^{\beta * PET}} \tag{2.11}$$

where the variables are defined as *alpha* being a percentage of the released energy that

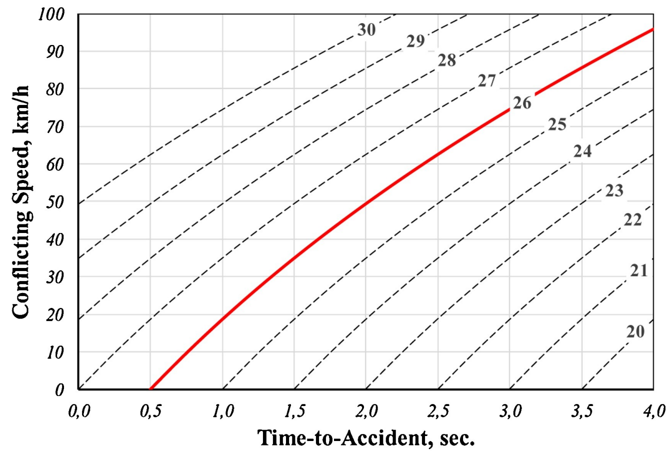

Figure 2.8. Classifying conflict severity, adopted from [7]

will affect the vehicle's occupant(s); ΔK_e being the change in total kinetic energy before and after the crash; β being a weight from 0-1 representing the probability of a crash occurring; PET being post-encroachment-time. The purpose of CI is to create a measure that encompasses the probability and severity of a crash. It measures the probability by including PET while the severity of the crash is described by the kinetic energy released by the crash.

2.5 Spacetime

There are relatively few references that use spacetime to calculate SSMs, but some examples exist. A key reference are the definitions given in the Federal Highway Administration (FHWA) report on "Surrogate Safety Measures from traffic simulation models" [39]. In this report, they define the SSMs in a graph with "distance from the conflict point" on the y-axis and time on the x-axis. This method is similar to the one proposed in this thesis but does not include the third axis, so valuable information is removed.

References exist wherein spacetime maps have been used to help solve the problem of path planning and more specifically the problem of path planning when multiple units are moving. [40] proposes the use of a spacetime map to create a reserved table that other units then avoid. These reserved tables are created when each unit path finds its way to its goal destination in spacetime and marks it. This paper uses a similar idea as to what is proposed in this thesis, but is used for a different purpose.

Chapter 3 Methods

3.1 Introduction

This chapter introduces the representations related to spacetime for their use in SSM calculations. It begins with how vehicles and objects are defined, then it describes the scenarios that are used for testing the motion of vehicles. Next, it explains ray casting in spacetime to determine measure values. Lastly, the chapter explains how and why the objects are dilated.

3.2 Spacetime Set Up

As shown in the literature review, the calculation of SSMs can be complex. To simplify, this thesis proposes representing vehicles and objects in a spacetime map. By utilizing a spacetime map, the representation of SSMs becomes simplified as distances along an axis with corresponding slopes. Detailed explanations regarding this concept are provided later in Chapter 4.

In this thesis, the spacetime map is defined as a coordinate system where space is represented by a two-dimensional plane (the x-y plane), while time is represented by the z-axis. Illustrated in Figure 3.1.

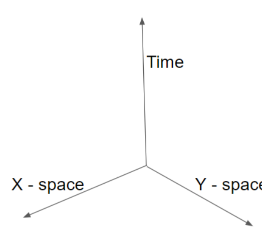

Figure 3.1. Showing the axes in spacetime

With time being depicted along the z-axis, the movement of a vehicle traveling at a constant speed in a straight line is represented as a diagonal line within the spatial dimensions. A visual example of such representation can be observed in Figure 3.2, with an isometric view in Figure 3.3. The angle of the trajectory is a key indicator of the vehicle's speed. A steeper, more vertical trajectory suggests slower movement, while a flatter, more horizontal trajectory indicates faster movement.

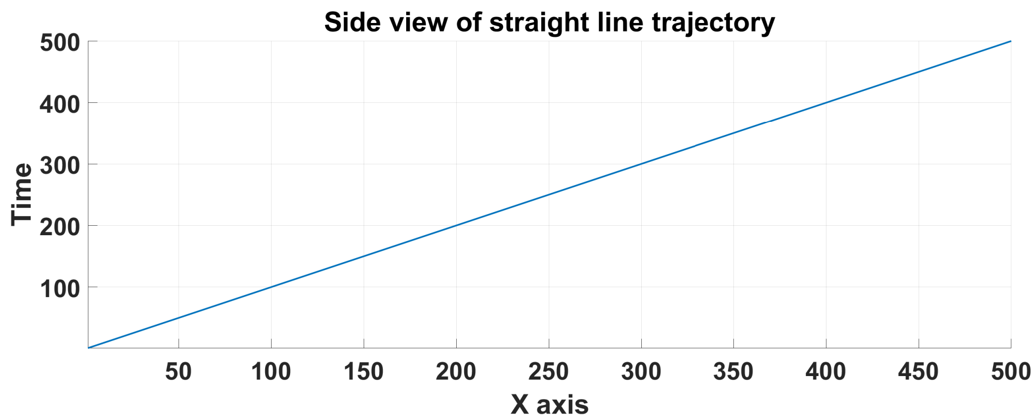

Figure 3.2. Side view of the straight-line trajectory

3.2.1 Vehicle Set Up

A 2D representation of a standard vehicle is used to represent the vehicle in space. This 2D representation can be seen as the red rectangle bounding the vehicle. See Figure 3.4 for reference. The vehicle properties are as follows: width is 1.82 m, length is 4.71 m, the length from the center of gravity (CG) to the front axle (A) is 1.42 m, the length from the CG to the rear axle (B) is 1.42 m, the length from the CG to the front bumper (Lf) is 2.22 m, the length from the CG to the rear bumper (Lr) is 2.47 m. These measurements are from a BMW 3 series.

3.2.2 Trajectories Analyzed

To simplify the discussion of a generalized method to analyze SSMs, this thesis considers only primary motions of vehicles as one of five different base trajectories. These five

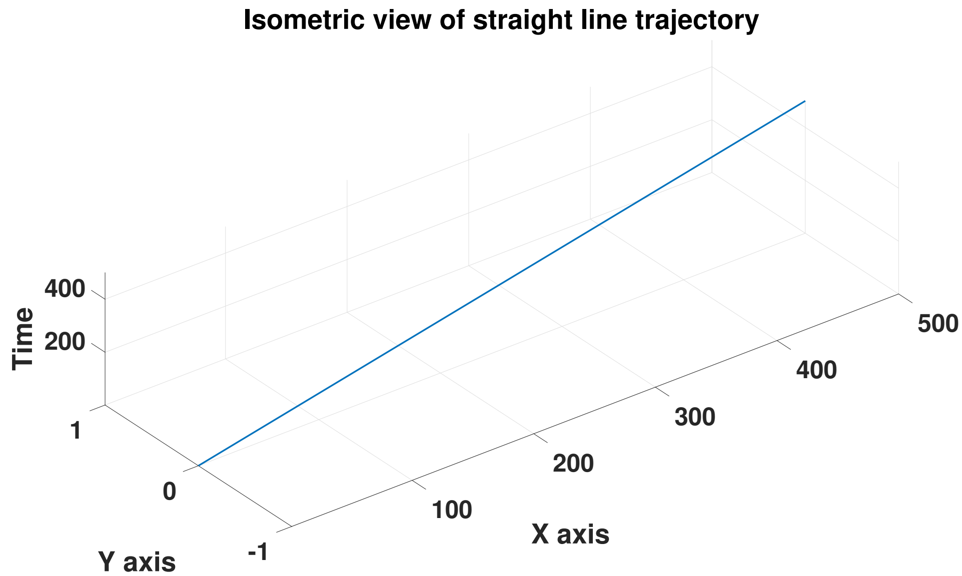

Figure 3.3. Isometric view of the straight-line trajectory

base trajectories are used because they can build more complex trajectories, yet are also individually unique. The five trajectories are a lane change, a half-lane change, stopping at a stop sign, a right turn, and a left turn. These example trajectories are depicted in Figure 3.6.

Within the code implementation, each trajectory is represented by a data structure consisting of time, x-position, y-position, and the yaw angle of the vehicle. The yaw angle is calculated according to the ISO (International Organization for Standardization) conventions, illustrated in 3.5.

It is important to note that since plotting is done in spacetime, the entries into the time data structure play a significant role in determining the trajectory. As illustrated in Figure 3.6, when the vehicle maintains a constant speed, it generates a diagonal line. Conversely, if the vehicle comes to a complete stop, one of the base trajectories would exhibit a distinct "flipped z" shape.

3.3 Plotting Vehicles in Spacetime

Taking the base trajectories discussed above, vehicle plotting can be performed. For an example, see Figure 3.7 for the half-lane change.

The vehicle can be seen in the red outline, while the blue is the vehicle's shadow. The shadow is used to help the viewer see the trajectory in more traditional XY coordinates.

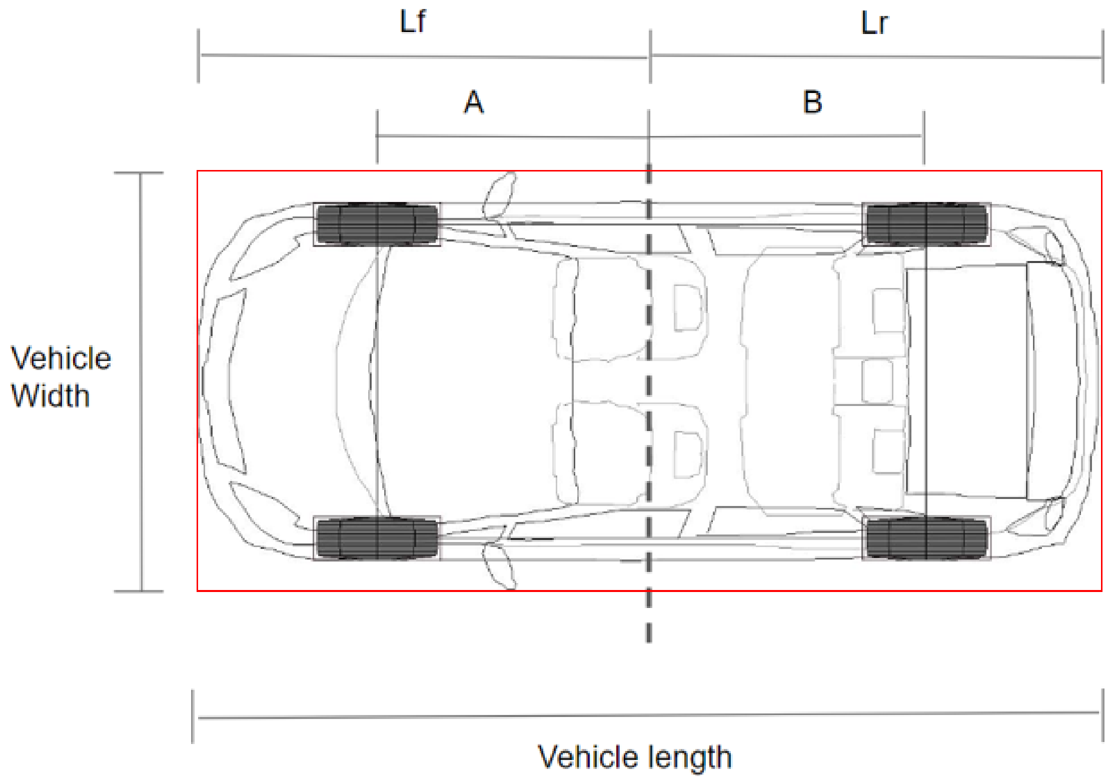

Figure 3.4. Vehicle Parameters

For an example of a change in velocity, Figure 3.8 shows stopping at a stop sign. It can be seen that when the vehicle is stopped, it makes a pillar.

It is important to note that all the spacetime plots do not have axes that are scaled equally. Instead, the Z and the X axes are usually scaled so that the spacetime plots are easy to visualize. There are some issues that occur when performing this scaling, such as the vehicles becoming compressed causing them to appear as compressed rectangles.

3.3.1 Adding Objects to the Simulation

Objects in this thesis are represented as 3D mesh structures, as they facilitate ray casting, which is how collisions are being detected in this thesis. Ray casting involves projecting a ray in a defined direction and then identifying any intersections it encounters. The aforementioned 3D structures are comprised of triangles connected with vertices. To construct the mesh representation, slices of the object (with each slice corresponding to a specific time step) are needed. These slices are 2D outlines of the object at each time step. For example, if the object remains stationary in space, the slices of the object would align vertically resembling a pillar-like structure, illustrated in Figure 3.9.

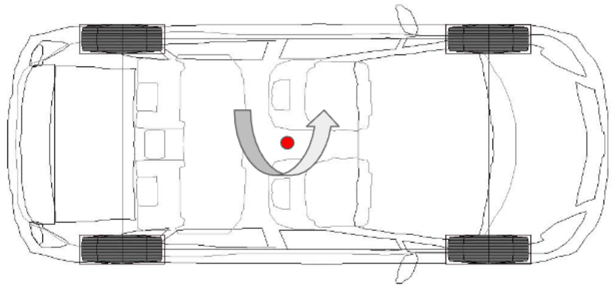

Figure 3.5. Shows positive yaw is counter clockwise (ISO)

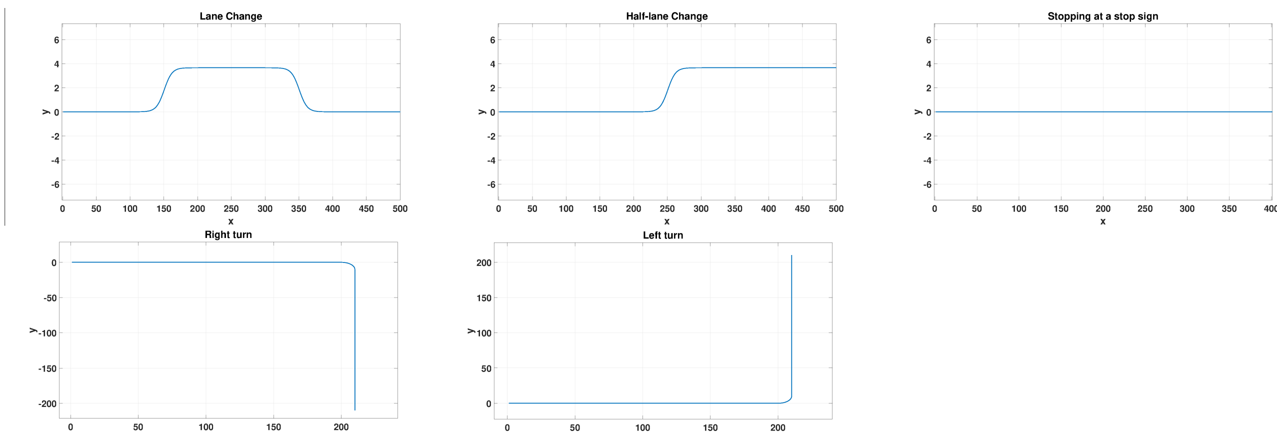

Figure 3.6. Vehicle trajectories

This stack of 2D outlines would then be connected using triangles to form faces. Figure 3.7 provides an overall depiction of this process, while Figure 3.10 offers a closer view of the resulting 3D objects.

It is important to note that if the object remains static in space without undergoing any changes in shape (object dilation), the simulation can be expedited by only considering the first and final time slices of the object. By focusing solely on these two slices, unnecessary computations are avoided, resulting in significant computational savings. In other words, processing time is reduced if object meshes only use time slices when the objects are moving.

Consider a scenario where the object remains stationary during the first half of the simulation but begins moving at a constant velocity for the remaining half. In this case, only the initial slice and the final slice from the first half are required, along with every

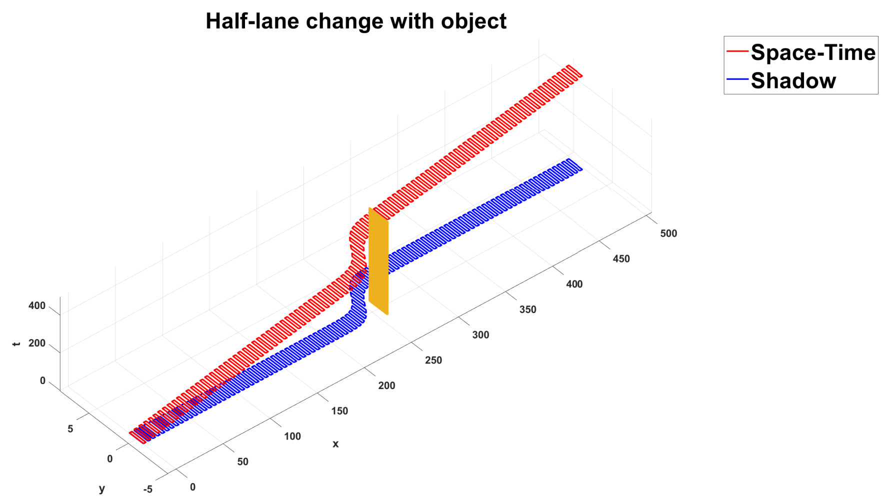

Figure 3.7. Half-lane change with object

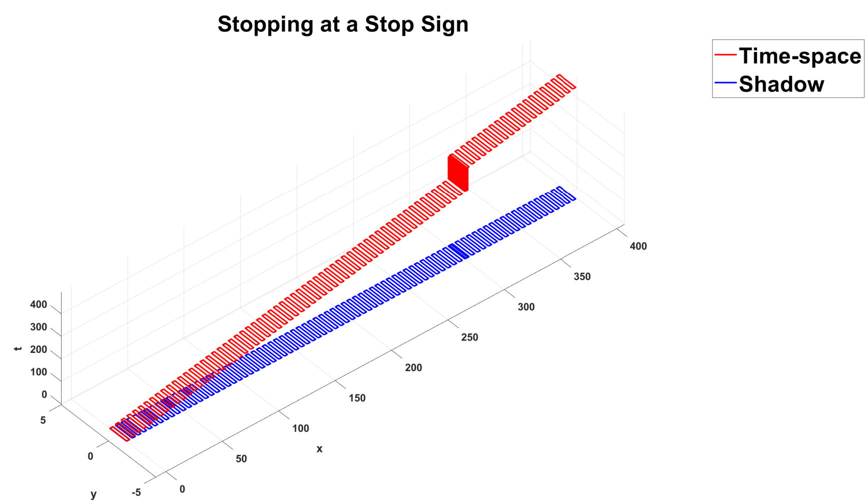

Figure 3.8. Stopping at a Stop Sign

slice from the second half. This principle also applies when the object undergoes changes in shape; only the first and last slices of time slices should be used for objects that are static.

3.3.2 Object Dilation

Since ray casting is used to determine if a collision occurs, some situations arise where a collision would occur but it is not registered by this point representation . For example, as illustrated in Figure 3.11 the ray cast does not detect the object, despite a collision

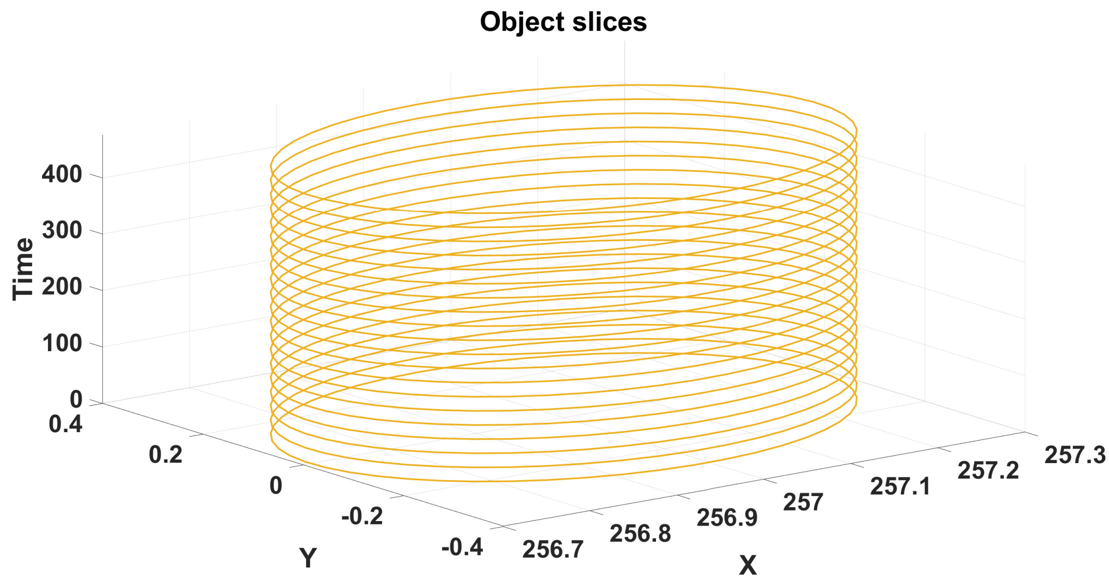

Figure 3.9. An image showing the stacking up of object slices

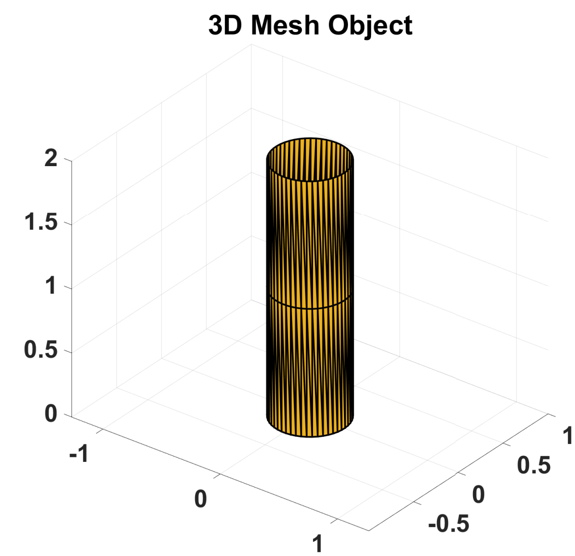

Figure 3.10. A zoomed-in image of 3D mesh objects

being imminent.

Another situation where ray casting misrepresents a collision is when the object is smaller than the envelope of the vehicle and doesn't intersect the ray cast. To address these problems, this thesis employs Minkowski sums to dilate objects. By applying Minkowski sums, the objects are dilated or expanded, effectively enlarging their boundaries to account for the vehicle width. A Minkowski sum combines two polygons to

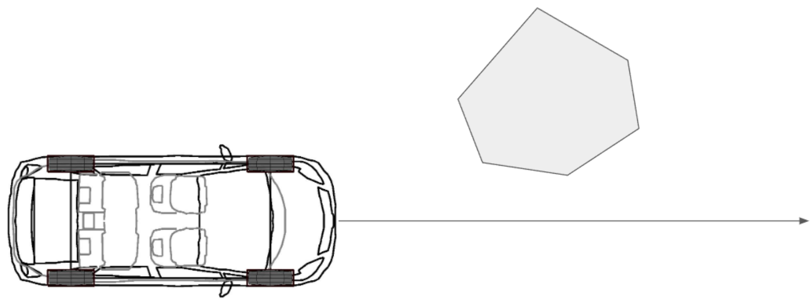

Figure 3.11. Example of when the ray casting does not detect a collision

create a larger polygon considering the boundaries' extremes. The resulting polygon allows for more accurate collision detection as it accounts for the true size of the vehicle. Figure 3.12 illustrates the concept of a Minkowski sum, in which the yellow triangle is the object, the blue rectangle is the vehicle and the red polygon is the new envelope.

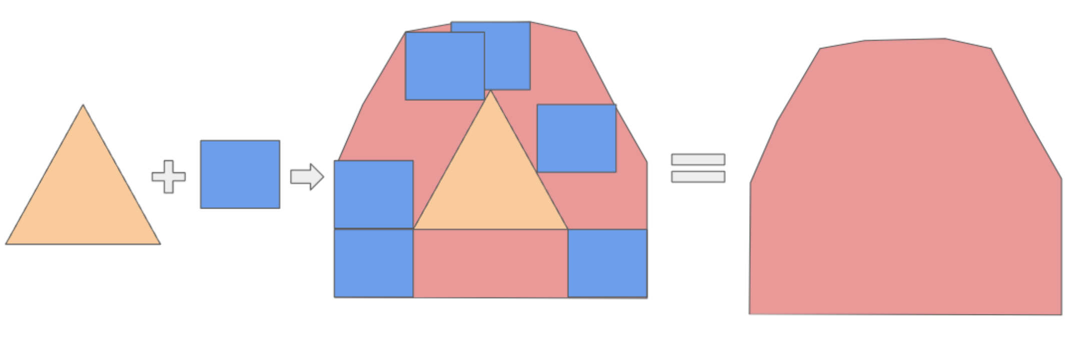

Figure 3.12. Example of a dilated object being created

Dilation is done usually along all dimensions, but to save on computation time, this thesis only performs dilation along only one axis - the axis perpendicular with the direction of travel or the vehicle width. Dilating along the axis parallel (vehicle length) with the direction of travel does not help in registering a collision, as illustrated in Figure 3.13.

The new dilated envelope replaces the old obstacle/object representation. By replacing the original obstacle/object with the Minkowski sum dilation, it is now guaranteed that the vehicle does not misrepresent a collision; this can be seen in Figure 3.14. The vehicle registers the newly dilated object and detects an imminent collision.

Dilation is done to an object at every unique time step, which occurs when there is a change in the vehicle's yaw angle. This follows the framework set in the "Adding Objects to the Simulation" section. It is done for similar reasons as focusing on solely unique time steps, thereby significantly reducing computation time. In this thesis, the inclusion

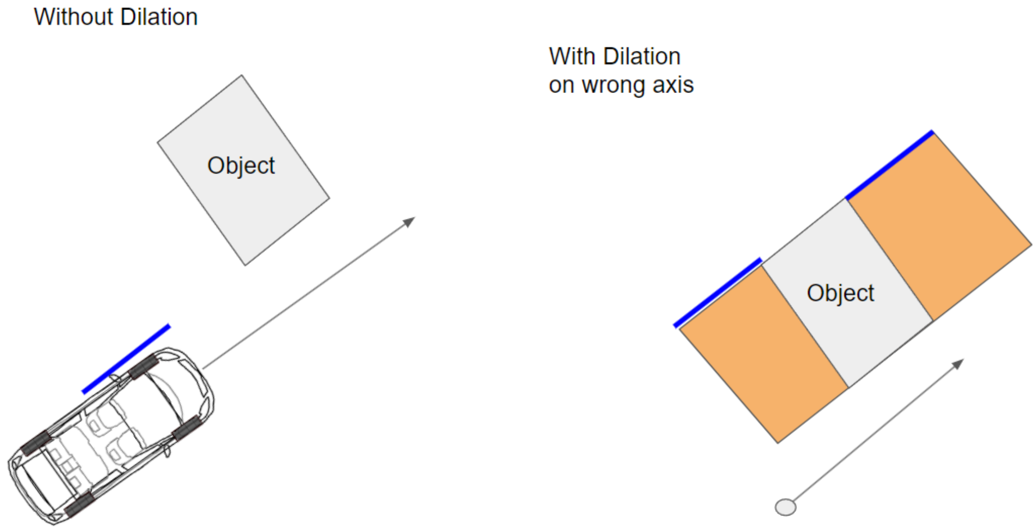

Figure 3.13. Example of dilating an object with the wrong axis

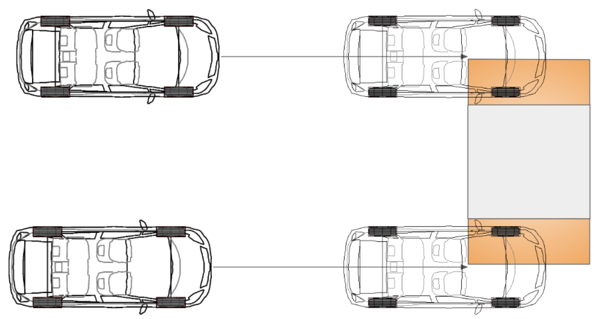

Figure 3.14. Example of how a dilated object prevents the vehicle from colliding

of unique time steps involves performing dilation along the perpendicular direction to the vehicle's direction of travel. To maintain perpendicularity, the object needs to rotate with the vehicle.

Figure 3.15 provides an example of the object's appearance after the dilation process has been applied. The figure offers a closer view of the areas with unique time steps, illustrating a change in yaw angle. This zoomed-in image also highlights how only unique slices are used, as where the yaw angle changes the mesh is dense and outside it is sparse.

3.3.3 Adding Lanes to the Simulation

In order to calculate TLC, it is necessary to incorporate lanes into the simulation. Lanes are defined using the FHWA (Federal Highway Administration) guidelines, where the standard width between lanes is 12 ft [41]. Figure 3.16 illustrates the depiction of lanes

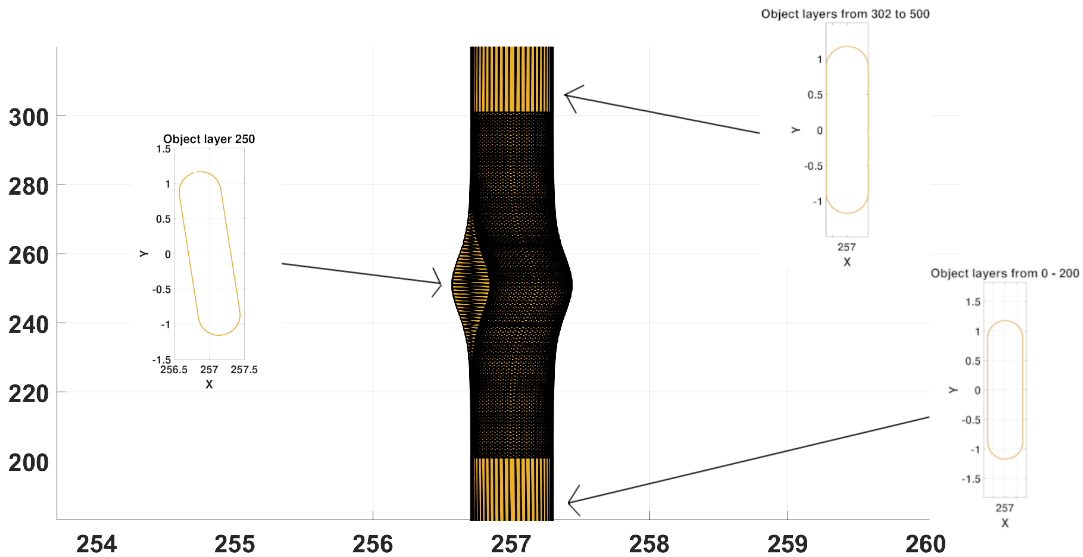

Figure 3.15. An example of a dilated object using only unique time steps

and their definition within the simulation. It is important to note that these definitions are the centers of lane markers.

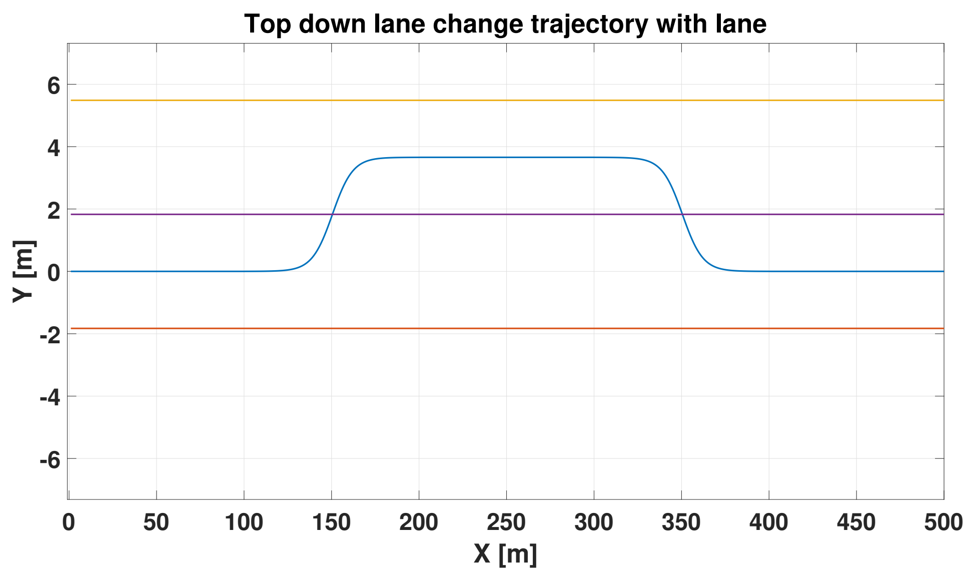

Figure 3.16. Showing how lanes are plotted in 2D, with the blue line being the vehicle trajectory and the other colors being the lanes

To facilitate ray casting in order to calculate TLC, these center lines need to be converted into 3D meshes. This conversion involves creating polygons that represent the lane markers on the road. The width of these markers is defined as 6 in or 152.4 mm according to the FHWA guidelines [42]. To create these polygons, the center line is transposed along the y axis, half the width of a marker (3 in) in one direction and again by the same amount in the other direction. This process results in the creation of a polygon, as depicted in the top-left portion of Figure 3.17.

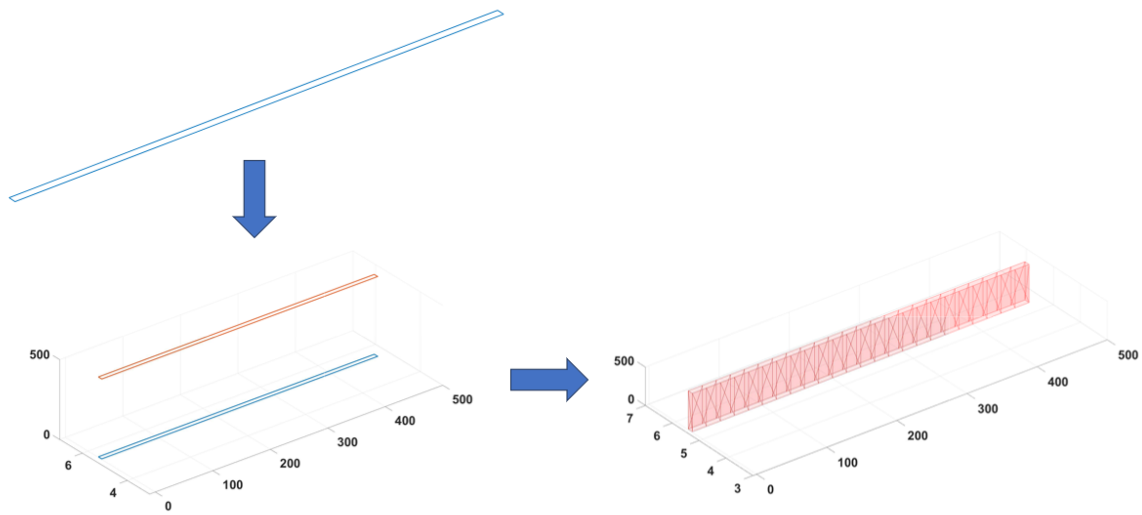

Figure 3.17. Displaying the process to create 3D meshed lanes

Using this newly created polygon, the same process used for objects can be done here; placing one polygon at $z = 0$ and one at $z =$ max height and then converting it to a mesh. These processes can be seen in figure 3.17, as the bottom two images. Once this mesh has been created, Figure 3.18 shows how these meshes are plotted in the simulation.

Figure 3.19 illustrates what mesh lanes on a curved road resemble.

3.4 Ray Casting

In order to calculate SSMs, it is necessary to detect collisions. To achieve this, the technique of ray casting is employed. Ray casting involves casting rays from a specific starting point in a given direction to identify potential collisions. To perform ray casting effectively, this thesis uses two key inputs to define the ray: the direction of the rays and the starting point for each ray. This method is used instead of other approaches for defining a ray as those alternative methods are not applicable in the context of collision detection where knowing the location of an intersection is crucial. Figure 3.20 illustrates ray casting, with blue polygons being a 3D mesh, the red 'X' representing the

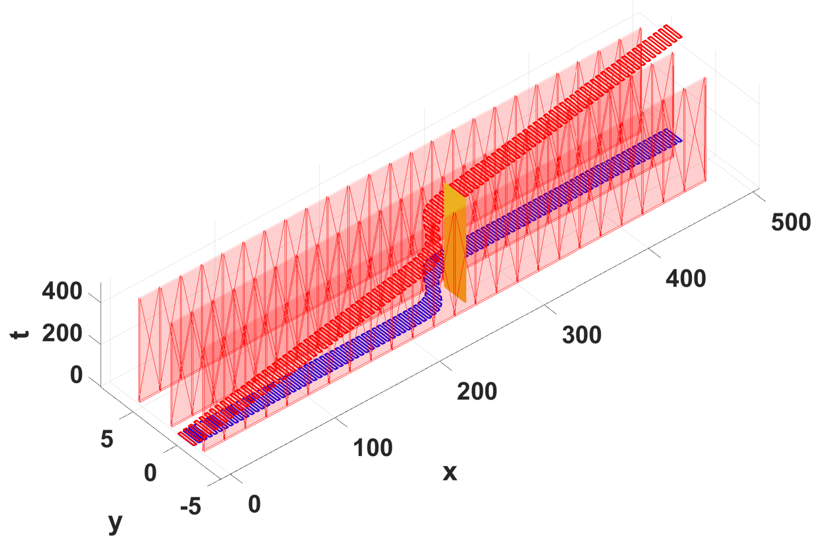

Figure 3.18. Showing how lanes are plotted in 3D

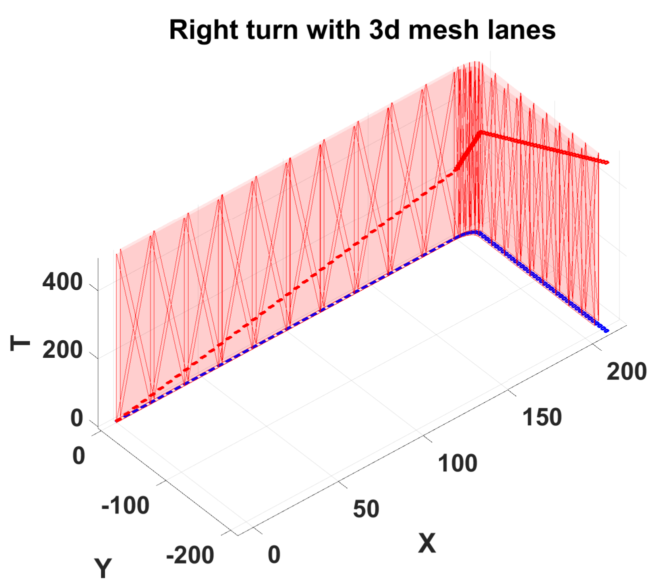

Figure 3.19. Showing how curved lanes are plotted in 3D

intersection point, the blue 'X' the origin of the ray, the black ray being the ray cast, and the blue ray being the unit vector.

To determine the direction of the ray cast, a unit vector is obtained from the vehicle's trajectory using body fixed coordinates. The unit vector represents the orientation and heading of the vehicle, thus the vector points in the direction of travel. The unit vector needs to be calculated for every point along the vehicle's trajectory to take into

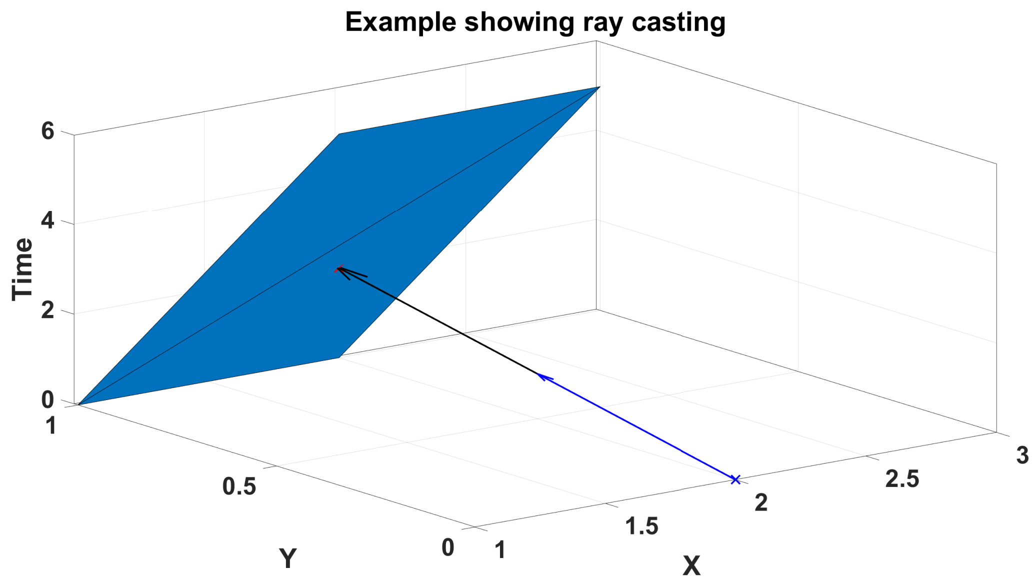

Figure 3.20. An example of ray casting towards a wall

account all the vehicle's movements. This ensures that no movements are unaccounted for, thereby improving the accuracy of collision detection. Figure 3.21 demonstrates what all these unit vectors look like when plotted.

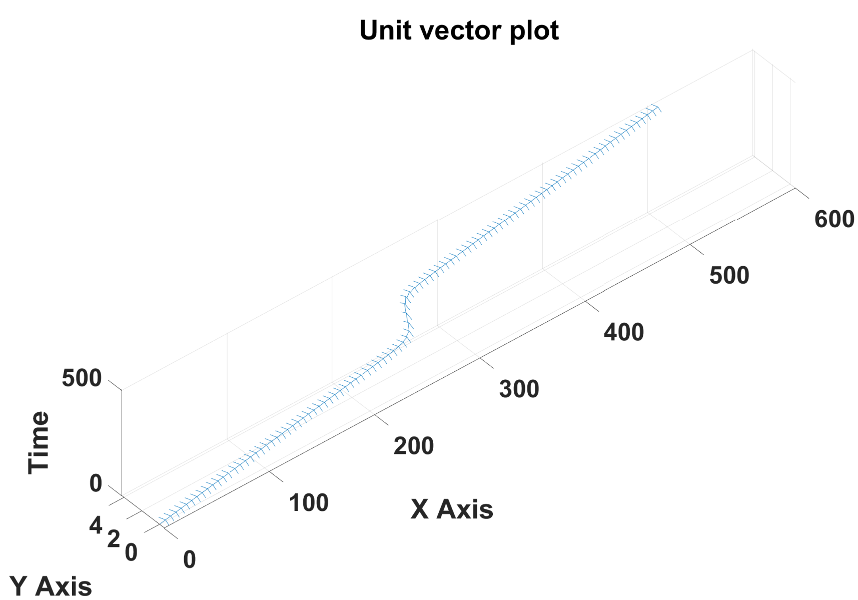

Figure 3.21. How the Unit vectors at each time step look

Once the unit vectors are determined, these vectors are used to determine the direction of the ray cast, with the starting point being the center of the rear axle on the vehicle. An algorithm developed by Möller and Trumbore [43] and implemented by Tuszynski [44] is used to perform the actual intersection detection between the object and the ray cast. This algorithm implementation indicates if the ray intersected the object, the distance to that intersection point, the barycentric coordinates of the intersection point, and the Cartesian coordinates of the intersection point.

This algorithm is used at every time step to determine if a collision occurs if the vehicle continues its path. If there is a collision, a line can be drawn from the vehicle position to the point of collision. This can be seen in Figure 3.22. The ray casting framework is used to not only detect collision with object, but also detect if a vehicle was previously in the same location and if the current vehicle crosses a lane. This is done by casting separate rays in the negative z-direction to determine if a vehicle was previously in the same location, or similar to objects, in direction of travel but checking instead for lanes.

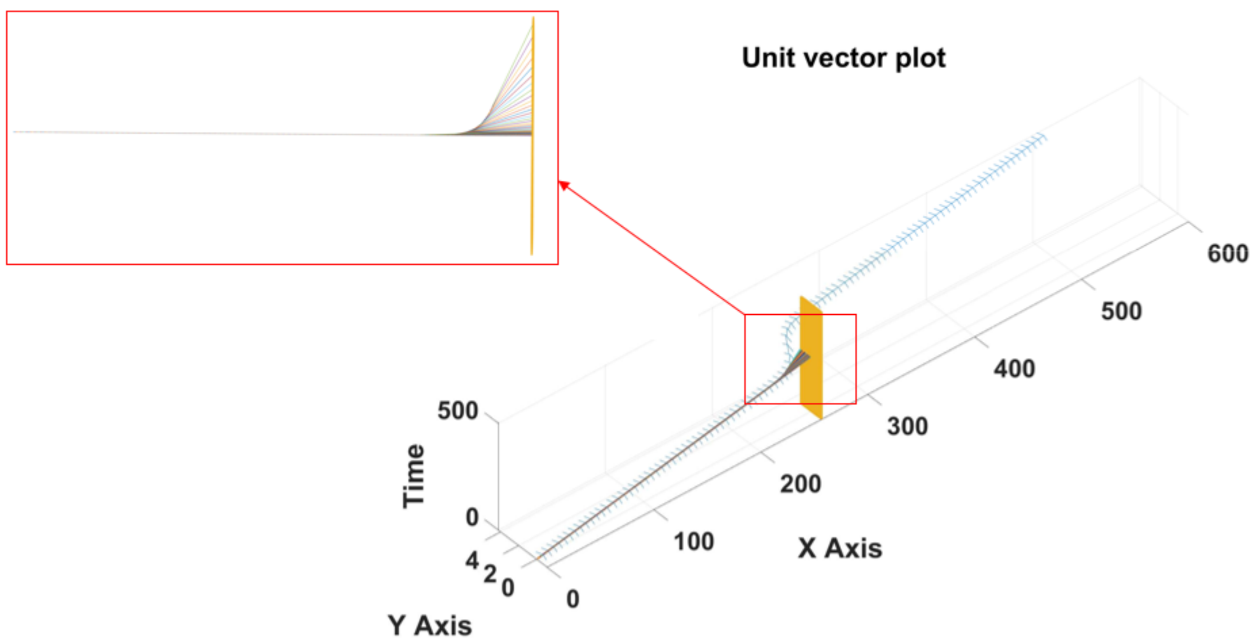

Figure 3.22. Lines drawn from vehicle to intersection point, with a zoom in around the intersection point

Chapter 4
Results

4.1 Introduction

The chapter presents each of the selected SSMs: TTC, TLC, PET, DRAC, SD, CSI, and show how each SSM can be calculated in spacetime using ray projections. It uses the framework developed in Chapter 3 and uses knowledge gained in Chapter 2 to calculate and show each SSM in spacetime.

4.2 Time-to-collision

As discussed in Chapter 2, TTC is the time to a collision if a vehicle were to continue its path and velocity. To calculate this measure in spacetime, detecting if there would be a collision is the first condition to be satisfied. This is done using ray casting, as discussed in Chapter 3 and illustrated in Figure 4.1.

If an intersection can be determined, the point of intersection is returned. Using this intersection point and the origin of the ray, the slope can be calculated, which represents the inverse velocity of the vehicle. The horizontal distance derived from this slope calculation corresponds to the spatial extent that the vehicle must traverse to reach the collision point with the object while the vertical distance is the time that it would take to reach the collision point, or TTC.

To prove that in spacetime the TTC is the vertical change between the intersection point and the point on the vehicle, a mathematical analysis can be performed. First, using the ray cast as the line to analyze, the slope of this line is time divided by space, shown in equation 4.1

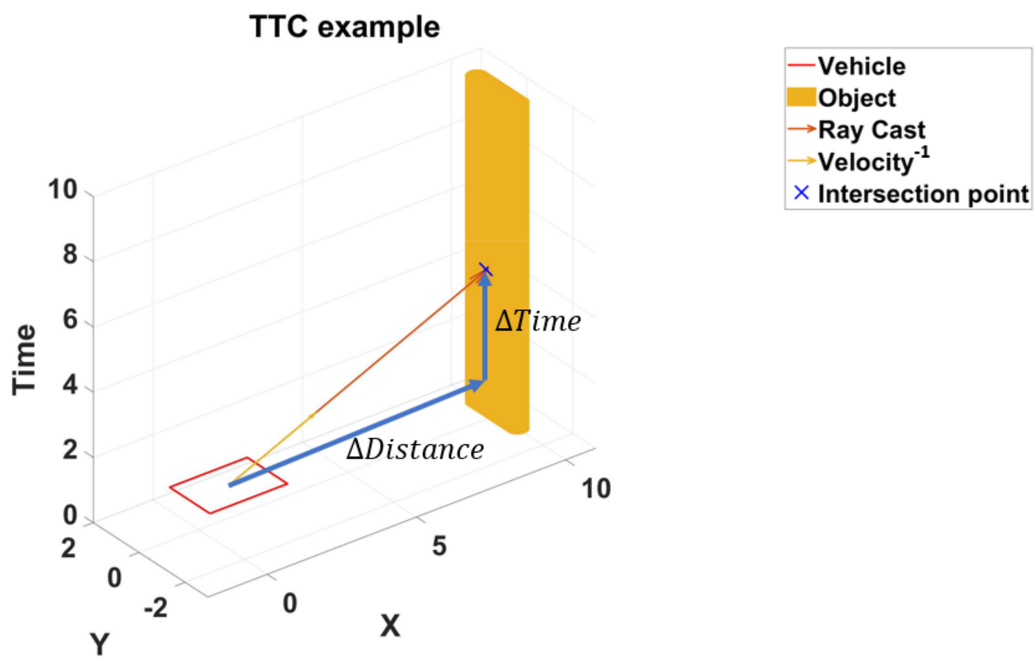

Figure 4.1. Showing TTC ray casts in 3D

$$Slope = \frac{\Delta Time}{\Delta Distance} \tag{4.1}$$

with Hayward's definition of TTC being the distance to collision divided by the velocity, shown in equation 4.2:

$$TTC = \frac{\Delta Distance}{\Delta Velocity} \tag{4.2}$$

the slope can be substituted into the TTC equation for velocity. Doing this reduces the TTC equation down to just time, or the vertical height between the intersection point and the vehicle location. This process is illustrated in figure 4.1

The TTC calculation can be performed every time step but only returns values if there is a predicted collision, illustrated in Figure 4.2. Performing this calculation using the half-lane trajectory, ever decreasing TTC values are obtained. These values can then be plotted for better comprehension, as seen in Figure 4.3. Given that the vehicle in this thesis has a constant velocity, the TTC values exhibit a linear decrease as the vehicle approaches the object. These values follow the understanding for TTC, as the closer the vehicle gets to an object the smaller the TTC values become.

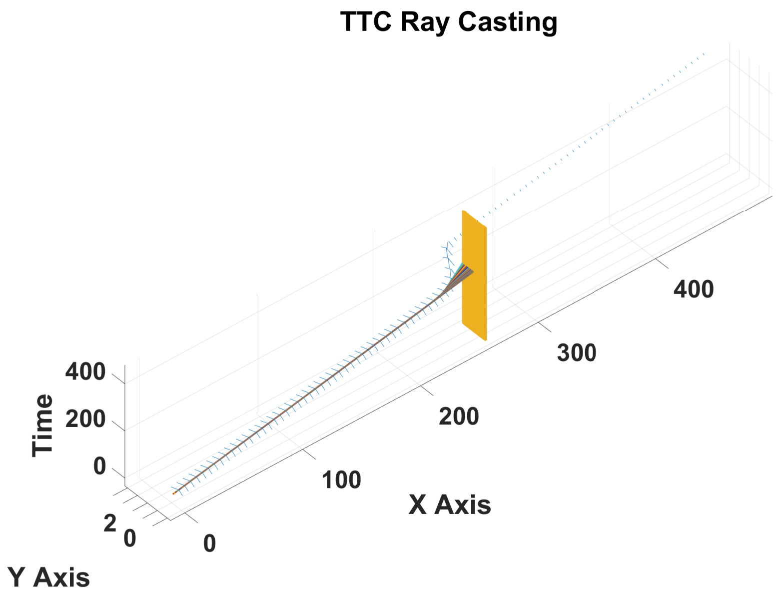

Figure 4.2. TTC for the simulation, showing ray casts for each time step toward an object

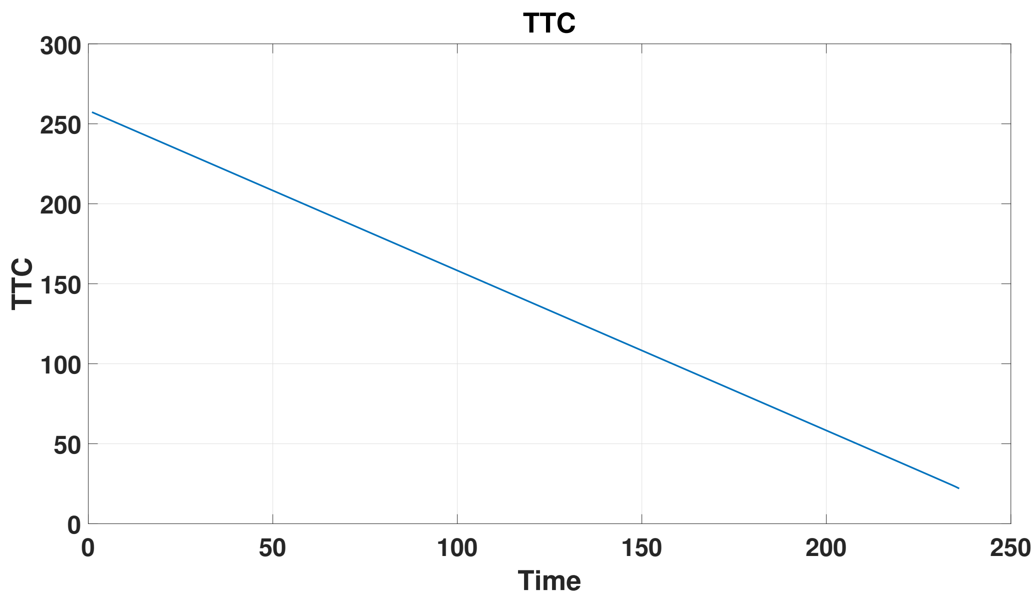

Figure 4.3. TTC vs timestep

4.2.1 Time-to-lane Departure

Time-to-lane departure, or time-to-lane crossing (TLC), is very similar to TTC. The calculation is the same except that the ray cast is now detecting collisions with the lanes. Similarly, the vertical distance between the intersection point and the origin of the ray, the rear axle, is the time-to-lane departure. This value goes to zero as the vehicle gets closer to a lane. Once the value reaches zero, the algorithm searches for another lane. This searching is performed by ray casting to the other lanes and checking what the value returned; if the value is greater than zero and not the previous lane, that value is used.

Figure 4.4 shows how these ray casts are plotted; the blue is for the further lane while the green are ray casts to the middle lane. The red transparent objects are the lanes in their 3D mesh form. A better view is shown in Figure 4.4 which illustrates a top-down view of the ray casts.

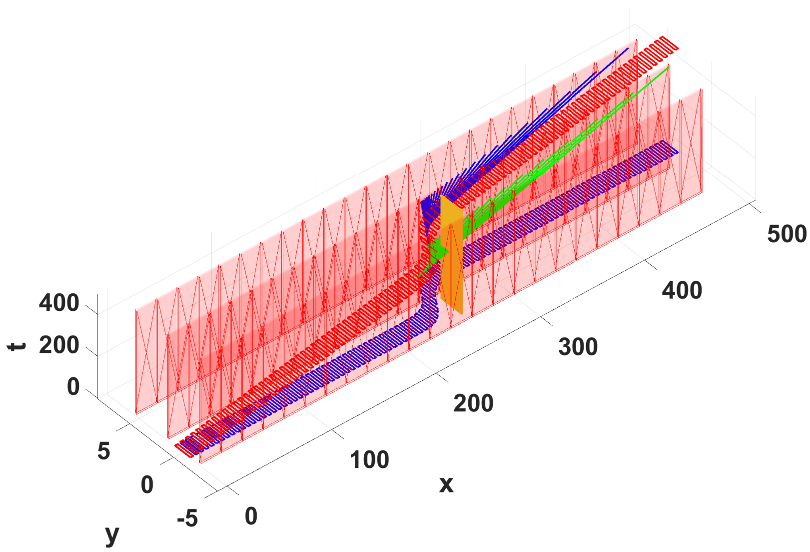

Figure 4.4. Showing TLC ray casts in 3D

Using those ray casts, the TLC of each lane can be calculated and graphed as seen in Figure 4.6. The graphs show that as the vehicle makes the lane change, the TLC goes to zero until it reaches the lane marker. Once it passes the middle lane, the TLC to the outside lane starts with a relatively small number as the vehicle is pointed toward the lane and then approaches infinity as the vehicle is pointed away from the lane.

4.2.2 Relative Lane Position

Relative lane position is a straightforward metric to measure. Assuming that there is horizontal driving along the x-axis, only the current y position is needed. Using the current y position and the y position of the lanes at the current time step, the relative lane position can be calculated. For more complex trajectories, the center line of the

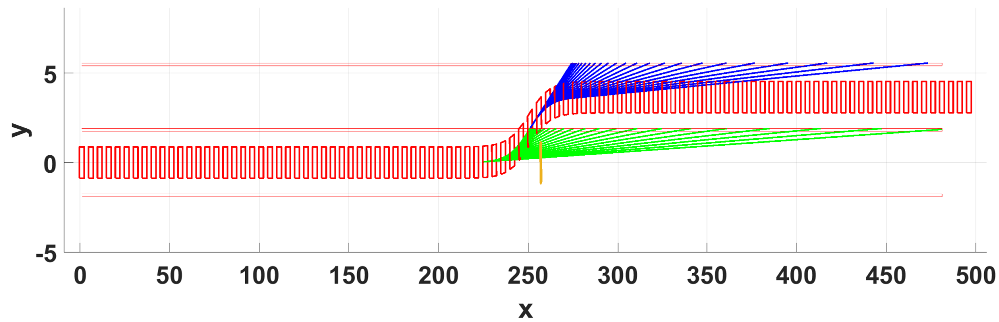

Figure 4.5. Showing TLC ray casts top down

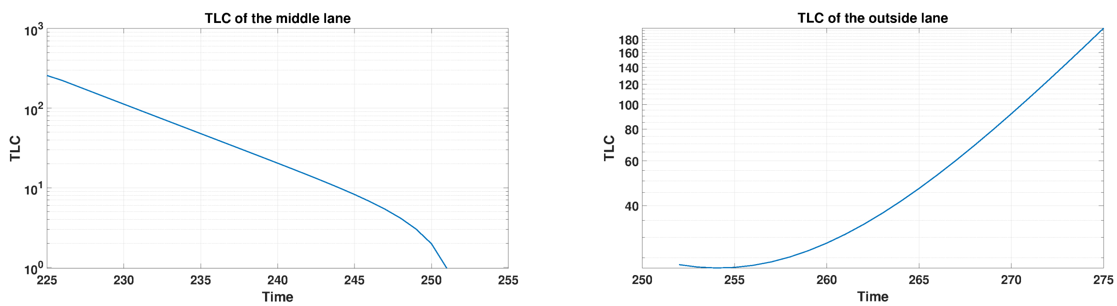

Figure 4.6. TLC

road needs to be defined. The relative lane position is then the difference between the center line and the trajectory.

4.3 Post-encroachment Time

Since PET calculates the time differential between the exit of a vehicle's rear bumper out of and the entrance of another vehicle's front bumper into a defined conflict area, this safety measure is only valid in cases where the paths of two vehicles intersect. An intersection can be detected by means of ray casting.

In order to apply ray casting, a spacetime 3D mesh of the 2D position data of both vehicles over a certain time series (represented by the z-axis) must be created. It is of note that because of reasons similar to those regarding dilation of objects in chapter 3, the first vehicle must be dilated in both dimensions of its envelope before creating this 3D mesh. Once this mesh in spacetime is created, a ray is cast from the path of the second vehicle to that of the first vehicle in the negative z-direction. The PET is then calculated by simply measuring the z-height of the ray at the point of intersection, this can be illustrated in Figure 4.7. Figure 4.7 depicts a ray casting in which a vehicle drives (blue rectangle) over the same area a previous vehicle (red object) was. The ray casts are shown as the dark blue lines projected vertically downwards. Figure 4.8 depicts PET being calculated over a longer trajectory, with the current vehicle being the blue arrows.

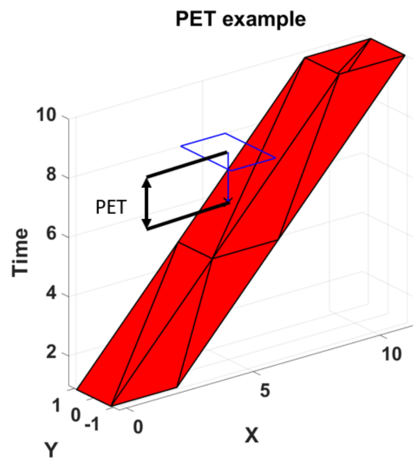

Figure 4.7. PET ray casting

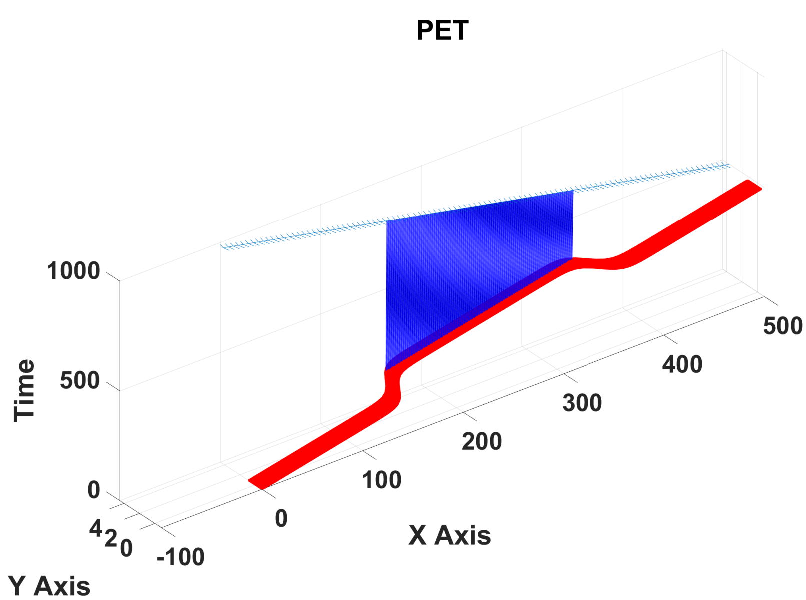

Figure 4.8. PET ray casting

4.4 Deceleration Rate to Avoid Crash

To calculate the DRAC of a vehicle, similar to all the other metrics, a collision course must first be determined before the calculation can be made. Once a collision course has been determined, the horizontal component of the intersecting ray is the distance and the inverse of the slope is the velocity. By using the equation defined in Chapter 2, DRAC can then be calculated. Figure 4.9 shows the DRAC for Figure 4.1. Figure 4.9 shows that as the vehicle gets closer to the object, the deceleration needed increases.

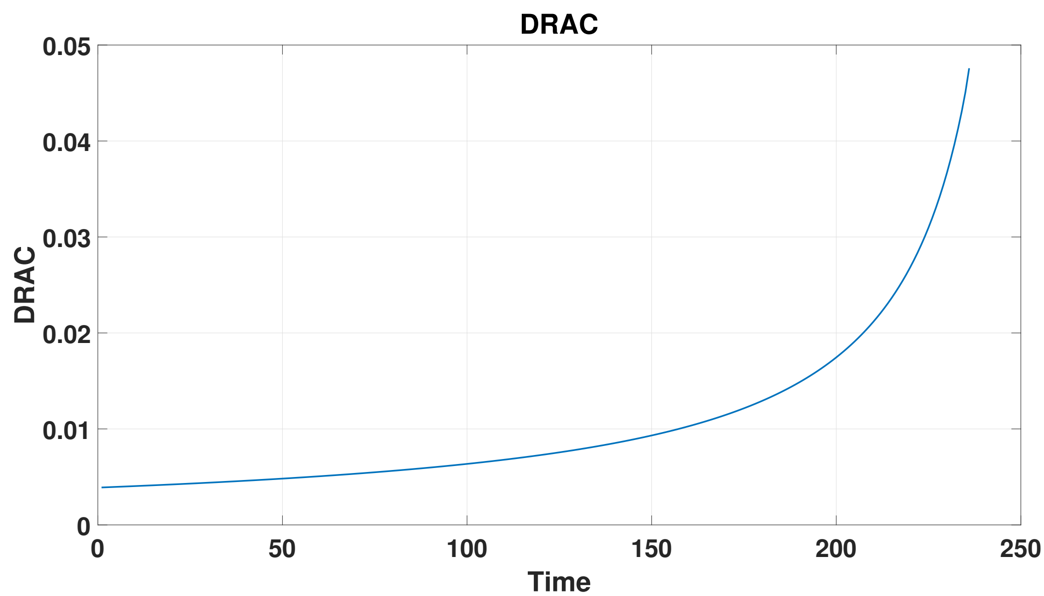

Figure 4.9. DRAC of a vehicle

4.5 Speed Differential

To calculate the speed differential, there first must be a detected collision course between the objects/vehicles. Once a collision course has been determined, the only information needed is the slopes of both the primary vehicle and the object/vehicle it would collide with. After determining the inverse of these slopes, simply subtracting them would give the speed differential.

4.6 Conflict Severity Index

Calculating conflict severity index (CSI) can be done two different ways: using the method developed by Hyden [22] or using the method developed by [8]. As described inthe first method, creating a ratio between TA (time-to-accident) and CS (conflicting speed) results in the graph seen in Figure 4.10. In Figure 4.10, it can be seen that the CSI usually stays constant with the severity only increasing as the vehicles get closer to each other.

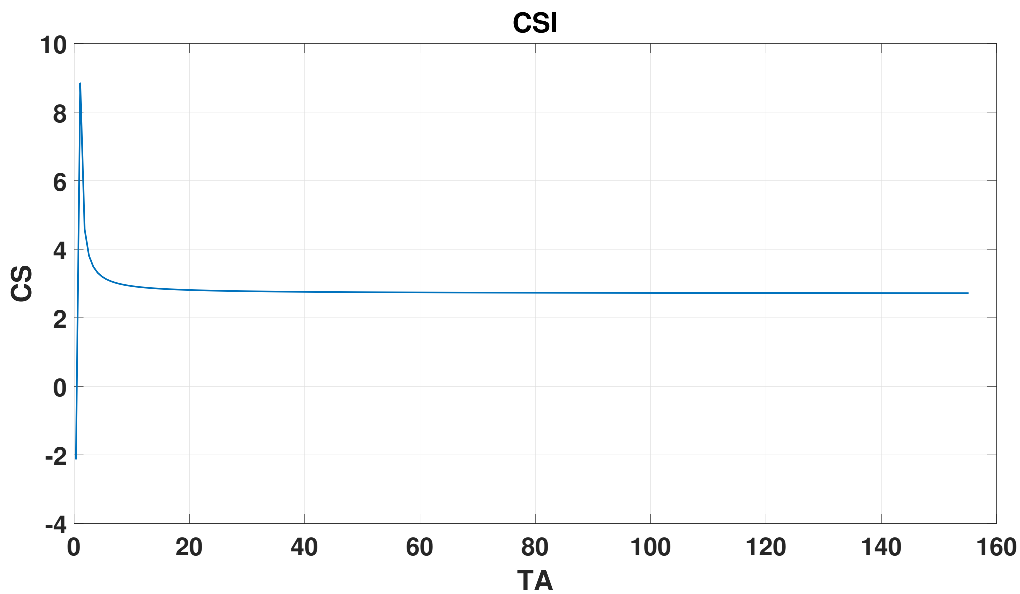

Figure 4.10. CSI

As for the method developed by [8], the PET is calculated the same way as discussed in PET section shown in figure 4.11. The change in kinetic energy is calculated using the inverse of the slopes since that is analogous to the velocity of the vehicles.

Combining that information, the CSI can be calculated throughout the trajectory. This CSI can then be plotted show in 4.12, where it can be seen that the CSI increases exponentially once the blue vehicle gets closer to the red vehicle.

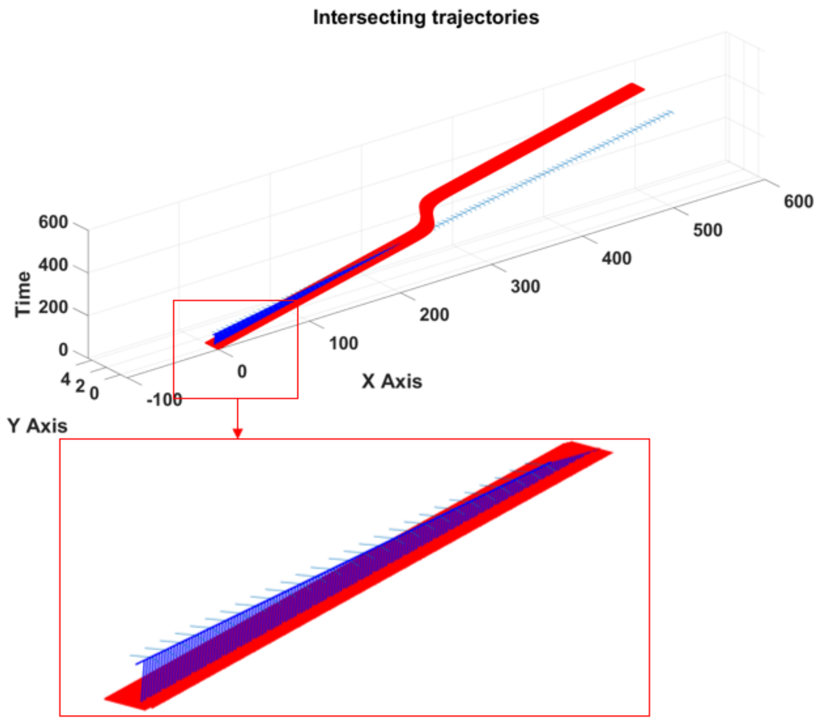

Figure 4.11. PET calculation for [8] CSI

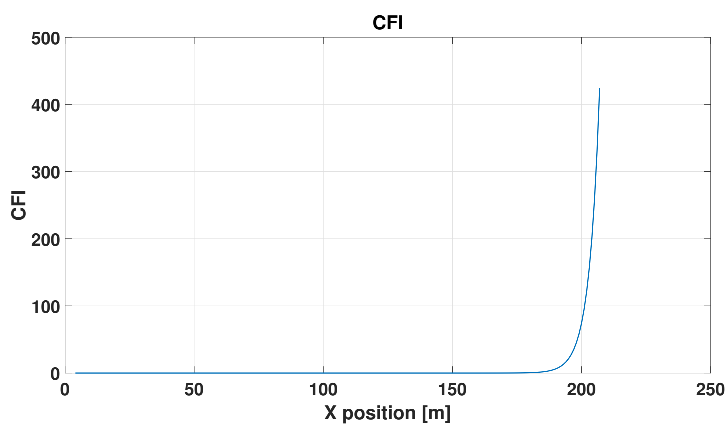

Figure 4.12. PET calculation for [8] CSI

Chapter 5
Conclusion and Future Work

This chapter summarizes the work done within this thesis. It then describes possible future research topics to further develop the idea of using spacetime in the calculation of SSMs.

5.1 Conclusion

This thesis introduces SSMs as a method to estimate the safety of a vehicle through means of measuring parameters that does not require historical crash data. There are a variety of SSMs, many of which are shown in Chapter 2. In that chapter, the studied SSMs (TTC, TLC, PET, SD, DRAC, CSI, and RLP) are described in detail, highlighting a diverse way of calculating each, and each requiring different parameters. This variety makes implementing and understanding these SSMs difficult.

This thesis then presents a unified spacetime solution to calculate SSMs. By representing a vehicle in spacetime, space being the x- and y-axis and z being time, the SSMs can be calculated using a ray cast and simple calculations. Chapter 3 describes spacetime representations and necessary adjustments to use this method, including object dilation and converting objects from slices to 3D meshes. These modifications are done to facilitate ray casting wherein a triangle-ray intersection algorithm is used to determine possible collisions with objects, vehicles, and markings. Chapter 4 then describes in detail how each of the primary SSMs can be determined using spacetime, with the primary outcome that each SSM can be converted to a length along an axis, enabling each to be calculated via a straightforward math operation.

5.2 Future Work

This thesis also revealed avenues for further work. There are SSMs that were not discussed in this thesis, and these can be examined to determine their simplicity in representation within spacetime. Each of these SSMs have their own benefits and drawbacks, but each may be useful for determining the safety of a vehicle.

Another avenue for further research is comparing this unified method against traditional methods of calculating SSMs. Some of the aspects that can be looked at is a comparison of computational demands or the differences in accuracy. This would allow one to determine the relative effectiveness of the unified spacetime methods.

Lastly, the method proposed can be used in path planning algorithms to optimize safety for driving. Since most SSMs are simplified to a single measure a distance along the time axis, spacetime representations allow path planning to be directly optimized to enhance safety outcomes.

Bibliography

[1] Johnsson, C., A. Laureshyn, and T. D. Ceunynck (2018) "In search of surrogate safety indicators for vulnerable road users: a review of surrogate safety indicators," https://doi.org/10.1080/01441647.2018.1442888.
URL https://doi.org/10.1080/01441647.2018.1442888

[2] Miller, R. and Q. Huang (2002) "An adaptive peer-to-peer collision warning system," pp. 317–321 vol.1.

[3] Jiménez, F., J. E. Naranjo, and F. García (2013) "An Improved Method to Calculate the Time-to-Collision of Two Vehicles," **11**(1), pp. 34–42.
URL https://doi.org/10.1007/s13177-012-0054-4

[4] Hou, J., G. F. List, and X. Guo (2014) "New Algorithms for Computing the Time-to-Collision in Freeway Traffic Simulation Models," **2014**, p. 761047.
URL https://doi.org/10.1155/2014/761047

[5] Godthelp, H., P. Milgram, and G. J. Blaauw (1984) "The Development of a Time-Related Measure to Describe Driving Strategy," https://doi.org/10.1177/001872088402600302.
URL https://doi.org/10.1177/001872088402600302

[6] van Winsum, W., K. Brookhuis, and D. de Waard (2000) "A comparison of different ways to approximate time-to-line crossing (TLC) during car driving," **32**(1), pp. 47–56.
URL https://www.sciencedirect.com/science/article/pii/S0001457599000482

[7] Laureshyn, A., M. de Goede, N. Saunier, and A. Fyhri (2017) "Cross-comparison of three surrogate safety methods to diagnose cyclist safety problems at intersections in Norway," improving cyclist safety through scientific research, ICSC2015.
URL https://www.sciencedirect.com/science/article/pii/S0001457516301488

[8] Alhajyaseen, W. K. M. (2015) "The Integration of Conflict Probability and Severity for the Safety Assessment of Intersections," **40**(2), pp. 421–430.
URL https://doi.org/10.1007/s13369-014-1553-1

[9] Lee, Y.-C. and J. H. Mirman (2018) "Parents perspectives on using autonomous vehicles to enhance childrens mobility," **96**, pp. 415–431.
URL https://www.sciencedirect.com/science/article/pii/S0968090X18301645

[10] Bennett, R., R. Vijaygopal, and R. Kottasz (2020) "Willingness of people who are blind to accept autonomous vehicles: An empirical investigation," **69**, pp. 13–27.
URL https://www.sciencedirect.com/science/article/pii/S1369847819301792

[11] Zhang, L., F. Chen, X. Ma, and X. Pan (2020) "Fuel Economy in Truck Platooning: A Literature Overview and Directions for Future Research," **2020**, p. 2604012.
URL https://doi.org/10.1155/2020/2604012

[12] Phan, D., A. Bab-Hadiashar, C. Y. Lai, B. Crawford, R. Hoseinnezhad, R. N. Jazar, and H. Khayyam (2020) "Intelligent energy management system for conventional autonomous vehicles," **191**, p. 116476.
URL https://www.sciencedirect.com/science/article/pii/S0360544219321711

[13] Pimentel, J. and J. Bastiaan (2018) "Characterizing the Safety of Self-Driving Vehicles: A Fault Containment Protocol for Functionality Involving Vehicle Detection," pp. 1–7.

[14] Wang, J., L. Zhang, Y. Huang, and J. Zhao (2020) "Safety of Autonomous Vehicles," **2020**, p. 8867757.
URL https://doi.org/10.1155/2020/8867757

[15] Jardim, A. S., A. M. Quartulli, and S. V. Casley (2013) "A Study of Public Acceptance of Autonomous Cars," .

[16] Othman, K. (2021) "Public acceptance and perception of autonomous vehicles: a comprehensive review," **1**(3), pp. 355–387.
URL https://doi.org/10.1007/s43681-021-00041-8

[17] Jefferson, J. and A. Mcdonald (2019) "The Autonomous Vehicle Social Network: Analyzing Tweets After a Recent Tesla Autopilot Crash," .

[18] "FARS 2019 Final File and 2020 Annual Report File," .

[19] "Work zone data," .
URL https://workzonesafety.org/work-zone-data/

[20] Elvik, R. (1988) "Some difficulties in defining populations of entities for estimating the expected number of accidents," **20**(4), pp. 261–275.

URL https://www.sciencedirect.com/science/article/pii/0001457588900541

[21] Elvik, R. and A. Mysen (1999) "Incomplete Accident Reporting: Meta-Analysis of Studies Made in 13 Countries," https://doi.org/10.3141/1665-18.
URL https://doi.org/10.3141/1665-18

[22] Hydén, C. and L. Linderholm (1987) "The Swedish Traffic-Conflicts Technique," *International Calibration Study of Traffic Conflict Techniques*, pp. 133–139.

[23] Tarko, A. P. (2018) "Surrogate measures of safety," pp. 383–405.

[24] Wang, C., Y. Xie, H. Huang, and P. Liu (2021) "A review of surrogate safety measures and their applications in connected and automated vehicles safety modeling," **157**, p. 106157.
URL https://www.sciencedirect.com/science/article/pii/S0001457521001883

[25] Hayward, J. C. (1972) "NEAR-MISS DETERMINATION THROUGH USE OF A SCALE OF DANGER," *Highway Research Record*.

[26] Qu, C., W.-Y. Qi, and P. Wu (2018) "A High Precision and Efficient Time-to-Collision Algorithm for Collision Warning Based V2X Applications," pp. 1–5.

[27] MO, Z. (2020) "A NEW SURROGATE SAFETY INDICATOR FOR MICROSCOPIC ROAD SAFETY EVALUATION," .

[28] Naseralavi, S., N. Nadimi, M. Saffarzadeh, and A. R. Mamdoohi (2013) "A general formulation for time-to-collision safety indicator," **166**, pp. 294–304.

[29] Songchitruksa, P. (2004) copyright - Database copyright ProQuest LLC; ProQuest does not claim copyright in the individual underlying works; Last updated - 2023-03-02.
URL https://ezaccess.libraries.psu.edu/login?url=https://www.proquest.com/dissertations-theses/innovative-non-crash-based-safety-estimation/docview/305150930/se-2

[30] Winsum, W. V. and H. Godthelp (1996) "Speed Choice and Steering Behavior in Curve Driving," https://doi.org/10.1518/001872096778701926.
URL https://doi.org/10.1518/001872096778701926

[31] Allen, B. L., B. T. Shin, and P. J. Cooper (1978) *Tech. rep.*

[32] Goyani, J., A. B. Paul, N. Gore, S. Arkatkar, and G. Joshi (2021) "Investigation of Crossing Conflicts by Vehicle Type at Unsignalized T-Intersections under Varying Roadway and Traffic Conditions in India," https://ascelibrary.org/doi/pdf/10.1061/JTEPBS.0000479.
URL https://ascelibrary.org/doi/abs/10.1061/JTEPBS.0000479

[33] Cooper, D. F. and N. Ferguson (1976) "Traffic studies at T-Junctions. 2. A conflict simulation Record," **17**(Analytic).

[34] Evans, L. (1994) "Driver injury and fatality risk in two-car crashes versus mass ratio inferred using Newtonian mechanics," **26**(5), pp. 609–616.

[35] Joksch, H. C. (1993) "Velocity change and fatality risk in a crash–a rule of thumb," **25**(HS-042 059).

[36] Laureshyn, A., T. De Ceunynck, C. Karlsson, Å. Svensson, and S. Daniels (2017) "In search of the severity dimension of traffic events: Extended Delta-V as a traffic conflict indicator," **98**, pp. 46–56.

[37] Bagdadi, O. (2013) "Estimation of the severity of safety critical events," **50**, pp. 167–174.

[38] Kraay, J. H., A. van der Horst, and S. Oppe (2013) "Manual conflict observation technique DOCTOR (Dutch Objective Conflict Technique for Operation and Research)," .

[39] Gettman, D. (2003) "Surrogate Safety Measures from Traffic Simulation Models," **1840**.

[40] Silver, D. (2021) "Cooperative Pathfinding," **1**(1), pp. 117–122.
URL `https://ojs.aaai.org/index.php/AIIDE/article/view/18726`

[41] URL `https://mutcd.fhwa.dot.gov/rpt/tcstoll/chapter443.htm`

[42] Adminstration, F. H. (2022) Dept. of Transportation, Federal Highway Administration.

[43] Möller, T. and B. Trumbore (2005) "Fast, Minimum Storage Ray/Triangle Intersection," , p. 7es.
URL `https://doi.org/10.1145/1198555.1198746`

[44] Tuszynski, J."Triangle/Ray Intersection," [Online; accessed June 2, 2023.