Simulator to generate realistic data from a vehicle driving in a mine

Department of Mathematics, Linköping University

Emil Kari

LiTH-MAT-EX–2023/14–SE

Credits: **30 hp**

Level: **A**

Supervisors: **Ivar Simonsson**,
Epiroc AB

Fredrik Berntsson,
Department of Mathematics, Linköping University

Examiner: **Martin Singull**,
Department of Mathematics, Linköping University

Linköping: **December 2023**

Contents

Abstract

This project aims to develop a simulator for generating realistic data from vehicles operating in underground mines, encompassing positional data and sensor values of the velocity and angle. The project addresses the challenge of analyzing the Hybrid Positioning algorithm within Mobilaris Onboard™, a navigation system for underground mines. The absence of the 100% ground truth for vehicle positions in the post-analysis of sensor log files necessitates the creation of this simulator. The project's mission includes generating vehicle paths and corresponding sensor readings, focusing on realism. Additional considerations include introducing realistic noise and integrating the simulator's output with visualization tools. Furthermore, the project aims to develop a tool for comparing simulated sensor values with actual sensor data, facilitating algorithm refinement and development. The project also incorporates time series analysis to interpret the sensor data generated by the simulator. This approach is crucial for understanding patterns and trends in the vehicle's positional and velocity data over time, providing valuable insights for refining the navigation algorithm.

Keywords:
Time series analysis, Navigation system, Underground mines, Post-analysis.

Acknowledgements

I would like to express my deepest gratitude to Ivar Simonsson and Emil Lindh at Epiroc AB for their invaluable guidance, patience and expertise. I would also like to thank the other colleagues at Epiroc AB for their insights and camaraderie throughout this project. My gratitude also goes to my academic supervisor, Fredrik Berntsson and the examiner, Martin Singull, for their expertise and advice throughout this project.

Chapter 1

Introduction

1.1 Background

Mobilaris Onboard^{TM} stands out as a valuable tool, providing guidance to drivers navigating underground mines, similar to how GPS devices assist drivers on the surface. A pivotal element of the Onboard product is the Hybrid Positioning algorithm, commonly called HybridP. This algorithm utilizes the vehicle's sensor values and readings from nearby Wi-Fi access points and Bluetooth Low Energy beacons to determine the vehicle's position within the mine as accurately as possible.

In mining operations, when vehicles are equipped with Onboard drives, the logs containing sensor values and algorithm parameters are recorded and stored in a cloud service. However, a challenge arises in post-analysis because the log files do not capture the vehicle's precise position. This limitation stems from the HybridP's inability to ascertain the real position with 100% certainty. Consequently, the absence of true positions challenges the team when comparing and evaluating the positioning algorithm.

1.2 The mine

This thesis focuses on the Kristineberg mine in the Lycksele municipality in Västerbotten County. The mining operations in the Kristineberg mine have been ongoing since 1946 and it is known for extracting zinc, gold, silver, copper and lead from its ore and waste rock. The mine road is a well-designed two-way road with ample pockets strategically placed to enable smooth passing of

oncoming traffic and it extends to a depth of 1350 meters.

1.3 Purpose

This study aims to create a simulator that generates realistic sensor data together with realistic positions. The simulator's output will be logged in a format consistent with the existing log files. While the log files encompass various categories of sensor values, extending beyond those illustrated in Figure 1.1, our focus is specifically on generating velocity, angle and positions $[x, y]$ for each time step. The log files utilized in this thesis exclusively represent runs from the Kristineberg mine. Notably, log files often exhibit varying quantities of sensor values due to differences in the duration of drivers' time spent in the mine. In the case of the log file depicted in Figure 1.1, the driver's total time in the mine is nearly three hours.

time	angle	directionInd	hybridpStable	nodeId	prevNodeId	speed	x	y
2023-02-08T07:47:52+01:00	-0.2117747	1	true	7633-7634_2	7633	36	2306	1645
2023-02-08T07:47:53+01:00	-0.2057652	1	true	7634-7635_0	7633-7634_2	34	2292	1648
2023-02-08T07:47:54+01:00	-0.2020589	1	true	7634-7635_2	7634-7635_0	33	2283	1650
2023-02-08T07:47:55+01:00	-0.18333082	1	true	7634-7635_5	7634-7635_2	31	2270	1652
2023-02-08T07:47:56+01:00	-0.23457345	1	true	7635-930_1	7634-7635_5	28	2260	1654
2023-02-08T07:47:57+01:00	-0.38740912	1	true	930-931_0	7635-930_1	25	2252	1656
2023-02-08T07:47:58+01:00	-0.2010675	1	true	930-931_1	930-931_0	24	2247	1657
2023-02-08T07:47:59+01:00	-0.2010675	1	false	930-931_1	930-931_0	24	2247	1657
2023-02-08T07:48:00+01:00	-0.60256547	1	true	931-933_4	931-933_3	25	2205	1673
2023-02-08T07:48:01+01:00	-0.64244664	1	true	933-934_0	931-933_4	29	2198	1678
2023-02-08T07:48:02+01:00	-0.65955037	1	true	933-934_1	933-934_0	29	2194	1681
2023-02-08T07:48:03+01:00	-0.6526592	1	true	933-934_2	933-934_1	28	2191	1684
2023-02-08T07:48:04+01:00	-0.704517	1	true	933-934_4	933-934_2	26	2184	1690
2023-02-08T07:48:05+01:00	-0.71235466	1	true	934-936_0	933-934_4	28	2177	1695
2023-02-08T07:48:06+01:00	-0.6950118	1	true	934-936_1	934-936_0	28	2174	1697
2023-02-08T07:48:07+01:00	-0.7138671	1	true	936-938_0	934-936_1	27	2164	1705
2023-02-08T07:48:08+01:00	-0.7094037	1	true	936-938_1	936-938_0	28	2161	1708
2023-02-08T07:48:09+01:00	-0.7168848	1	true	936-938_2	936-938_1	26	2158	1711

Figure 1.1: A segment of a log file from the Kristineberg mine, covering a duration of 17 seconds. Angle is specified in radians, velocity in kilometers per hour, x and y in meters.

1.4 English support

ChatGPT has been utilized to improve the English.

Chapter 2

Theory

This chapter presents the theoretical background that underlies the method chapter. The theory is essential for understanding the methodology used to create the simulator, which is the main focus of this thesis.

2.1 Node graph

The node graph, illustrated in Figure 2.1, provides a simplified representation of the mine map. Bold points represent node IDs, each possessing unique x and y positions. It is important to emphasize that the separation between consecutive node IDs does not exceed 5 meters. Furthermore, each node ID may have a variable number of neighbors. A node ID with more than two neighbors indicates the initiation of a subroad or a pocket.

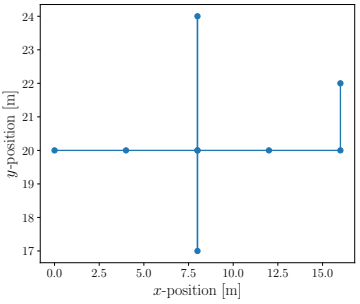


Figure 2.1: Segment within a made-up node graph.

2.2 Bézier curve

A Bézier curve is a parametric curve defined by a set of control points. The comprehension of a Bézier curve is based on the concept of convex sets. According to [4], the definitions of a Bézier curve are as follows:

Definition 1. *Let the control points be P_0, P_1,..., P_n. The convex hull formed by the control points $\{P_i\}_{i=0}^n$ consists of all points that can be written as a convex linear combination of $\{P_i\}_{i=0}^n$.*

Definition 2. *A Bézier Curve for any degree n is defined as*

$$s(t) = \sum_{i=0}^{n} \binom{n}{i} t^i (1-t)^{n-i} P_i, \quad t \in [0,1]. \tag{2.1}$$

Theorem 1. *The endpoint interpolation properties of the Bézier curve are $s(0) = P_0$ and $s(1) = P_n$.*

Theorem 2. *The endpoint tangent properties are $s'(0) = n(P_1 - P_0)$ and $s'(1) = n(P_n - P_{n-1})$.*

Proof. Since

$$s(t) = (1-t)^n P_0 + nt(1-t)^{n-1} P_1 + \left(\sum_{i=2}^{n-1} \binom{n}{i} t^i (1-t)^{n-i} P_i \right) + t^n P_n. \tag{2.2}$$

We can compute the derivative as

$$\begin{aligned} s'(t) = &-n(1-t)^{n-1} P_0 + n\left((1-t)^{n-1} - t(n-1)(1-t)^{n-2}\right) P_1 + nt^{n-1} P_n \\ &+ \left(\sum_{i=2}^{n-1} \binom{n}{i} (it^{i-1}(1-t)^{n-i} - t^i (n-i)(1-t)^{n-i-1}) P_i \right). \end{aligned} \tag{2.3}$$

Inserting $t = 0$ gives

$$s'(0) = -nP_0 + nP_1 = n(P_1 - P_0) \tag{2.4}$$

and inserting $t = 1$ gives

$$s'(1) = -nP_{n-1} + nP_n = n(P_n - P_{n-1}). \tag{2.5}$$

□

Example 1. *The red Bézier curve of order 2 in Figure 2.2 consists of 3 control points.*

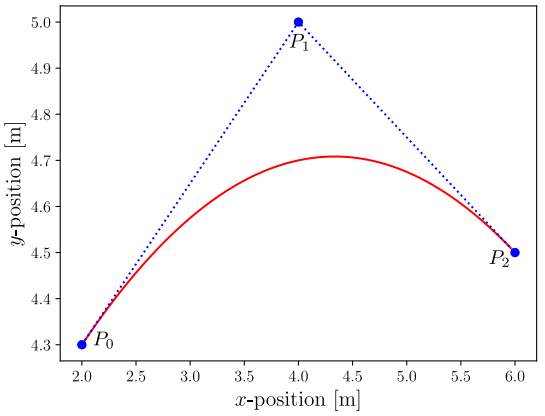


Figure 2.2: Bézier curve of 3 control points.

The curve connects two endpoints, denoted as P_0 and P_2. The direction of the tangent at point P_0 is given by the blue dashed line that goes from P_0 to P_1. Similarly, the direction of the tangent at point P_2 is given by the blue dashed line that goes from P_1 to P_2.

2.3 Stochastic process

The definition of a stochastic process is based on the concepts of probability measure and sample space [8].

Definition 3. *A stochastic process $X(t)$ consists of an experiment with a probability measure $P[\cdot]$ defined on a sample space S and a function that assigns a time function $x(t, s)$ to each outcome s in the sample space of the experiment.*

2.3.1 Stationary process

A stochastic process is stationary if the probability density function of any set of samples does not vary in time. A stochastic process is, in a wide sense, stationary if its mean function and correlation function do not change by shifts in time.

Definition 4. *A stochastic process $X(t)$ is stationary if and only if for all sets of time instants $t_1, \ldots, t_m$ and any time difference τ,*

$$f_{X(t_1),\ldots,X(t_m)}(x_1, \ldots, x_m) = f_{X(t_1+\tau),\ldots,X(t_m+\tau)}(x_1, \ldots, x_m). \tag{2.6}$$

Definition 5. *A function X(t) is a wide sense stationary stochastic process if and only if for all t,*

$$E[X(t)] = \mu_X \text{ and } R_X(t, \tau) = R_X(0, \tau) = R_X(\tau), \tag{2.7}$$

where $R_X(\tau)$ is the autocorrelation function of the stochastic process $X(t)$.

2.3.2 Gaussian process

The Gaussian process is a stochastic process in which every finite linear combination of a collection of stochastic variables is normally distributed. To understand the Gaussian process, the definition of the Dirac delta function according to [3] is needed.

Definition 6. *$X(t)$ is a Gaussian stochastic process if and only if $\mathbf{X} = \begin{bmatrix} X(t_1) & \ldots & X(t_k) \end{bmatrix}^T$ is a Gaussian random vector for any integer $k > 0$ and any set of time instants $t_1, t_2, \ldots, t_k$.*

Theorem 3. *If X(t) is a wide sense stationary Gaussian process, then X(t) is a stationary Gaussian process.*

Definition 7. *The Dirac delta function $\delta(t)$ is defined as having the following properties*

$$\delta(t) = 0 \ \forall t \ , t \neq 0 \tag{2.8}$$

and

$$\int_{-\infty}^{\infty} h(t)\delta(t)\, dt = h(0), \tag{2.9}$$

for any function h(t) which is continuous on the interval $[-\infty, \infty]$.

Definition 8. *$W(t)$ is a white Gaussian noise process if and only if $W(t)$ is a stationary Gaussian stochastic process with the properties $\mu_W = 0$ and the autocorrelation function $R_W(\tau) = \eta_0\delta(\tau)$, where η_0 denotes a positive scalar and $\delta(\tau)$ is the Dirac delta function.*

2.4 QQ plot

The quantile-quantile plot visualizes how well one empirical cumulative distribution follows another. To compare the distributions, the two cumulative distribution functions are the same if and only if their quantiles are the same. Following [1], the quantile point q for a random variable X is a point that satisfies

$$P(X > q) = F_X(q) = p, \tag{2.10}$$

where F_X is the cumulative distribution function (CDF) of X. We can find the inverse of CDF since CDF is monotonically increasing. The QQ plot of two random variables X and Y is defined to be a parametric curve $C(p)$ parameterized by $p \in [0, 1]$:

$$C(p) = \left[F_X^{-1}(p), F_Y^{-1}(p)\right]. \tag{2.11}$$

Theorem 4. *The QQ plot for two Gaussian distributions is a straight line.*

Proof. Assume

$$X \sim N(\mu_1, \sigma_1^2),\ Y \sim N(\mu_2, \sigma_2^2).$$

Let $Z \sim N(0, 1)$ and $\Phi(z) = P(Z \leq z)$. Denote q_1 and q_2 to be the pth quantiles for X and Y respectively. Then we have

$$p = P\left(X \leq q_1\right) = P\left(\frac{X - \mu_1}{\sigma_1} \leq \frac{q_1 - \mu_1}{\sigma_1}\right) = \Phi\left(\frac{q_1 - \mu_1}{\sigma_1}\right). \tag{2.12}$$

Hence, the explicit parameterized QQ plot is given by

$$q_1(p) = \mu_1 + \sigma_1 \Phi^{-1}(p) \tag{2.13}$$

and

$$q_2(p) = \mu_2 + \sigma_2 \Phi^{-1}(p). \tag{2.14}$$

The implicit form without $\Phi^{-1}(p)$ is given by

$$\frac{q_1 - \mu_1}{\sigma_1} = \frac{q_2 - \mu_2}{\sigma_2}. \tag{2.15}$$

Rearranging the equation gives

$$q_1 = \frac{\sigma_1}{\sigma_2}(q_2 - \mu_2) + \mu_1. \tag{2.16}$$

We can see that (2.16) is the equation for a straight line. The slope of the line is given by the ratio σ_1/σ_2.

□

2.5 Descriptive statistics

One way to describe the characteristics of a stochastic variable X is by its moments. According to [2], the moments around the mean value are called central moments and are defined as

$$(X - E[X])^n, \quad \text{for } n = 1, 2, 3, \dots . \tag{2.17}$$

Central moments are denoted as

$$\mu_n = E\left[(X - E[X])^n\right]. \tag{2.18}$$

2.5.1 Skewness

The skewness for a distribution is defined as the third standardized central moment

$$c_s = \frac{\mu_3}{\sigma^3} = \frac{E\left[(X - E[X])^3\right]}{\sigma^3}, \tag{2.19}$$

where σ is the standard deviation. A positive skewness means the distribution is skewed to the right, while a negative value indicates a leftward skew. A perfectly symmetrical distribution, on the other hand, will have a skewness value of 0.

2.5.2 Kurtosis

The kurtosis for a distribution is defined as the fourth standardized central moment

$$c_k = \frac{\mu_4}{\sigma^4} = \frac{E\left[(X - E[X])^4\right]}{\sigma^4}. \tag{2.20}$$

The kurtosis of a Gaussian distribution is 3. When the kurtosis exceeds this value, indicated as $c_k > 3$, the distribution is described as leptokurtic. Such distributions are marked by a more pronounced peak and fatter tails than a Gaussian distribution, signaling a greater occurrence of extreme values. Conversely, if the kurtosis is less than 3, represented as $c_k < 3$, the distribution is considered platykurtic. Compared to a Gaussian distribution, these distributions are characterized by flatter peaks and slimmer tails, implying a lower incidence of extreme values.

2.6 Fourier analysis and filters

To understand the concept of filters based on Fourier analysis, let us introduce the Fourier and Laplace transform following [7].

Let f be a function defined on all of $\mathbb{R}$ with values in $\mathbb{C}$. The definition of the Fourier transform of f is

$$F(w) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{-jwx} f(x)\, dx, \tag{2.21}$$

where $F : \mathbb{R} \to \mathbb{C}$ and j is the imaginary number. Note that this assumes that the integral of the right-hand side in (2.21) is convergent.

Let g be a complex-valued piecewise continuous function on $[0, \infty)$. The definition of the Laplace transform of g is

$$G(s) = \int_{0}^{\infty} e^{-st} g(t)\, dt, \tag{2.22}$$

where $s \in \mathbb{R}$ and the integral must converge.

If g is defined on all of $\mathbb{R}$ in that case, we assume that $g(t) = 0$ for every $t < 0$. This assumption allows us to present a simple connection between Fourier transform (2.21) and Laplace transform (2.22). The connection is

$$G(s) = \int_{0}^{\infty} e^{-st} g(t)\, dt = \int_{-\infty}^{\infty} e^{-j(-js)t} g(t)\, dt = 2\pi F(-js), \tag{2.23}$$

which is a helpful conclusion to understanding the relationship between Fourier and Laplace transform.

2.6.1 Lowpass filter

A lowpass filter is used to filter away signals that are higher than the cutoff frequency. Functional differences in frequency response arise depending on which filter design is used. The lowpass filter that has been used in this thesis is the Butterworth lowpass filter. The knowledge of filters that are described below, following [6].

A transfer function describes the connection between input and output signals in a linear time-invariant system. The general form of a transfer function is

$$H(s) = \frac{b_m s^m + b_{m-1} s^{m-1} + \cdots + b_1 s + b_0}{a_n s^n + a_{n-1} s^{n-1} + \cdots + a_1 s + a_0}, \tag{2.24}$$

where m denotes the order of the numerator polynomial and n denotes the order of the denominator polynomial.

The Butterworth filter is determined by a transfer function in the s-domain, which should contain only poles and no zeros; therefore, its transfer function in a general form can be concluded from (2.24) as

$$H_{But}(s) = \frac{b_0}{a_n s^n + a_{n-1} s^{n-1} + \cdots + a_1 s + a_0}. \tag{2.25}$$

For the normalized Butterworth filter, the cutoff frequency is $\omega_c = 1$ rad/s and the coefficient of the numerator is $b_0 = 1$. The coefficients of the denominator are

$$a_0 = 1 \tag{2.26}$$

and

$$a_k = [\cos((k-1)\pi/2n)/\sin(k\pi/2n)]a_{k-1}, \tag{2.27}$$

for $k = 1, 2, 3, \ldots, n$.

Let us consider the relationship between Butterworth polynomials (2.25) and the filter's amplitude response. As (2.26) and (2.27) shows, for $n = 1$, the Butterworth filter's transfer function is

$$H_{But1}(s) = \frac{1}{s+1}. \tag{2.28}$$

The transfer function (2.28) to its $j\omega$-domain form is

$$H_{But1}(j\omega) = \frac{1}{j\omega + 1}. \tag{2.29}$$

If we assume that ω is the ratio between the current frequency, ω_x, and ω_c, as

$$\omega(\text{rad/s}) = \frac{\omega_x}{\omega_c}, \tag{2.30}$$

then, the general form of the amplitude response of the Butterworth lowpass filter is

$$|H_{But\,n}(j\omega)| = \frac{1}{\sqrt{1 + \left(\frac{\omega_x}{\omega_c}\right)^{2n}}}, \tag{2.31}$$

for $n = 1, 2, 3, \ldots, N$. For Butterworth lowpass filter of order 1 gives $N = 1$. As the filter's order N increases, we will gradually approach the ideal characteristic even more closely.

2.7 Simple Moving Average

The Simple Moving Average (SMA) is a popular technique used in time series analysis. As per its definition, SMA calculates the average of values over specific time periods. According to [5], the definition formula is

$$\text{SMA} = \frac{p_1 + p_2 + \cdots + p_n}{n}, \tag{2.32}$$

where n is the number of total periods and p_n is the value at period n.

Chapter 3

Method

To address the purpose of the thesis, the initial step involves generating the $[x, y]$ positions of a vehicle navigating in the mine. Subsequently, realistic sensor readings corresponding to these simulated positions are generated. Let v_t^* and θ_t^* represent the simulated sensor values for velocity and angle at time points t, respectively. Additionally, let v_t and θ_t denote the true values for velocity and angle at the same time points. The simulated sensor values are defined by the following equations:

$$v_t^* = v_t + \epsilon_{1t}, \quad \text{for } t = 1, 2, 3, \ldots, n \tag{3.1}$$

and

$$\theta_t^* = \theta_t + \epsilon_{2t}, \quad \text{for } t = 1, 2, 3, \ldots, n, \tag{3.2}$$

where ϵ_{1t} represents the velocity noise and ϵ_{2t} represents the angle noise for time points t.

The process of generating realistic sensor values can be segmented into two distinct problems. The first problem involves creating realistic v_t and θ_t values. The second problem revolves around generating ϵ_{1t} and ϵ_{2t} values. Introducing noise is crucial to enhance the realism of the simulated sensor values, ensuring that velocity and angle exhibit slight variations in each run. Additionally, it is essential for the simulator to ensure that the output produces realistic sensor values in conjunction with realistic positions when executed from the main function. Therefore, evaluation methods of historical data are required.

3.1 The primary road

The primary road is a modified segment of the node graph derived from the Kristineberg mine, as depicted in Figure 3.1. In this adapted version, subroads have been eliminated. End nodes have one neighbor, while other nodes have two neighbors. The inception of the primary road occurred at the mine opening and was truncated, where the road began to exhibit excessive divergence. This truncation gathers more consistent data, ensuring that vehicles travel along the same route. The objective is to obtain more precise measurements for historical data time series analysis.

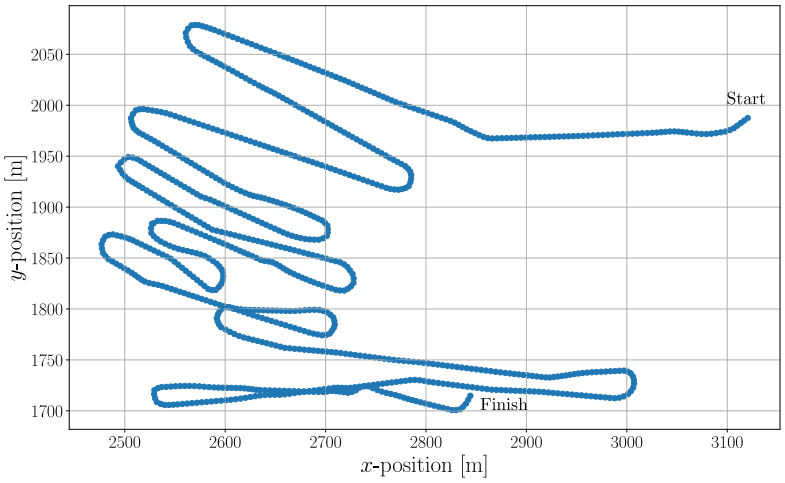


Figure 3.1: The primary road.

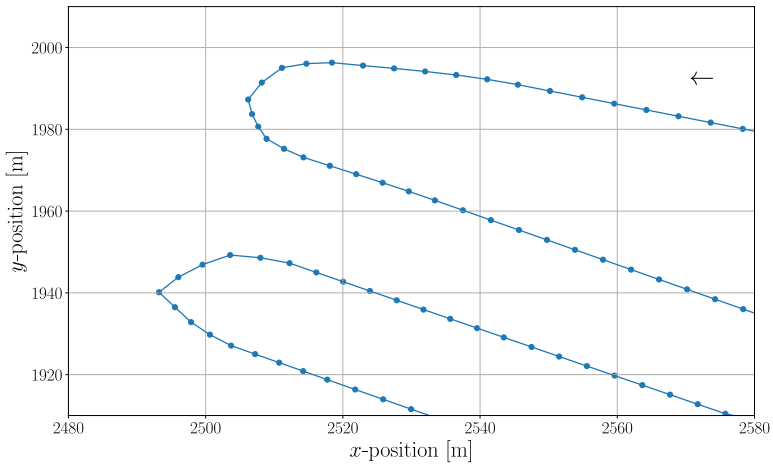


Figure 3.2: Part of the primary road.

Upon closer examination of the primary road curves, as illustrated in Figure 3.2, it becomes evident that they possess excessive steepness, rendering them impractical for vehicular navigation and giving rise to unrealistic driving scenarios. These steep curves arise from the linear assumption between each node ID.

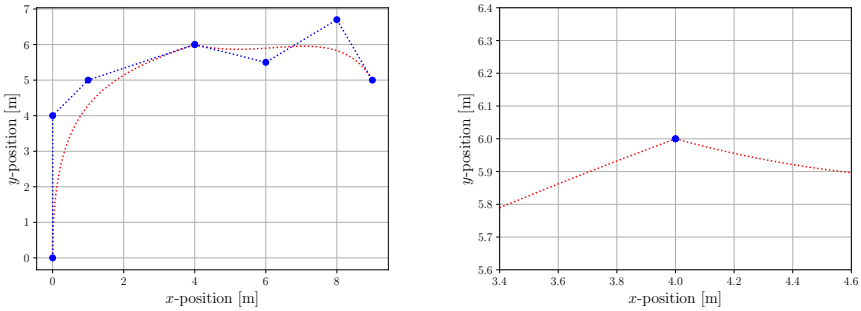
3.2 Simulate the roadway

The simulation of the drive incorporates the use of Bézier curves. The rationale behind employing Bézier curves is to mitigate the presence of unrealistically steep sections in the primary road. Applying Bézier curves to the primary road transforms these steep curves into smoother trajectories, resulting in a more realistic driving experience.

3.3 Improve the roadway

To enhance the adherence of the Bézier curves to the primary road, the primary roads are segmented into multiple paths and individual Bézier curves are generated for each path. Nevertheless, a challenge arises as the connection points between these paths may exhibit unevenness. We refer to Figure 3.3, where two

Bézier curves have been applied to a made-up road, highlighting the potential for uneven connection points.



(a) Two Bézier curves.

(b) Zooming in at the connecting point of the two Bézier curves.

Figure 3.3: Two Bézier curves of the made-up road.

To ensure the smoothness of each path, it is necessary for the control points before and after the connecting point, as well as the connecting point itself, to be linear. Let the initial control points be denoted as $p_1 = [x_1, y_1]$ and $p_2 = [x_2, y_2]$. To achieve linearity, it is necessary to introduce the new points $p_1^* = [x_1^*, y_1^*]$ and $p_2^* = [x_2^*, y_2^*]$, as shown in Figure 3.4. The specifics of these points are detailed by the following equations:

$$y_1^* = k_2 x_1^* + m_2 \tag{3.3}$$

and

$$y_2^* = k_1 x_2^* + m_1, \tag{3.4}$$

where k being the slope coefficient and m indicating the intersection point with the y-axis. The coordinates of the connecting point between each path are denoted as $[x_0, y_0]$. The coefficient k_1 represents the slope of the line from p_1 to $[x_0, y_0]$, while k_2 represents the slope of the line from $[x_0, y_0]$ to p_2. The distance between the connecting point and p_1^*, denoted a, is given by

$$a = \sqrt{(x_0 - x_1^*)^2 + (y_0 - y_1^*)^2}. \tag{3.5}$$

Similarly, the distance between the connecting point and p_2^*, denoted b, is given by

$$b = \sqrt{(x_0 - x_2^*)^2 + (y_0 - y_2^*)^2}. \tag{3.6}$$

The objective is to compute $\bar{p}_1$ and $\bar{p}_2$, where

$$\bar{p}_1 = \frac{[x_1^*, y_1^*] + [x_1, y_1]}{2} \tag{3.7}$$

and

$$\bar{p}_2 = \frac{[x_2^*, y_2^*] + [x_2, y_2]}{2}. \tag{3.8}$$

By upgrading the control points p_1 and p_2 to $\bar{p}_1$ and $\bar{p}_2$ respectively, each Bézier curve achieves smoothness, as depicted in Figure 3.5.

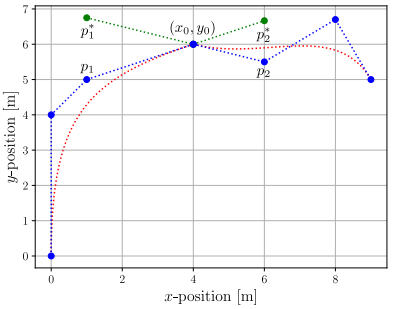
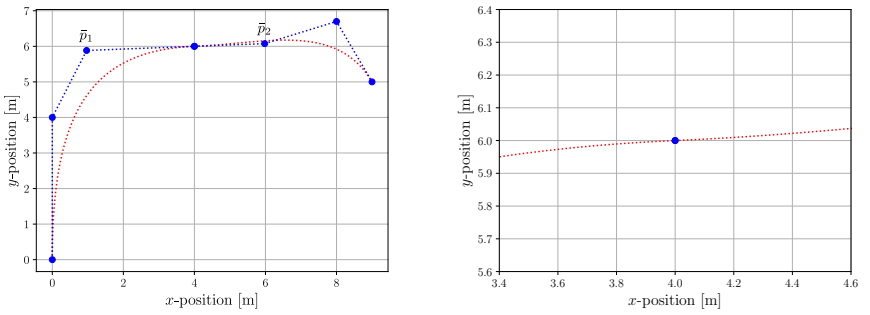


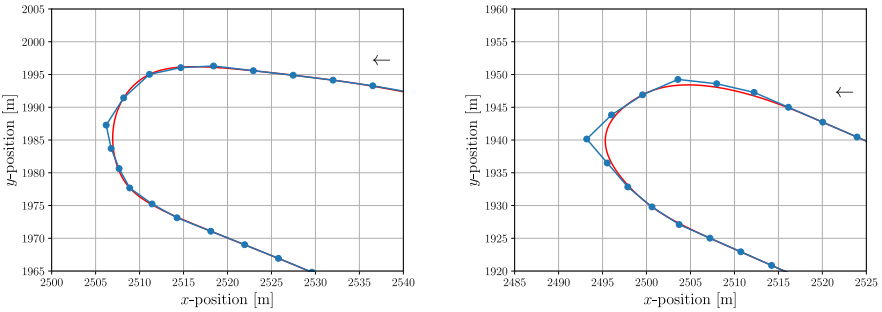
Figure 3.4: Two Bézier curves of the made-up road with the help points.



(a) Two Bézier curves with the new control points.

(b) Zooming in at the connecting point of the two Bézier curves with the new control points.

Figure 3.5: Two Bézier curves of the made-up road with the new control points.



(a) Bézier curve of a curve of the primary road. (b) Bézier curve of another curve of the primary road.

Figure 3.6: Bézier curves of the primary road. The Bézier curves have the order of 4.

Figure 3.6 illustrates the implementation of Bézier curves for the primary road, incorporating the transformation of control points as outlined in Equations (3.7) and (3.8). The resulting red curves represent the composite Bézier curves formed by connecting each segment of the primary road. Notably, the smooth curves of the red line contribute to a realistic driving experience. This red line is subsequently designated as the roadway.

3.4 Calculate the position, angle and velocity

Within the simulator, a function is dedicated to computing the coordinates $[x, y]$ based on the given velocity and angle. Another function determines the velocity and angle from the specified position $[x, y]$. The transition from velocity and angle to $[x, y]$ is achieved by utilizing the polar coordinate system,

$$\left[x_{i+1}, y_{i+1}\right] = r_{i+1}\left[\cos(\theta_{i+1}), \sin(\theta_{i+1})\right]. \tag{3.9}$$

Transitioning from $[x, y]$ to velocity and angle

$$r_{i+1} = \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2} \tag{3.10}$$

and

$$\theta_{i+1} = \arctan\left(\frac{y_{i+1} - y_i}{x_{i+1} - x_i}\right). \tag{3.11}$$

It is important to note that the first set of values represents the origin.

3.5 Create a realistic velocity

To model a realistic velocity profile, it is essential for the velocity to decrease during curves and increase on straight segments. To achieve this, a new angle, denoted as γ, is introduced and defined by

$$\gamma_{k+1} = 180 - |\theta_{k+1} - \theta_k|. \tag{3.12}$$

Here, γ_k represents the angle between two consecutive lines, as Figure 3.7 illustrates. The purpose of γ_k is to provide a straightforward means of determining whether the vehicle is traversing a curve or a straight line. In a straight line, γ_k assumes a value of 180 degrees, whereas a smaller value indicates a slight difference in angle between successive lines, signifying a curve.

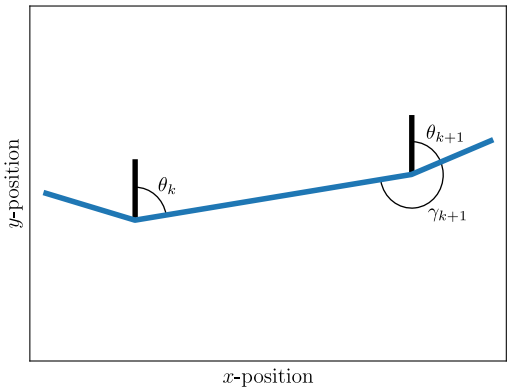


Figure 3.7: Definition of γ and θ.

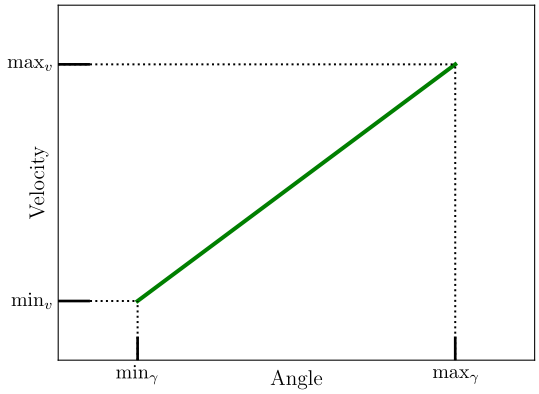


Figure 3.8: Relationship between the velocity and angle γ.

The roadway comprises twenty thousand coordinates, resulting in nearly identical γ values. A linear relationship is established between the velocity and the angle γ to generate a realistic velocity profile, as depicted in Figure 3.8. The linear equation is given by:

$$v_{k+1} = \frac{\max_v - \min_v}{\max_\gamma - \min_\gamma}(\gamma_{k+1} - \min_\gamma) + \min_v. \tag{3.13}$$

This linear relationship ensures that even slight variations in the angles γ result in significant changes in velocities, effectively causing a decrease in velocities during curves. The constants $\max_v$ and $\min_v$ can be easily adjusted based on the desired values, with the evaluation models of historical data from log files guiding the selection of optimal parameter values. Similarly, $\max_\gamma$ and $\min_\gamma$ represent the highest and lowest angle values within the roadway, respectively.

To implement the values of the roadway, we study the equation

$$t_i = \frac{\left|\left[x_i, y_i\right] - \left[x_{i+1}, y_{i+1}\right]\right|}{v_i} \simeq 1. \tag{3.14}$$

Equation (3.14) evaluates the coordinates after a one-second interval and marks a bold point on the roadway. The choice of one second corresponds to the log files' timestep.

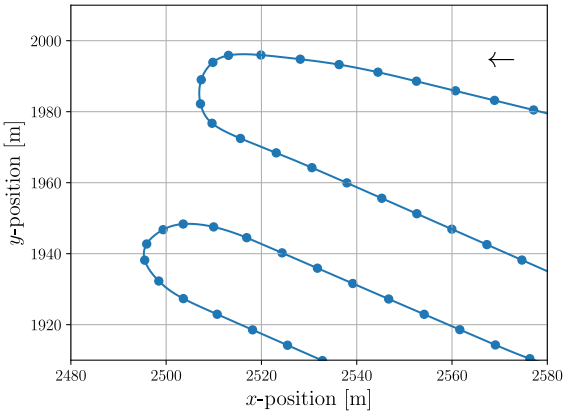


Figure 3.9: Simulated values for a section of the roadway without using SMA.

The spacing between consecutive points in Figure 3.9 signifies the distance covered by the vehicle in one second. A smaller distance between bold points indicates a lower velocity, while a longer distance reflects a higher velocity. Although Figure 3.9 illustrates a decrease in velocity within curves, achieving more realistic simulated values necessitates a more pronounced reduction in velocity.

3.6 Simple Moving Average on velocity

We apply Simple Moving Average (SMA) to the velocity data to obtain more realistic simulated values, see Figure 3.10. The application of SMA results in a new velocity vector that is diminished in size compared to the original full-size velocity vector. To ensure compatibility in size, a modified SMA approach is required. This modification involves centering the new velocity vector and filling in values both before and after the vector. These additional values can be identical to the nearest existing values.

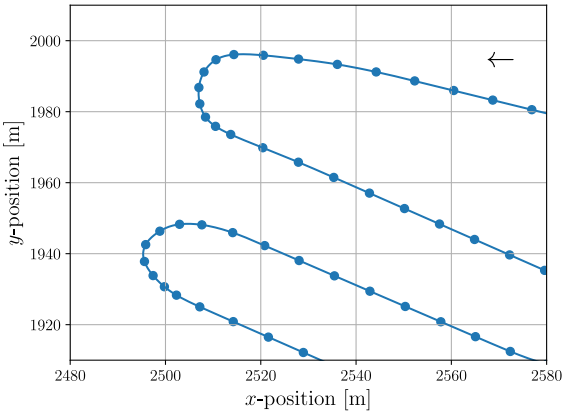


Figure 3.10: Simulated values for a section of the roadway using SMA.

3.7 Model the noise in angle

The examination of the noise in the angle involved the utilization of the Butterworth filter. The output is divided into raw, noise and filtered signals by incorporating historical angle data into the Butterworth filter. The objective of the lowpass filter is to simulate realistic steering wheel movement. Achieving realistic steering wheel movements entails testing different filters, adjusting filter orders and selecting appropriate cutoff frequencies. To ensure realism, the filter must eliminate irregular curves where the wheel's movement is excessively constrained.

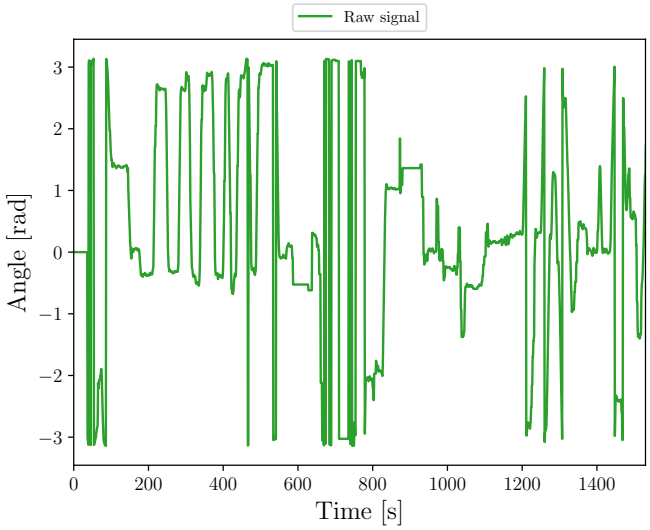


Figure 3.11: The raw signal, where the angle is defined in the interval $[-\pi, \pi]$.

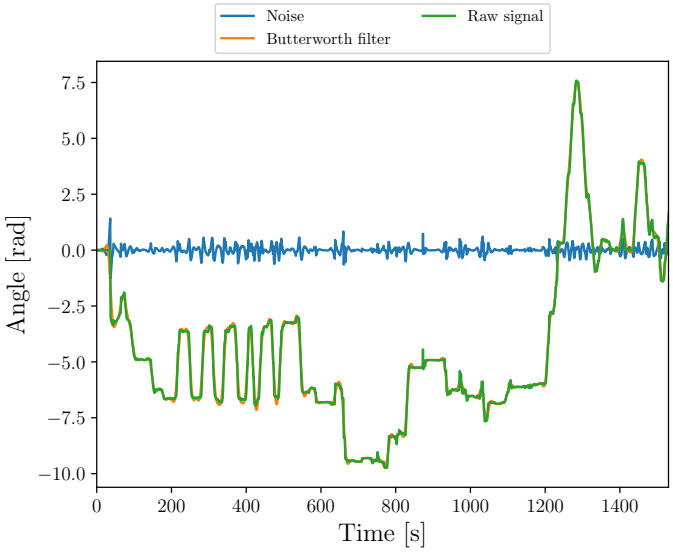


Figure 3.12: Filtering the raw signal, where the angle is defined in $[-\infty, \infty]$.

Figure 3.11 illustrates that the raw signal is exclusively defined within the interval $[-\pi, \pi]$, with a vertical jump occurring whenever the raw signal surpasses this interval. In Figure 3.12, the raw signal is converted to achieve continuity within the $[-\infty, \infty]$ interval. Upon applying the filter, it becomes evident that the noise introduced by the filter exhibits an increase at curves and a decrease at straight lines. No rationale is provided for the variation in noise between straight lines and curves. For the purposes of this thesis, it is assumed that the noise should remain independent of the substantial movements of the steering wheel.

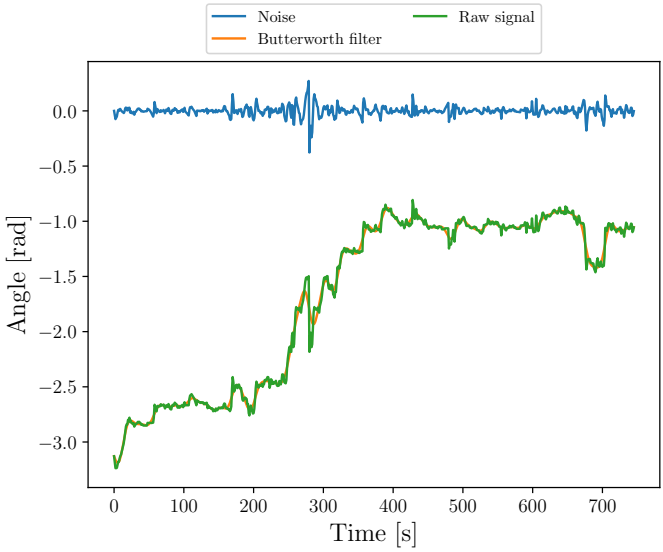
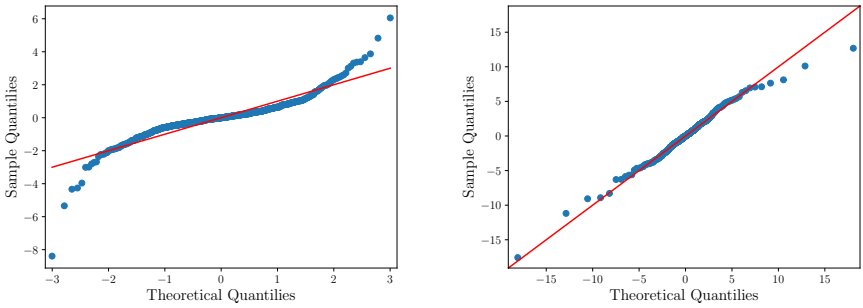


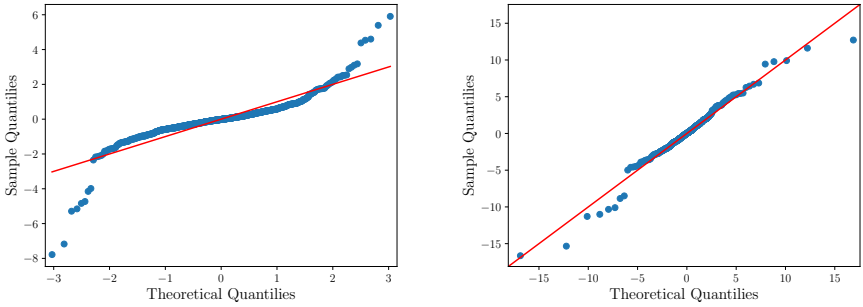
Figure 3.13: Filtering the raw signal of small turnings.

In investigating the noise, we excluded segments where the driver did not engage in significant turns and concatenated those segments consecutively. This process seamlessly connected the concluding part of one nearly straight line to the commencement of the next, repeating this procedure for all relevant sections. Subsequently, applying a Butterworth filter allowed for the exploration of the noise. See Figure 3.13 for a visual representation of this method.



(a) QQ plot Gaussian distribution. (b) QQ plot Student's t-distribution.

Figure 3.14: QQ plot of the angle noise from one log file.



(a) QQ plot Gaussian distribution. (b) QQ plot Student's t-distribution.

Figure 3.15: QQ plot of the angle noise from another log file.

Utilizing the QQ plot facilitates the determination of the noise distribution. Figures 3.14 and 3.15 reveal that the QQ plots of the Gaussian distribution exhibit fat tails. This phenomenon is attributed to the over-dispersion of the data, where an increased frequency of outliers results in fatter tails. Over-dispersed data is commonly characterized by a leptokurtic profile, indicating a higher prevalence of extreme values, see Section 2.5.2. Moreover, Figures 3.14 and 3.15 suggest a better alignment with the Student's t-distribution, implying a potential Student's t-distribution for the data. Built-in functions are used to

determine the degrees of freedom, which indicate how many values in the results can vary independently.

3.8 Model the noise in velocity

The method used to model noise in velocity differed from that for angle noise. In the log files, velocity values are recorded as integers. However, the resulting velocities are expressed in decimals when filtering the raw signal. This implies that the noise is solely a consequence of rounding these values to the nearest integer.

Evaluation methods are essential for modeling velocity noise. Through evaluation methods, we can compare simulated values with historical data, aiming to mimic the historical values closely. The evaluation method depicted in Figure 4.1 illustrates that simulated values in straight lines appear excessively uniform. Given that historical data are integers, simulated values should exhibit more variability. To address this, white noise is introduced. The selection of white noise parameters, specifically a standard deviation of 0.3 for simulated values equal to or greater than 6.5 m/s and a standard deviation of 0.1 for values less than 6.5 m/s, has been determined through testing. Incorporating this white noise into the simulated values enhances their resemblance to historical data, as shown in Figure 4.2.

3.9 Adding the noise to the simulated values

For each simulated point $[x_i, y_i]$, which includes noise, the precise distance and angle to the next noise-free simulated point $[x_{i+1}, y_{i+1}]$ are determined. Subsequently, noise in both velocity and angle is applied to these updated values of distance and angle, measured from the point $[x_i, y_i]$ to $[x_{i+1}, y_{i+1}]$. This process of recalculating the distance and angle at each step is crucial to prevent the cumulative effect of noise, which might otherwise result in a significant deviation from the intended path on the roadway.

3.10 Evaluation methods

Evaluation methods of historical data are required to know if the simulated values are realistic. Two different evaluation methods have been employed. The first evaluation method analyzes the velocity for a section based on the primary road for the simulated values and the historical log files. The section commences

and concludes at the same node ID on the primary road. The methodology aims to identify corresponding matches between the node IDs in the historical data at the start and end segments of the primary road. The starting segment encompasses the primary road's first 50 node IDs, while the ending segment comprises the last 50 node IDs. If a node ID in the historical data is found in both the initial and final segments, the primary road is segmented based on these matches. In cases of multiple hits in each segment, the segmentation involves the first hit in the initial segment and the last hit in the final segment. The cut in the primary road induces changes in the Bézier curves and simulated values in the evaluation method for each historical log file. The distance from the first to the last hit can be easily calculated from the velocities by integrating them. To facilitate a clearer analysis of the first evaluation method, it is necessary to plot the distance from the start on the x-axis against the velocity on the y-axis.

The second evaluation approach also assesses velocity along a path defined by the primary road. Unlike the first evaluation method, which examines velocity along a path starting at the beginning and ending at the end of the primary road, the second evaluation method analyzes velocities for each node ID on the primary road. Since log files often include multiple driving instances, extracting velocities for the corresponding node ID on the primary road from various log files results in multiple velocities for each node ID.

However, the simulated values are contingent on Bézier curves, rendering the approach used to identify each node ID for historical data ineffective for simulated values. The algorithm calculates the smallest Euclidean distance between positions for each t_i from Equation (3.14) and each node ID on the primary road. Subsequently, the algorithm identifies the node ID with the smallest Euclidean distance and matches it with the corresponding simulated values. To facilitate a more comprehensive analysis using the second evaluation method, plotting the index of each node ID of the primary road on the x-axis and the corresponding velocity of each node ID on the y-axis is essential.

For a more thorough examination of the simulated and historical values using the second evaluation method, it is beneficial to compute the mean and standard deviation for historical velocities at each node ID. Furthermore, there is a considerable demand for confidence intervals regarding historical velocity.

In the analysis of the QQ plot for each node ID, the data points exhibit similar characteristics when compared against both the Gaussian and the Student's t-distributions. This similarity is largely attributed to each dataset's high degrees of freedom. However, it's important to note that the data do not align perfectly

with a Gaussian distribution. Additionally, since each node ID comprises fewer than 30 data points, the application of the Central Limit Theorem (CLT) is less straightforward. The CLT typically requires a larger sample size to assume that the sampling distribution of the mean will be approximately Gaussian. Despite this, for the purpose of this analysis, confidence intervals are calculated using the Student's t-distribution. The formula for the confidence interval is:

$$I_\mu = \left(\bar{x} - t_{a/2}(n-1)\frac{s}{\sqrt{n}}, \quad \bar{x} + t_{a/2}(n-1)\frac{s}{\sqrt{n}} \right), \tag{3.15}$$

where $\bar{x}$ represents the sample mean, n is the number of data points and s is the standard deviation.

Chapter 4

Results

This chapter presents the output of the simulator.

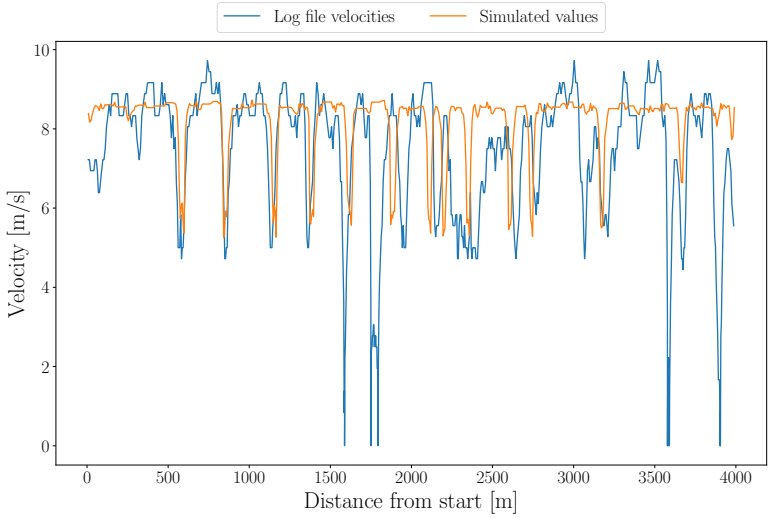


Figure 4.1: The first evaluation method from one log file without the noise. The orange line represents the simulated values, while the blue line depicts the historical data from a single log file. The simulated values are implemented with the Bézier curves of order 4.

Figure 4.1 illustrates the first evaluation method from one log file without the noise of the simulated values. In the first evaluation method, the simulated values and historical data start at the same node ID and finish at the same node ID of the primary road. Initially, the curves of historical data and simulated values closely align. However, a mismatch between the two becomes apparent in the curve from the distance 1750 meters from the start.

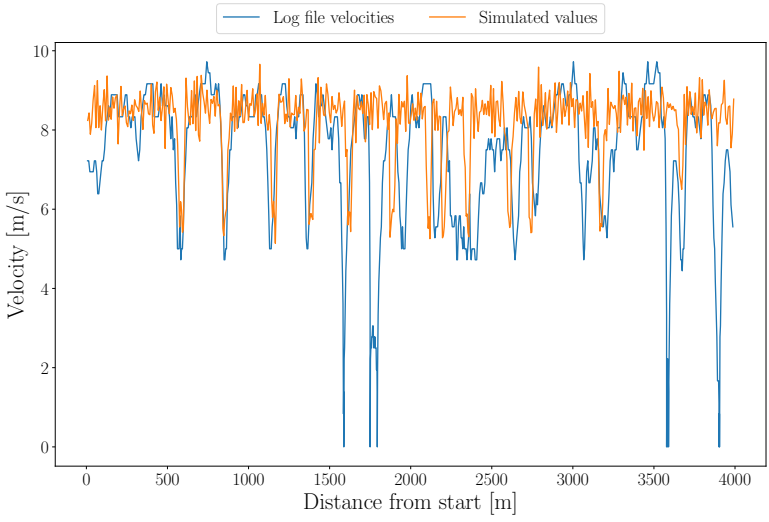


Figure 4.2: The first evaluation method from one log file with the noise. The orange line represents the simulated values, while the blue line depicts the historical data from one log file. The simulated values are implemented with the Bézier curves of order 4.

Figure 4.2 depicts the same scenario as Figure 4.1 but with the introduction of noise. By incorporating white noise into the simulated values, it is evident that the simulated values more closely emulate the historical data. This is attributed to the fact that historical data are integers, implying that simulated values should exhibit more oscillation than the historical data.

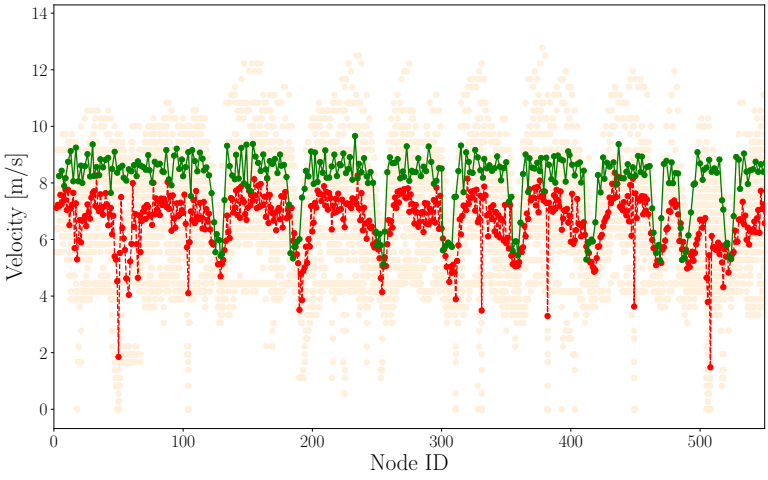


Figure 4.3: The second evaluation method from several log files. The light points represent velocities for each node ID. The red points denote the average velocity for each node ID, while the green points represent simulated values corresponding to the node IDs with measured values. The simulated values are implemented with the Bézier curves of order 4.

The second evaluation method in Figure 4.3 examined velocities for each node ID along the primary road. It is noticeable that there are differences between the simulated values and the average velocity of each node ID. This dissimilarity arises from the choice of parameters $\max_v$ and $\min_v$ in Equation (3.13), which have been set higher than the average historical velocities due to the lower historical velocities during braking caused by oncoming traffic. Furthermore, the simulated values exhibit appropriate braking and acceleration compared to the historical velocity average. However, it is observed that the simulated values tend to have slightly steeper acceleration and braking profiles compared to the average historical velocities.

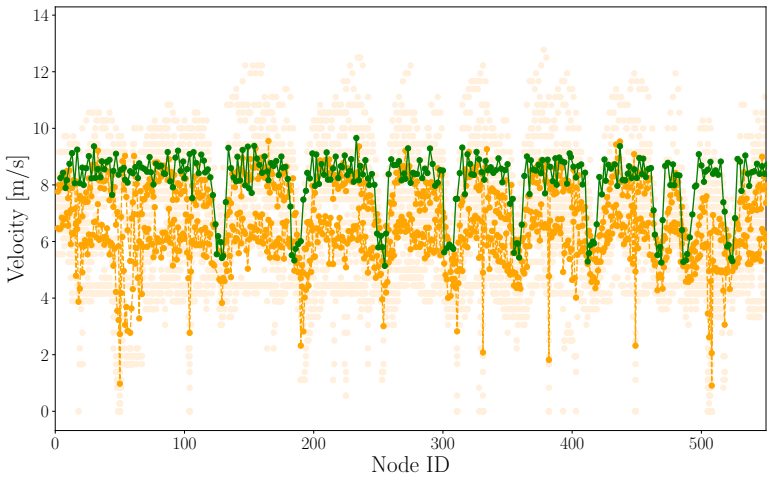


Figure 4.4: The 95% confidence interval for the second evaluation method, considering multiple log files, is depicted. Light points represent velocities at each node ID, while orange points denote the 95% confidence interval. Green points represent simulated values corresponding to the node IDs with measured values. The simulated values are implemented with the Bézier curves of order 4.

The orange points in Figure 4.4 represent the 95% confidence interval for velocities at each node ID, calculated using Equation (3.15). Green points represent simulated values corresponding to the node IDs with measured values. The simulated values have been aligned with the upper bound of the confidence interval to account for potential braking in the presence of oncoming traffic.

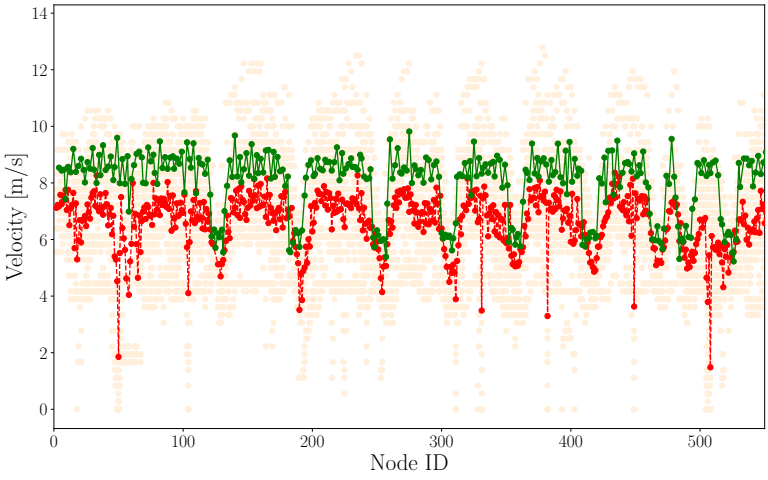


Figure 4.5: The second evaluation method from several log files. The light points represent velocities for each node ID. The red points denote the average velocity for each node ID, while the green points represent simulated values corresponding to the node IDs with measured values. The simulated values are implemented with the Bézier curves of order 6.

Figure 4.5 illustrates a scenario similar to Figure 4.3 but with the Bézier curves of order 6. Notably, the simulated values result in earlier and more intense braking as well as delayed and more pronounced acceleration compared to the simulated values in Figure 4.3.

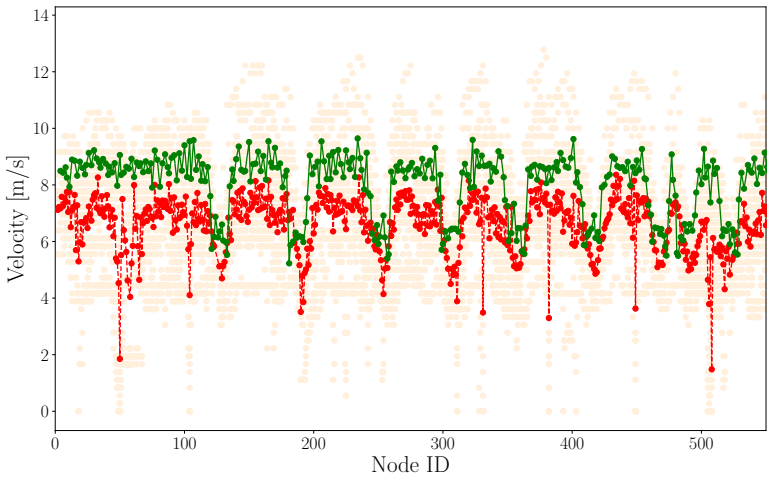


Figure 4.6: The second evaluation method from several log files. The light points represent velocities for each node ID. The red points denote the average velocity for each node ID, while the green points represent simulated values corresponding to the node IDs with measured values. The simulated values are implemented with the Bézier curves of order 8.

Figure 4.6 illustrates a scenario similar to Figure 4.3 but with the Bézier curves of order 8. Notably, the simulated values result in earlier and more intense braking as well as delayed and more pronounced acceleration compared to both the simulated values in Figures 4.3 and 4.5.

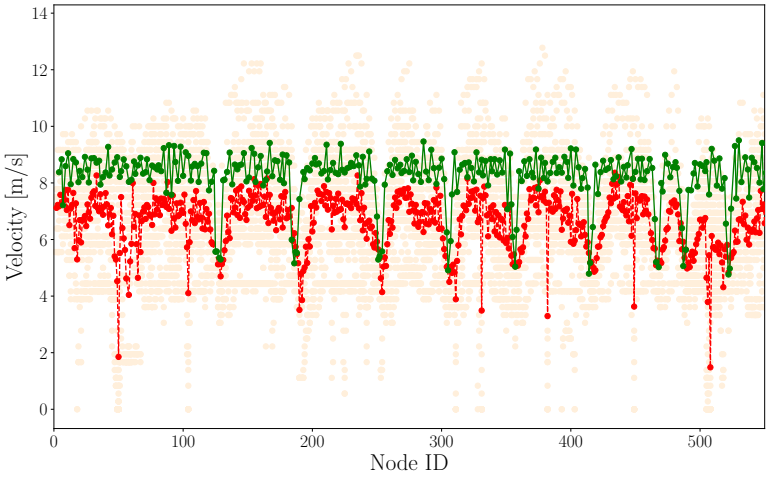


Figure 4.7: The second evaluation method from several log files. The light points represent velocities for each node ID. The red points denote the average velocity for each node ID, while the green points represent simulated values corresponding to the node IDs with measured values. The simulated values are implemented with the Bézier curves of order 2.

Figure 4.7 depicts a scenario similar to Figure 4.3 but with the Bézier curves of order 2. Notably, the simulated values exhibit excessively narrow slopes in curves, leading to delayed braking and earlier acceleration compared to Figures 4.3, 4.5 and 4.6.

Chapter 5

Discussion

In the first evaluation method, the simulated values and historical data start at the same node ID and finish at the same node ID of the primary road. The evaluation algorithm allows the driver to make varied turns within the specified interval. Figure 4.2 illustrates a closer alignment between simulated values and historical data initially, where the acceleration and braking patterns in the simulated values closely match those in the historical data.

However, the driver executes abrupt and intense braking in multiple instances, likely prompted by the identification of oncoming traffic. Notably, at the brake spot, approximately 1750 meters from the start, the driver maintains a low velocity for an extended period, indicative of navigating through a small pocket to make room for other vehicles. In such scenarios, it is common for the vehicle to maneuver slowly backward and forward within the pocket before exiting. This additional mileage within the pocket results in the historical data covering a slightly longer distance than the simulated values. Furthermore, the driver briefly sustains a low velocity at the hard brake spot around 1600 meters from the start. This behavior is likely a response to the oncoming driver needing to navigate through the pocket.

In the second evaluation model, the alignment between simulated values and historical velocities is evident in Figures 4.3 and 4.4. Notably, numerous data points in the historical velocity of each node ID are exceptionally low due to oncoming traffic. This aspect is considered when selecting the parameters $\max_v$ and $\min_v$ in Equation (3.13). The objective is to ensure that simulated values surpass the average historical velocity of the node ID, aligning with the upper limit of the confidence interval.

Regrettably, the acceleration and braking remain slightly too steep. This can be attributed to the higher count of node IDs compared to simulated values, resulting in incomplete results for each node ID. Additionally, the second evaluation model does not differentiate between how vehicles navigate curves. Given the asymmetry of curves, distinct driving behaviors exist when a vehicle enters a curve instead of exiting one. Another notable observation is the considerable variation in the velocity of each node ID in the historical data, stemming from different vehicles traveling at distinct velocities. For instance, a truck must maintain a lower curve velocity than a car.

The impact of the Bézier curves order brings about variations in the simulated values as illustrated in Figures 4.3, 4.5, 4.6 and 4.7. The Bézier curves with an order below 4 yield curves with excessively narrow slopes, leading to delayed braking and premature acceleration. Conversely, the Bézier curves with an order exceeding 4 result in overall early and intense braking, coupled with delayed and forceful acceleration. The optimal outcome is achieved with the Bézier curves of order 4, where simulated values exhibit well-timed braking and acceleration.

The assessment techniques reveal that the simulator produces realistic sensor data along with realistic positions. Nonetheless, assessing the simulated values against historical data proves challenging, given the significant variations in historical data stemming from various factors such as oncoming traffic, diverse vehicles and distinct driving styles present in multiple log files.

Chapter 6

Future research

Although the simulated values appear realistic, further research is necessary to determine how they compare to the historical data. One possible avenue of research is to examine the direction of the drivers in each node ID of the second evaluation method to identify when they align with the simulated values. Additionally, creating a simulator that is tailored to different vehicles would be beneficial, as the historical velocities exhibit significant variation, as demonstrated in the second evaluation model. Finally, another area of future research would involve developing a simulator that can accurately identify the optimal order of the Bézier curve for each curve to generate realistic sensor values.

Bibliography

[1] M.K. Chung. *Brain Network Analysis.* Cambridge University Press, 2019.

[2] J.G. Dolecek. *Random Signals and Processes Primer with MATLAB.* Springer, 2013.

[3] P.P.G. Dyke. *An Introduction to Laplace Transforms and Fourier Series.* Springer, 2001.

[4] G.E. Farin. *Curves and Surfaces for Computer-Aided Geometric Design.* Academic Press, Inc., San Diego, CA, 1997.

[5] P. Meesad, S. Sodsee, W. Jitsakul, and S. Tangwannawit. *Recent Advances in Information and Communication Technology 2021.* Springer, 2021.

[6] D.K. Mynbaev and L.L. Scheiner. *Essentials of Modern Communications.* John Wiley & Sons, 2020.

[7] A.M. Pinkus and S. Zafrany. *Fourier Series and Integral Transforms.* Cambridge University Press, 1997.

[8] R.D. Yates and D.J. Goodman. *Probability and Stochastic Processes: A Friendly Introduction for Electrical and Computer Engineers.* John Wiley & Sons, 2005.