Controller Design of A Self-Driving Bike

Qianlei Jia
Electrical Engineering
Chalmers University of Technology
Gothenburg, Sweden
qianlei@chalmers.se

Sarah Stoiber
Electrical Engineering
Chalmers University of Technology
Gothenburg, Sweden
stoiber@chalmers.se

Jonas Sjöberg
Electrical Engineering
Chalmers University of Technology
Gothenburg, Sweden
jonas.sjoberg@chalmers.se

***Abstract*—A control system comprising both lateral and longitudinal controllers is designed for a self-driving bike.**

***Index Terms*—Control system, self-driving bike**

I. Introduction

The AutoBike project is a collaborative effort between Chalmers University of Technology and Mälardalens University, with idustry partners like Volvo Cars, AstaZero, Veoneer, Autoliv and Cycle Europe AB. A large part of the work has been performed in student projects. The project aims at developing autonomous bikes capable of following predefined trajectories with high precision and consistency. Sensor systems in modern vehicles detect and predict the imminent maneuver of a bike. One of the main goals of the project is to carry out repeatable tests to validate automotive safety systems. This serves as the basis for the work presented in this article. The git repository associated with this article is available on GitHub [1].

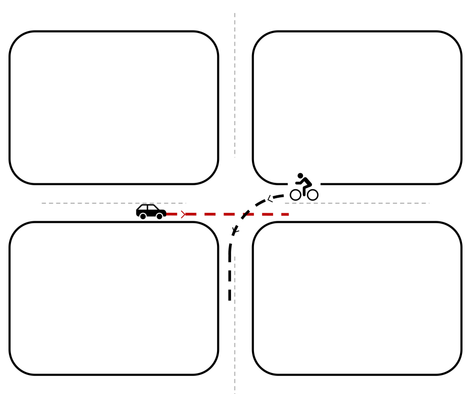

Fig. 1. Potential crash scenario where the bike turns left while the car continues straight.

II. Controller Design

A. Platform

Fig. 2 shows one of the bikes in this project along with hardware components. The state vector of the bike is defined below.

$$X = [x,\, y,\, \psi,\, \varphi,\, \dot{\varphi},\, \delta,\, v] \tag{1}$$

where x and y indicate the position of the bike, ψ is the heading angle, φ is the roll angle, $\dot{\varphi}$ is the roll rate, δ is the steering angle, and v is the velocity.

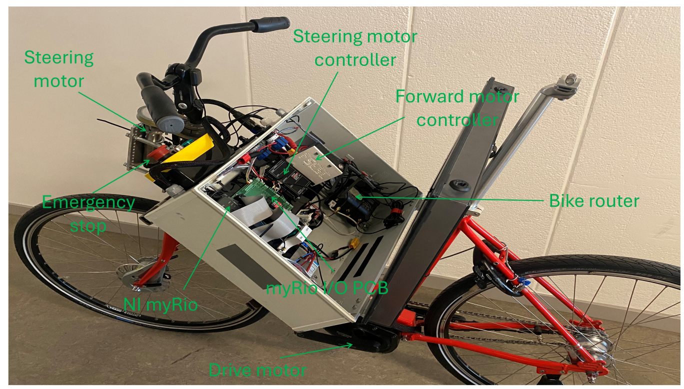

Fig. 2. Red bike.

B. Controller

Fig. 3 is the whole structure of the system.

1) Balancing Controller: The balancing controller takes the roll reference, φ_{ref}, as an input and controls the steering rate, $\dot{\delta}$. It operates as a PD controller. The steering angle reference, δ_{ref}, is converted to a roll reference, φ_{ref}, and is sent from the trajectory controller to the balancing loop. The balancing controller is a PD controller that outputs a steering angular velocity or steering rate, $\dot{\delta}$.

2) Trajectory Controller: The inputs to the trajectory controller are the lateral and heading errors, e_1 and e_2. The geometric illustration of these two errors are shown in Fig. 4. The controller then outputs the reference steering angle. To minimize the lateral and heading errors, an LQR controller is employed. Additionally, a feedforward term is used to enhance the bike's responsiveness to heading changes from the predefined path.

3) Speed Controller: The speed controller is a P controller with an open-loop feedforward term. The feedforward term is introduced to compensate for ground friction. The output is the forward motor's rotational speed, which is then converted to the bike's linear velocity through the bike model.

4) Longitudinal Controller with Position Error: Fig. 5 illustrates how to compute the reference values of X and Y at the current time t_{now}. $X_{ref}(n)$ and $Y_{ref}(n)$ indicate reference values of X and Y that the bike should be at time t_n. $X_{ref}(t_{now})$ and $Y_{ref}(t_{now})$ indicate reference values that the bike should be at t_{now}.

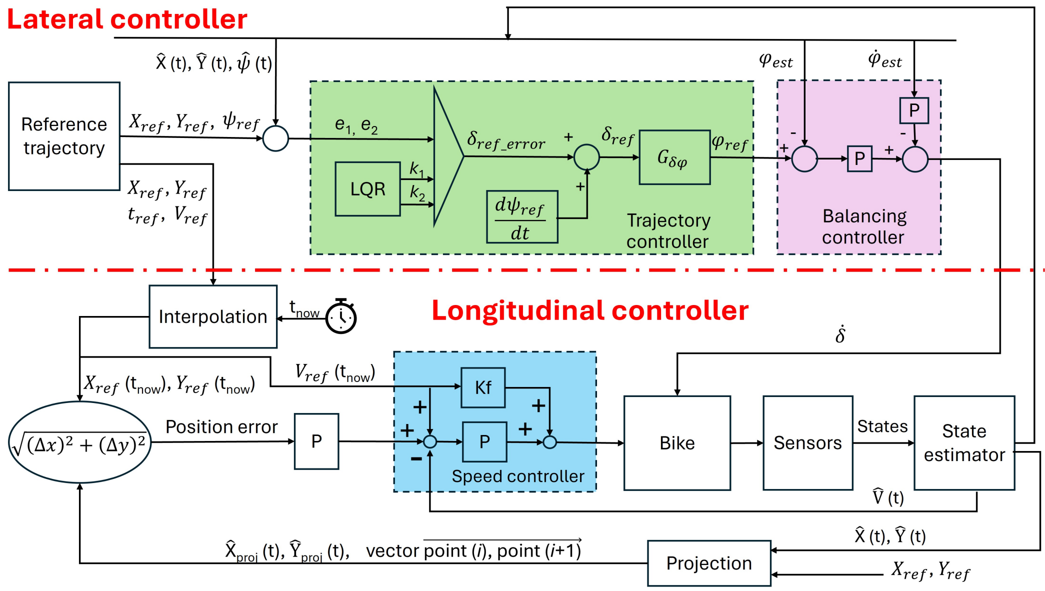

Fig. 3. Whole structure of the controller system.

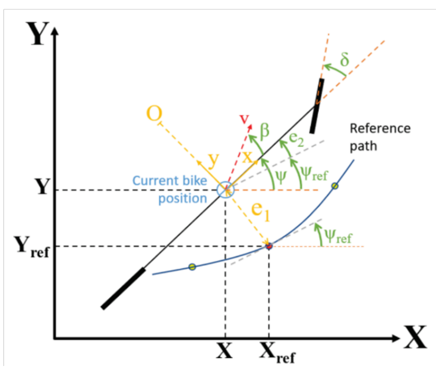

Fig. 4. A top view of the balanced bike is shown, where its current position is at (X, Y), and the reference point is at (X_{ref}, Y_{ref}). The lateral error, e_1, represents the deviation between the bike's position and the reference point, while the heading error, e_2, indicates the angular difference between the bike's current heading and the reference heading at (X_{ref}, Y_{ref}).

The first step is to determine n according to the following inequality.

$$t_n < t_{now} < t_{n+1} \tag{2}$$

Once n is identified, $X_{ref}(n)$, $Y_{ref}(n)$, $X_{ref}(n+1)$, and $Y_{ref}(n+1)$ can be obtained from the reference trajectory. The interpolation is then used to calculate $X_{ref}(t_{now})$ and $Y_{ref}(t_{now})$. The reference speed $V_{ref}(t_{now})$ is also extracted from the reference trajectory and is directly used as an input to the speed controller.

Fig. 6 illustrates how to compute the position error of the bike at t_{now}. The estimated position is projected onto the line segment defined by the closest reference point and its successor along the trajectory. Finally, the position error is calculated with respect to the direction of the reference trajectory, and its sign is determined based on vector projection to indicate whether the position is ahead or behind.

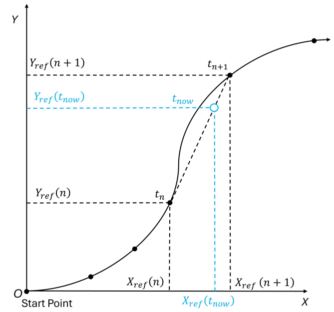

Fig. 5. Illustration of computation for X_{ref} and Y_{ref} at current time.

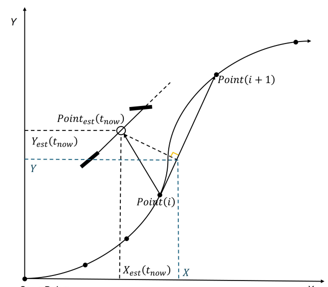

Fig. 6. Illustration of the computation for position error.

References

[1] GitHub, [Online], Available: https://sites.google.com/view/autobikes/project-overview.