

Autonomous Golf Cart

University of West Florida
Hal Marcus College of Science and Engineering

Erik C. LaBrot
December 1, 2025

Faculty Advisor:
Dr. Tarek Youssef

Presentation Panelists:
Dr. Tarek Youssef
Dr. Minh Ta
Dr. Yazan Alqudah

Abstract

This report details the development of an autonomous driving system implemented on a modified utility golf cart. The system described implements both low level and high level control, realized by a Texas Instruments c2000 series microcontroller, and a Jetson edge computing device. The low level controller interfaces with the golf cart that has been modified in a drive-by-wire style to allow for automated control of steering, throttle, and brake. Likewise, the high level controller interfaces with the c2000 by providing it with velocity and steering angle setpoints, deriving these from a GPS based waypoint follower.

To support controller design, a planar vehicle model is derived that encompasses the longitudinal dynamics and lateral kinematics. This model forms the basis for the controller and simulation design. The simulation is used for controller tuning and algorithm verification before deployment to the physical golfcart.

Contents

List of Figures	4
List of Tables	5
1 Introduction	8
1.1 Motivation and Project Goals	8
1.2 Project Scope and Constraints	8
1.3 Contributions	8
2 System Overview	9
2.1 Vehicle Platform	9
2.2 Sensing and Computation	9
2.3 Control Architecture	9
3 Vehicle Hardware Rework	10
3.1 Initial Vehicle Condition	10
3.2 Component Update	10
3.3 Electrical Rework	10
4 Modeling	11
4.1 Mechanical Layout and Drivetrain	11
4.2 Simplifications and Assumptions	12
4.2.1 Velocity Operating Region	12
4.2.2 Terrain Considerations	12
4.2.3 Aerodynamics	12
4.2.4 Suspension Dynamics	13
4.2.5 Drivetrain Dynamics	13
4.2.6 Physical Modeling	13
4.3 Longitudinal Dynamics	13
4.4 Lateral Kinematics	14
4.5 Combined Model	16
5 Low-Level Control Design	18
5.1 Control Objectives and Architecture	18
5.2 Longitudinal Speed Controller Design	19
5.2.1 Drive Motor PI Controller	20

5.2.2	Brake PI Controller	21
5.3	Steering Controller Design	21
5.3.1	Steering PI Controller	22
5.4	Summary	22
6	Low Level Control Implementation	23
6.1	TI c2000 Microcontroller	23
6.2	MATLAB Toolbox for c2000 Microcontroller	23
6.3	Simulink Model	23
7	Simulation	24
7.1	CARLA Sim	24
7.2	Gazebo Sim	24
7.2.1	Simulation Model	24
7.2.2	Simulation Sensors	24
7.2.3	Simulation Interfaces	25
8	High Level Control Design	26
8.1	Jetson Nano	26
8.2	ROS2	26
8.3	GPS Waypoint Line Following	26
9	Results	27
10	Conclusions	28
A	Derivations	30
B	Vehicle Parameters	31
C	Diagrams	32

List of Figures

4.1	Mechanical Drivetrain Layout	11
4.2	Steering System Layout	12
4.3	Diagram of Forces Acting on Platform along Longitudinal Axis	14
4.4	Diagram of Bicycle Model	15

List of Tables

1	Symbols used in the planar configuration, frames, and state-space model . .	6
2	Symbols used in the longitudinal dynamics model	7

Nomenclature

To contain a list of all symbols and their definitions used in the report.

Table 1: Symbols used in the planar configuration, frames, and state-space model

Symbol	Description
x	x -position of vehicle reference point in $\{W\}$, m
y	y -position of vehicle reference point in $\{W\}$, m
ψ	Yaw heading of the vehicle w.r.t. $\{W\}$, rad
v	Vehicle longitudinal speed along body x -axis, m/s
\dot{x}	Time derivative of x , m/s
\dot{y}	Time derivative of y , m/s
$\dot{\psi}$	Yaw rate, rad/s
\dot{v}	Vehicle longitudinal acceleration, m/s ²
δ	Front wheel steering angle, rad
L	Wheelbase of the vehicle, m
O	Instantaneous centre of rotation of the vehicle
R	Distance from O to the vehicle reference point (radius of rotation), m
θ	Road grade angle (inclination of terrain), rad
\mathbf{q}	Lateral configuration vector, $[x \ y \ \psi]^\top$
\mathbf{x}	State vector of planar model, $[x \ y \ \psi \ v]^\top$
\mathbf{u}	Control input vector, $[T_{\text{motor}} \ T_{\text{brake}} \ \delta]^\top$
$\mathbf{f}(\mathbf{x}, \mathbf{u})$	Nonlinear state-space dynamics, $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$
$\{W\}$	World (inertial) reference frame
$\{B\}$	Body-fixed reference frame attached to the vehicle

Table 2: Symbols used in the longitudinal dynamics model

Symbol	Description
F_x	Net longitudinal force acting on the vehicle, N
F_{drive}	Driving force at the tire–road interface, N
F_{brake}	Braking force at the tire–road interface, N
F_{roll}	Rolling resistance force, N
m	Vehicle mass, kg
J_{eq}	Equivalent rotational inertia of wheels and hubs, referred to the wheel, kg m^2
r_w	Effective wheel radius, m
m_{eq}	Equivalent mass including rotational inertia, kg
T_{motor}	Drive motor shaft torque, N m
T_{brake}	Brake torque applied at the wheel(s), N m
i_{tot}	Total gear ratio (motor to wheel), dimensionless
η_d	Total driveline efficiency, dimensionless
C_{rr}	Rolling resistance coefficient, dimensionless
g	Gravitational acceleration, m/s^2

Chapter 1

Introduction

Introduce the project

1.1 Motivation and Project Goals

[Discuss the purpose of the project]

1.2 Project Scope and Constraints

[Discuss what is and is not included in the project]

1.3 Contributions

[Discuss my contributions to project]

Chapter 2

System Overview

High level description for the golf cart

2.1 Vehicle Platform

[Physical platform as received, brief overview of modifications physical modifications]

2.2 Sensing and Computation

[Describe sensors and computational hardware on the vehicle]

2.3 Control Architecture

[Describe the control architecture in terms of components (e.g. high-level, low level, simulation, etc)]

Chapter 3

Vehicle Hardware Rework

[Go from initial condition of the vehicle to current state. Rewire etc.]

3.1 Initial Vehicle Condition

[Describe as-is vehicle state at project start]

3.2 Component Update

[Discuss updated components (arduino/rpi to c2000/jetson)]

3.3 Electrical Rework

[Discuss golfcart rewire from wiring harness to DIN rails]

Chapter 4

Modeling

This chapter lays out the mathematical framework used in the forthcoming design work. The model derived here is used to dictate simulation modeling as well as controller design. A brief overview of the platform's drivetrain and steering mechanisms is provided, before they are translated to a mathematical expression analytically.

Ultimately, a simplified model based on the bicycle kinematic model is derived, involving the golf cart's longitudinal dynamics to model the relationship between motor torque and velocity.

4.1 Mechanical Layout and Drivetrain

The drivetrain of the golf cart consists of a permanent magnet alternating current (PMAC) motor connected to the original transaxle of the cart. This is accomplished by connecting the motor output to the transaxle input via a chain. From this point, power is transferred to the rear wheels via the transaxle.

The steering system of the golf cart consists of a rack and pinion, controlled by a main steering column to which a DC motor is attached. This connection is mechanically achieved between the output shaft of the motor's gearbox and the steering column via a chain and sprocket. The steering column then actuates the steering rack as if the steering wheel were being turned, according to the DC motor input.



Figure 4.1: Mechanical Drivetrain Layout

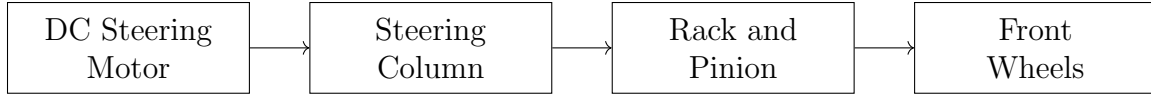


Figure 4.2: Steering System Layout

4.2 Simplifications and Assumptions

The essential idea of the mathematical model for the golf cart is to capture how motor torque relates to linear velocity, and how steering angle changes relate to translation of the platform. The following constraints are imposed to make simulation modeling and controller design more straightforward for this component of the project:

4.2.1 Velocity Operating Region

Due to the low speed operating region of the golf cart (i.e. less than 20 mph), the effects of certain lateral dynamic forces, such as lateral tire slip, are ignored. This simplification is based on Kong et al. [1], whose work shows that below certain speeds and lateral forces, a simplified model is sufficient to describe platform motion. Additionally, a pure rolling model is assumed, where the tires do not slip in the longitudinal direction.

4.2.2 Terrain Considerations

For modeling purposes, terrain is presumed to be level enough such that the inclination or declination of the cart does not meaningfully contribute to the sum of the longitudinal forces acting on the golf cart. Grade forces and load transfer dynamics are excluded from the model. Mathematically, this is represented as any force scaled by $\sin(\theta)$ set to 0.

4.2.3 Aerodynamics

To simplify the modeling math, aerodynamic drag is neglected. At the low speed region the platform operates at, the aerodynamic drag should be small enough to only cause a minor mismatch between simulation and reality.

4.2.4 Suspension Dynamics

Due to the low velocity operating region and terrain assumptions, effects of suspension and pitch dynamics are neglected. It is assumed that the influence these systems would have on the description and control of the vehicle are negligible.

4.2.5 Drivetrain Dynamics

The inertial effect of the drivetrain, such as motor and transaxle inertia, on longitudinal dynamics is neglected for now. Future work may explore experimental measuring of this value to improve the fidelity of the model. However, the inertial effect of the wheel and hub rotation in the sense of effective mass is considered, and is included in the term J_{eq} .

4.2.6 Physical Modeling

For modeling purposes, the cart is assumed to be a symmetric mass with a center of gravity in the geometric center of the body. The effect this has simplifies moments of inertia, and places the CoG equidistant from both axles of the platform.

4.3 Longitudinal Dynamics

The longitudinal dynamics are the body frame forces exerted along the horizontal plane of the platform. These can be derived from Newton's second law, as presented in literature such as Rajamani [2] and Gillespie [3]. Accounting for the assumptions and simplifications made when considering the modeling of the platform, the longitudinal dynamics are as follows:

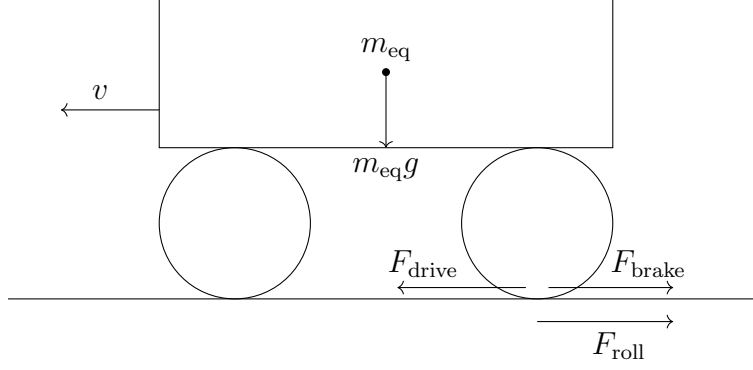


Figure 4.3: Diagram of Forces Acting on Platform along Longitudinal Axis

The above free body diagram is summarized in the following equation:

$$m_{\text{eq}} \dot{v} = \sum F_x \quad (4.1)$$

Where the sum of forces is:

$$\sum F_x = F_{\text{drive}} - F_{\text{brake}} - F_{\text{roll}} \quad (4.2)$$

And the individual forces are given by:

$$F_{\text{drive}} = \frac{\eta_d i_{\text{tot}}}{r_w} T_{\text{motor}} \quad (4.3)$$

$$F_{\text{brake}} = \frac{T_{\text{brake}}}{r_w} \quad (4.4)$$

$$F_{\text{roll}} = C_{rr} m g \quad (4.5)$$

$$m_{\text{eq}} = m + \frac{J_{\text{eq}}}{r_w^2} \quad (4.6)$$

4.4 Lateral Kinematics

The lateral motion of the golf cart is modeled using a planar kinematic bicycle model. The goal of this model is to capture how changes in steering angle translate to motion of the cart in the world frame. As presented in Kong et al. [1], this simplified representation is sufficient at low speeds and for the smooth maneuvers considered in this work.

A world-fixed frame $\{W\}$ with coordinates (x, y) is defined, and a body-fixed frame $\{B\}$ is attached to the golf cart. The state of the lateral model is

$$\mathbf{q} = \begin{bmatrix} x \\ y \\ \psi \end{bmatrix} \quad (4.7)$$

where (x, y) is the position of a reference point on the cart, the center of the rear axle, expressed in $\{W\}$, and ψ is the yaw heading of the cart with respect to $\{W\}$. The inputs to the kinematic model are the longitudinal speed v along the body x -axis and the front wheel steering angle δ .

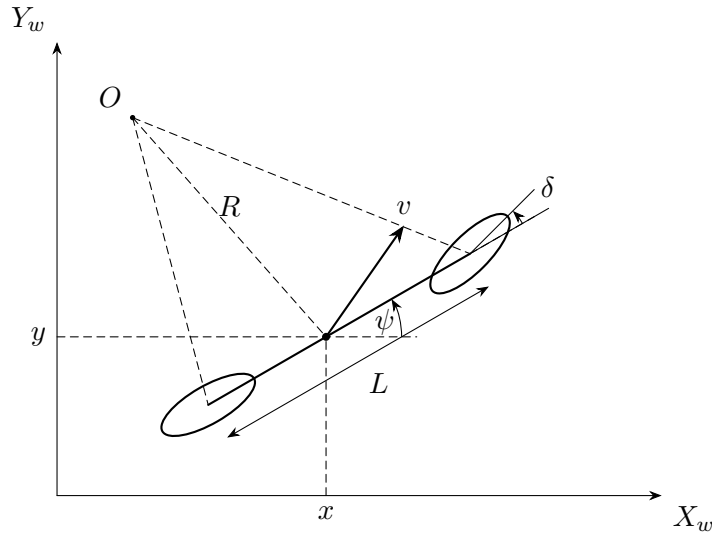


Figure 4.4: Diagram of Bicycle Model

Here, O labels the instantaneous center of rotation, R is the radius of rotation, v is the linear velocity of the cart expressed in $\{W\}$. L is the wheelbase of the platform, and δ is the steering angle expressed in $\{B\}$. Thus, the lateral motion is described by the standard kinematic bicycle equations

$$\dot{x} = v \cos \psi, \quad (4.8)$$

$$\dot{y} = v \sin \psi, \quad (4.9)$$

$$\dot{\psi} = \frac{v}{L} \tan \delta, \quad (4.10)$$

4.5 Combined Model

These two models combined represent the unified set of equations used to model the behavior of the platform. These equations will be used as a basis for designing the parameters of the simulation model, and for designing any necessary controllers. The equations are summarized as the state space model for the platform, and are provided in this section.

Combining the body frame representation (4.7) of the platform with its expression for velocity from the longitudinal dynamics yields the following state vector to describe the motion of the platform with reference to world frame coordinates:

$$\mathbf{x} = \begin{bmatrix} x \\ y \\ \psi \\ v \end{bmatrix} \quad (4.11)$$

The lateral motion is described by the kinematic bicycle model introduced in Section 4.4, with

$$\dot{x} = v \cos \psi, \quad (4.12)$$

$$\dot{y} = v \sin \psi, \quad (4.13)$$

$$\dot{\psi} = \frac{v}{L} \tan \delta, \quad (4.14)$$

where L is the wheelbase and δ is the front wheel steering angle.

The longitudinal dynamics are given by the force balance developed in Section 4.3. Recalling Newton's law in (4.1) with the longitudinal force decomposition in (4.2), the linear acceleration can be described as

$$\dot{v} = \frac{1}{m_{\text{eq}}} (F_{\text{drive}} - F_{\text{brake}} - F_{\text{roll}}), \quad (4.15)$$

where the effective mass m_{eq} and the individual force terms are defined in (4.3)–(4.5).

For later controller design it is convenient to identify the actuator inputs that enter these equations. The motor torque T_{motor} and brake torque T_{brake} determine the drive and brake forces through (4.3) and (4.4), while the steering angle δ appears in the bicycle kinematics in (4.14). These three control signals can be organized into the following control vector \mathbf{u} as

$$\mathbf{u} = \begin{bmatrix} T_{\text{motor}} \\ T_{\text{brake}} \\ \delta \end{bmatrix} \quad (4.16)$$

Using the relationships defined in (4.12)–(4.15), the combined planar model can be expressed in nonlinear state space form as

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) = \begin{bmatrix} v \cos \psi \\ v \sin \psi \\ \frac{v}{L} \tan \delta \\ \frac{\eta_d i_{\text{tot}}}{m_{\text{eq}} r_w} T_{\text{motor}} - \frac{1}{m_{\text{eq}} r_w} T_{\text{brake}} - \frac{C_{rr} m g}{m_{\text{eq}}} \end{bmatrix}. \quad (4.17)$$

Chapter 5

Low-Level Control Design

This chapter presents the low-level controllers used to regulate the golf cart's longitudinal speed and steering angle. The combined planar model in Chapter 4 provides the relationship between motor torque, brake torque, steering angle, and platform motion. That model is used here as the basis for control design.

This control layer maps a speed setpoint $v^*(t)$ and a steering setpoint $\delta^*(t)$ to actuator commands for the traction motor, friction brake, and steering motor. Three controllers are used:

- a drive motor controller that regulates longitudinal speed using motor torque,
- a brake controller that regulates longitudinal speed using friction brake torque,
- a steering controller that regulates the front wheel steering angle.

5.1 Control Objectives and Architecture

The low-level control layer has two main objectives:

1. Track a commanded longitudinal speed $v^*(t)$ in the operating region $0 \leq v \leq 20$ mph using the traction motor and friction brake.
2. Track a commanded steering angle $\delta^*(t)$ of the front axle, consistent with the kinematic bicycle model in Section 4.4.

A high-level controller generates the setpoints $v^*(t)$ and $\delta^*(t)$ based on its internal error minimization calculations. The low-level controllers operate on the tracking errors

$$e_v(t) = v^*(t) - v(t), \quad e_\delta(t) = \delta^*(t) - \delta(t), \quad (5.1)$$

and produce commands for the motor controller, brake actuator, and steering drive.

The longitudinal loop behaves similar to a cruise-control style system, with a speed regulator acting on throttle and brake and a higher-level module providing $v^*(t)$. The steering controller acts as a position servo for δ .

The following control-specific assumptions are made:

- The Kelly KLS motor controller and traction motor are modeled as an approximately linear gain between a normalized motor command and an effective drive torque. The brake actuator is modeled in the same way for brake torque.
- The design is restricted to forward motion; behavior very close to zero speed is handled by a stopped/hold mode in implementation.

5.2 Longitudinal Speed Controller Design

The longitudinal dynamics in Section 4.3 and Section 4.5 can be written as

$$\dot{v} = \frac{\eta_d \dot{i}_{\text{tot}}}{m_{\text{eq}} r_w} T_{\text{motor}} - \frac{1}{m_{\text{eq}} r_w} T_{\text{brake}} - \frac{C_{rr} m g}{m_{\text{eq}}}, \quad (5.2)$$

where T_{motor} and T_{brake} are the effective wheel torques. The rolling resistance term is treated as a constant disturbance for a given operating condition.

The dynamic model can be simplified based on the domain the error signal exists in. Let $u_{\text{th}} \in [0, 1]$ and $u_{\text{br}} \in [0, 1]$ be normalized throttle and brake commands. In the drive domain (brake released) and the brake domain (throttle released), the dynamics are modeled as

$$\dot{v} = g_d u_{\text{th}} - d, \quad (\text{drive domain}), \quad (5.3)$$

$$\dot{v} = -g_b u_{\text{br}} - d, \quad (\text{brake domain}), \quad (5.4)$$

where $g_d > 0$ and $g_b > 0$ are effectively gains and d collects the rolling resistance contribution. The corresponding transfer functions from command to speed (with d neglected in the nominal model) are

$$G_d(s) = \frac{V(s)}{U_{\text{th}}(s)} = \frac{g_d}{s}, \quad G_b(s) = \frac{V(s)}{U_{\text{br}}(s)} = -\frac{g_b}{s}. \quad (5.5)$$

Domain separation between drive and brake is implemented by a simple deadband on the speed error. Let $e_v = v^* - v$. Then

$$T_{\text{motor}}(t) = \begin{cases} T_{\text{drive}}(t), & e_v(t) > \Delta_v, \\ 0, & e_v(t) \leq \Delta_v, \end{cases} \quad T_{\text{brake}}(t) = \begin{cases} T_{\text{brake,cmd}}(t), & e_v(t) < -\Delta_v, \\ 0, & e_v(t) \geq -\Delta_v, \end{cases} \quad (5.6)$$

where Δ_v is a small deadband threshold, and T_{drive} and $T_{\text{brake,cmd}}$ are proportional to u_{th} and u_{br} through the actuator gains.

5.2.1 Drive Motor PI Controller

In the drive domain, the plant is the integrator

$$G_d(s) = \frac{g_d}{s}. \quad (5.7)$$

The drive controller is a PI compensator acting on the error $e_v = v^* - v$ with transfer function

$$C_v(s) = K_{p,v} + \frac{K_{i,v}}{s}. \quad (5.8)$$

With unity feedback, the closed-loop transfer function from v^* to v is

$$\frac{V(s)}{V^*(s)} = \frac{C_v(s)G_d(s)}{1 + C_v(s)G_d(s)} = \frac{g_d(K_{p,v}s + K_{i,v})}{s^2 + g_dK_{p,v}s + g_dK_{i,v}}. \quad (5.9)$$

The denominator in (5.9) defines the closed-loop characteristic polynomial

$$s^2 + g_dK_{p,v}s + g_dK_{i,v} = 0. \quad (5.10)$$

To obtain a desired second-order response, (5.10) is matched to

$$s^2 + 2\zeta_v\omega_{n,v}s + \omega_{n,v}^2 = 0, \quad (5.11)$$

where ζ_v is the damping ratio and $\omega_{n,v}$ is the natural frequency of the speed loop. Equating coefficients gives

$$K_{p,v} = \frac{2\zeta_v\omega_{n,v}}{g_d}, \quad K_{i,v} = \frac{\omega_{n,v}^2}{g_d}. \quad (5.12)$$

Integral action in $C_v(s)$ cancels the steady-state effect of the constant disturbance d in (5.3) for step changes in the speed setpoint.

5.2.2 Brake PI Controller

In the brake domain, the plant is

$$G_b(s) = -\frac{g_b}{s}. \quad (5.13)$$

The brake loop uses a PI controller acting on the error

$$e_b(t) = v(t) - v^*(t), \quad (5.14)$$

which is positive when the vehicle is faster than the setpoint. The controller transfer function is

$$C_b(s) = K_{p,b} + \frac{K_{i,b}}{s}. \quad (5.15)$$

With unity feedback and the sign convention in (5.14), the closed-loop characteristic polynomial again takes the form

$$s^2 + g_b K_{p,b} s + g_b K_{i,b} = 0. \quad (5.16)$$

Matching to (5.11) yields

$$K_{p,b} = \frac{2\zeta_b \omega_{n,b}}{g_b}, \quad K_{i,b} = \frac{\omega_{n,b}^2}{g_b}, \quad (5.17)$$

with $(\zeta_b, \omega_{n,b})$ chosen separately from the drive loop. In practice, braking is tuned to be more heavily damped (larger ζ_b) and sometimes slower (smaller $\omega_{n,b}$) than throttle actuation to improve ride comfort and limit jerk [2, ?].

5.3 Steering Controller Design

The steering controller regulates the front wheel steering angle δ to track a setpoint $\delta^*(t)$ generated by the high-level lateral controller. At the hardware level, the steering column is driven by a DC motor through a motor controller.

The steering mechanism is modeled as a first-order integrator,

$$\dot{\delta}(t) = k_s u_\delta(t), \quad (5.18)$$

where u_δ is a normalized steering command and $k_s > 0$ is an effective gain. The corresponding steering plant transfer function is

$$G_s(s) = \frac{\Delta(s)}{U_\delta(s)} = \frac{k_s}{s}, \quad (5.19)$$

where $\Delta(s)$ and $U_\delta(s)$ are the Laplace transforms of $\delta(t)$ and $u_\delta(t)$, respectively.

5.3.1 Steering PI Controller

The steering error is defined as

$$e_\delta(t) = \delta^*(t) - \delta(t), \quad (5.20)$$

and the controller uses a PI structure with transfer function

$$C_\delta(s) = K_{p,\delta} + \frac{K_{i,\delta}}{s}. \quad (5.21)$$

With unity feedback and plant (5.19), the closed-loop transfer function from δ^* to δ is

$$\frac{\Delta(s)}{\Delta^*(s)} = \frac{C_\delta(s)G_s(s)}{1 + C_\delta(s)G_s(s)} = \frac{k_s(K_{p,\delta}s + K_{i,\delta})}{s^2 + k_sK_{p,\delta}s + k_sK_{i,\delta}}. \quad (5.22)$$

The denominator defines the closed-loop characteristic polynomial

$$s^2 + k_sK_{p,\delta}s + k_sK_{i,\delta} = 0. \quad (5.23)$$

Matching to the standard second-order form (5.11) gives

$$K_{p,\delta} = \frac{2\zeta_\delta\omega_{n,\delta}}{k_s}, \quad K_{i,\delta} = \frac{\omega_{n,\delta}^2}{k_s}, \quad (5.24)$$

where $(\zeta_\delta, \omega_{n,\delta})$ are the damping ratio and natural frequency of the steering loop.

5.4 Summary

This chapter presented the low-level control design for the golf cart platform using the combined planar model in Chapter 4 as a basis. The equations outlined above describe the systems by which the platform translates its desired high level setpoints into motion of the platform by way of the low level control systems, constructed as standard PI controllers, map measured error to actuator input.

Chapter 6

Low Level Control Implementation

[Discuss controller implementation on C2000 leveraging TI hardware and Matlab tools]

6.1 TI c2000 Microcontroller

[Discuss advantages of TI c2000 microcontroller over other mcus]

6.2 MATLAB Toolbox for c2000 Microcontroller

[Discuss MATLAB integration of project]

6.3 Simulink Model

[Discuss derived Simulink Model as realized control system]

Chapter 7

Simulation

[Discuss CARLA sim briefly, Gazebo sim, sim model design]

7.1 CARLA Sim

[Discuss CARLA's poor fit for project]

7.2 Gazebo Sim

[Introduce Gazebo Sim]

7.2.1 Simulation Model

[Discuss URDF]

7.2.2 Simulation Sensors

[Discuss simulated GPS, LiDAR, Encoders]

7.2.3 Simulation Interfaces

[Discuss ROS-GZ Bridge]

Chapter 8

High Level Control Design

[Discuss implementation of GPS waypoint following]

8.1 Jetson Nano

[Describe jetson nano, inputs and outputs]

8.2 ROS2

[Explain ROS2's purpose and role in project]

8.3 GPS Waypoint Line Following

[Derive waypoint line following controller]

Chapter 9

Results

[Discuss results of all stages of project, design, simulation, testing]

Chapter 10

Conclusions

[Summarize project, discuss future work]

Bibliography

- [1] J. Kong, M. Pfeiffer, G. Schildbach, and F. Borrelli, “Kinematic and dynamic vehicle models for autonomous driving control design,” in *Proceedings of the 2015 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2015, pp. 1094–1099.
- [2] R. Rajamani, *Vehicle Dynamics and Control*, 2nd ed., ser. Mechanical Engineering Series. New York, NY: Springer, 2012.
- [3] T. D. Gillespie, *Fundamentals of Vehicle Dynamics*, ser. Premiere Series. Warrendale, PA: SAE International, 1992.

Appendix A

Derivations

[If I need to break out any math, here's where it will go]

Appendix B

Vehicle Parameters

[I might put detailed specs about golfcart here (e.g. mass, dimension)]

Appendix C

Diagrams

[If there are any diagrams that dont fit elsewhere]