Model Matching Control Applied to Bicycles

Kenneth Pasma

Master of Science Thesis

TU Delft Delft University of Technology

Delft Center for Systems and Control
& BioMechanical Engineering

Model Matching Control Applied to Bicycles

Master of Science Thesis

For the degree of Master of Science in Systems and Control and Mechanical Engineering at Delft University of Technology

Kenneth Pasma

August 15, 2024

Faculty of Mechanical Engineering (ME) · Delft University of Technology

Abstract

Model matching control allows for a virtual change in the vehicle's physical parameters, removing the need to alter the system physically. These vehicles are used as a versatile test bed for safe and repeatable experiments, and allow for safer vehicle operation by virtually keeping the driving behaviour constant in the face of physical changes such as wet roads. Although already applied on planes and cars, it has not been applied to bicycles. The proposed control method accomplishes model matching control by equating the state derivative of the physical system to that of the the virtual system. Theoretical analysis shows model matching control is applicable to bicycle dynamics, even if only the steer torque is available for control. Via experiments, I identify the bicycle's weave mode eigenvalue at different speeds, and the Bode gain between steer torque to lean rate and to fork angle for frequencies between one and three Hz. The experimentally identified values follow the theoretically predicted values closely but do not overlap perfectly. The effect of artifacts during the experiment and unmodelled dynamics are analysed with the use of simulations and are able to explain these perceived differences. From the results, this research concludes that model matching control is applicable to bicycles and is able to virtually change the bicycle's physical parameters. Future work will look into making the controller more robust to modelling errors.

Table of Contents

Chapter 1

Introduction

A vehicle's design determines its dynamics. The type of tyre, for example, influences the frictional coefficient and hence influences an automobile's cornering behaviour. Normally these parameters have to be altered physically in order to change the vehicle's dynamics. But through the use of well-placed actuators and appropriately controlling them it is possible to alter the vehicle's physical parameters virtually. In other words, the vehicle is physically unaltered, but dynamically behaves as a vehicle with altered parameters, referred to as the reference vehicle.

Model matching, which this thesis defines as virtually changing the vehicle parameters to match those of another vehicle, allows for different vehicle designs to be applied on a single vehicle without having to perform physical changes. This allows the vehicle to become a test bed, on which concepts are explored further or new ideas are tested in a safe and systematic way. It allows for human-in-the-loop testing to investigate a wide range of vehicle handling qualities, as these vehicles give back realistic visual and motion cues. For aviation, [1, 2] discuss built airplanes that all implement the concept of model matching. They show the vehicle's use case, demonstrating the wide range of experiments model matching can facilitate. For the purpose of research into collision avoidance and human factors research [3] developed an automobile capable of virtually changing its dynamic characteristics by using four-wheel steering and steer-by-wire, a steering actuation method that breaks the mechanical connection between the steer and front wheels and replaces it by an electrical one. In [4] the authors developed a four-wheel steered, steer-by-wire automobile capable of tracking the dynamics of a general nonlinear planer reference model. To demonstrate the controlled vehicle's abilities, they virtually lowered the friction coefficient between the tyres and the road, simulating a slippery road. According to the authors, their research can be used for repeatable experiments in researching driver behaviour when encountering a slippery part on the road.

Unlike physical designs, model matching can adapt to changes to the vehicle or reject disturbances during vehicle operation. This allows the vehicle to be safer to operate by keeping the operating behaviour constant. The control can be made robust to changes in vehicle dynamics or to disturbances. The research in [5] uses a two-degree-of-freedom controller, where

the feedforward is designed to follow a reference model, and an H_∞ controller is used as feedback. Applied to a four-wheel steered automobile, experiments show robustness to changes in road friction. The research in [6] proposes a sliding mode control scheme, designed to follow a reference model. The simulations show disturbance rejection to sudden side winds, and the controller being insensitive to automobile parameter variations. If the change in vehicle parameters is known beforehand, the controller can take this into account, as is done in [7]. There a model matching controller is able to virtually change the front tyre cornering stiffness of an automobile. They argued that loading the vehicle causes a shift in the center of mass. Then If this shift is known, the stiffness can be virtually altered such that the automobile has the same cornering stiffness as in the nominal situation.

Although the past research mainly focuses on applying model matching to cars or planes, bicycles could also gain the same benefits from model matching. Applied to bicycles, designers can use model matching to quickly test out how their design behaves without having to physically build it. It can be used in stores to fine-tune a bicycle to the customer's wishes, or researchers can use it as a test bench, testing out their concept on a wide range of bicycle types. The bicycle can also be dynamically altered to keep the handling the same, as is done with cars. In general, model matching shows potential for improving vehicle behaviour. Applying model matching to bicycles is a step forward in the application of the concept to all kinds of vehicles.

However, as far as the author knows, model matching has not been applied to bicycles yet. Therefore this thesis aims to investigate if the model matching methods applied in literature can be applied to bicycles as well, both in theory and in practice. More specifically, it investigates the method used in [8] where the rate of change of the states between the physical system and reference system are made equal. The limitations of this method when only being able to control the steer input are discussed as well. By doing so this thesis shares insights into what is necessary to achieve model matching for bicycles. Here the focus is on the dynamic behaviour of the uncontrolled bicycle. The haptics and human-machine interaction are left out for future work.

This thesis aims to answer this question:

How can the equations of motion of a bicycle be virtually altered to equal those of a reference bicycle?"

The question will be answered by investigating the method used for cars, as described in [8], in Chapter 2. This chapter applies the method to the dynamic model of a bicycle and investigates the limitations of only having control over steering. It further analyses the effect of different steering actuation methods on the necessary control and presents a design for a model matching controller. Then Chapter 3 presents the performed experiment that tries to validate the designed model matching controller and introduces its results. Chapter 4 analyses the effect of different unmodelled phenomena on the performance of the controller. Afterwards, the results of the experiment and analysis are discussed in Chapter 5 and conclusions are drawn in Chapter 6.

Chapter 2

Design of the Model Matching Controller

2-1 Introduction to the Problem of Model Matching

This section gives a short introduction to the concept of model matching control. The subsections explain this concept through the use of a toy example and show the general problem definition. The selected control method and its application to bicycles will be discussed in the next section.

2-1-1 Example of Virtually Changing a Physical Parameter

When describing a vehicle as a system, the system description includes the physical parameters of that vehicle. In this thesis, these parameters are defined as the physical attributes of a vehicle that directly affect its equations of motion. These are attributes such as the vehicle's

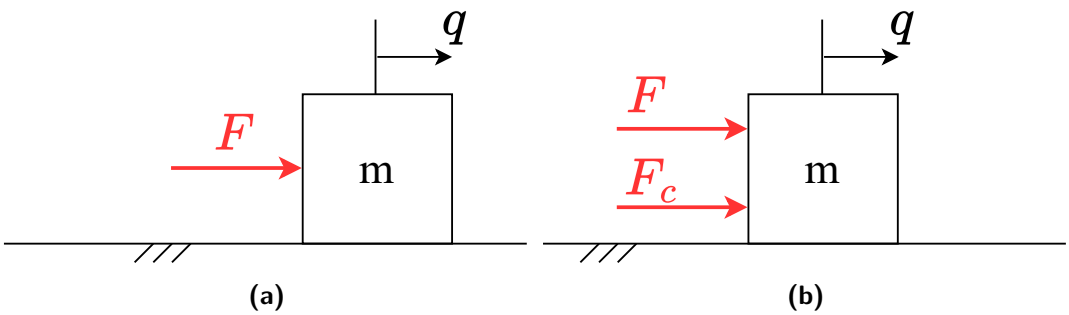

Figure 2-1: Representation of the toy example, a point mass model of a square block on the ground. Acting on the block is a constant force F and the control force F_c. The block has a displacement variable q and a mass m.

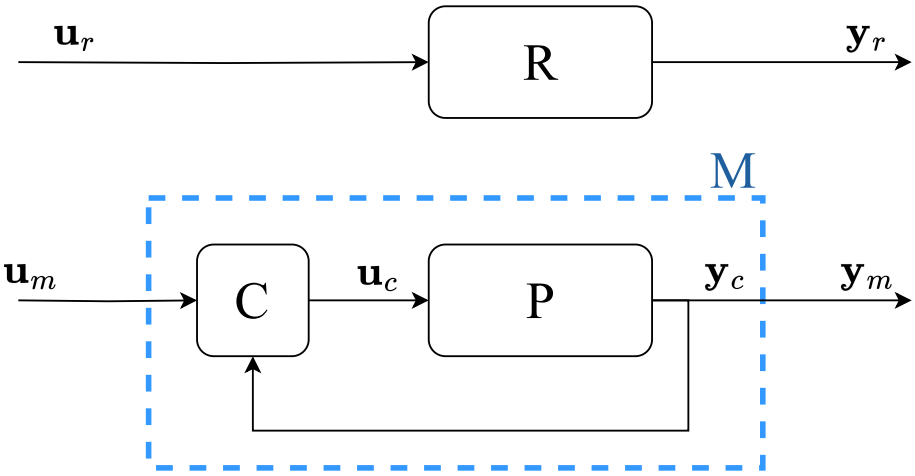

Figure 2-2: Control diagram that represents the general structure of model matching. Here P is the controlled system, R is the reference system, C is the controller, and M is the system that should match the reference through model matching.

mass, inertia, or center of mass. Changing these parameters means changing the system's equations of motion. For example, consider a point mass model of a square block, see Figure 2-1a, It has a displacement variable q and only one physical parameter which is its mass m. Assuming there are no forces acting on the block besides a constant force F, the equation of motion of this block is described by $\ddot{q} = \frac{F}{m}$. It is evident that changing this block's mass parameter will change its motion as the numerical values in the equation of motion differ.

Changing a parameter virtually means that the physical system has to behave as if the virtual change was physical, while in reality not undergoing any physical change. This is only possible by adding additional controlled forces. Consider again the point mass model of a square block in Figure 2-1a. Let m_v and m_p be the virtual and physical value of the mass, leading to the virtual and physical system dynamics $\ddot{q}_v = \frac{F}{m_v}$, and $\ddot{q}_p = \frac{F}{m_p}$ respectively. Without additional forces $\ddot{q}_v = \ddot{q}_p$ if and only if $m_v = m_p$, meaning the virtual change is zero. Let F_c be a controlled force added to the system, see Figure 2-1b. This changes the dynamics of the physical system into $\ddot{q}_p = \frac{F+F_c}{m_p}$. If F_c has a value of $(1 - \frac{m_p}{m_v})F$, then the physical system's dynamics are equal to that of the virtual system, while the physical mass parameter remains unchanged.

2-1-2 General Formulation of the Model Matching Problem

In general, virtually changing the system dynamics can be described by Figure 2-2. Consider the physical system P with input $\mathbf{u}_c(t)$ and output $\mathbf{y}_c(t)$. This system will be referred to as the controlled system. Also, consider the virtual system R with input $\mathbf{u}_r(t)$ and output $\mathbf{y}_r(t)$. This system will be referred to as the reference system. Then define a third system M, which consists of a controller C and the controlled system. It has an input $\mathbf{u}_m(t)$ and an output $\mathbf{y}_m(t)$, which is equal to $\mathbf{y}_c(t)$. These systems share the same dimensions for their input and output vectors respectively.

The goal of virtually changing the system dynamics is to design the controller C such that system M behaves like system R when given the same input. Mathematically this means that

when $\mathbf{u}_m(t) = \mathbf{u}_r(t)$, then $\mathbf{u}_c(t)$ should be chosen such that $\lim_{t\to\infty} \mathbf{y}_m(t) = \mathbf{y}_r(t)$. In this thesis, this problem is defined as the model matching control problem.

2-2 Model Matching Control Applied to Bicycles

The following subsections specify the necessary steps to practically apply model matching to a physical bicycle. They first mathematically present the used control method that achieves model matching control, and the used model that represents the bicycle system. Then the chosen control method is applied to the chosen system model, resulting in a general control law applicable to bicycles.

2-2-1 Selected Control Method for Solving the Model Matching Problem

This thesis applies the specific control method described in [8]. It controls the states of the controlled system such that their rate of change is equal to those of the reference system. This solves the model matching problem using the following logic. If two states change in the same manner and start with the same value, then they end with the same value. To define this mathematically, consider again the controlled system P and the references system R, whose state space representations are respectively given by

$$\begin{cases} \dot{\mathbf{x}}_c = \mathbf{A}_c\mathbf{x}_c + \mathbf{B}_c\mathbf{u}_c \\ \mathbf{y}_c = \mathbf{x}_c \end{cases} \tag{2-1}$$

$$\begin{cases} \dot{\mathbf{x}}_r = \mathbf{A}_r\mathbf{x}_r + \mathbf{B}_r\mathbf{u}_r \\ \mathbf{y}_r = \mathbf{x}_r \end{cases} . \tag{2-2}$$

Here $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{u} \in \mathbb{R}^m$, and $\mathbf{y} \in \mathbb{R}^p$ respectively are the state, input, and output vector, and the subscripts c and r indicate the controlled and reference system respectively. Then the above logic is represented mathematically by

$$(\dot{\mathbf{x}}_c(t) = \dot{\mathbf{x}}_r(t) \quad \forall t \geq 0) \wedge (\mathbf{x}_c(t_1) = \mathbf{x}_r(t_1)) \implies \mathbf{x}_c(t) = \mathbf{x}_r(t) \quad \forall\, t \geq t_1, \tag{2-3}$$

where t_1 is an arbitrary point in time. Assuming that all states can be measured and that there exists a t_1 where $\mathbf{x}_c(t_1) = \mathbf{x}_r(t_1)$, then this method will achieve $\lim_{t\to\infty} \mathbf{y}_c(t) = \mathbf{y}_r(t)$, and thus solve the model matching problem.

To calculate the control law that achieves an equal rate of change of the states, consider the controlled system being in an arbitrary state $\mathbf{x}$. The goal is to let the controlled system have a rate of change as if it were the reference system in that state. Therefore, when calculating the control law, consider both systems to be in the state $\mathbf{x}$. Then, equating $\dot{\mathbf{x}}_c$ to $\dot{\mathbf{x}}_r$ and solving for $\mathbf{u}_c$ gives the control law presented in (2-4).

$$\mathbf{B}_c\mathbf{u}_c = (\mathbf{A}_r - \mathbf{A}_c)\mathbf{x} + \mathbf{B}_r\mathbf{u}_r \tag{2-4}$$

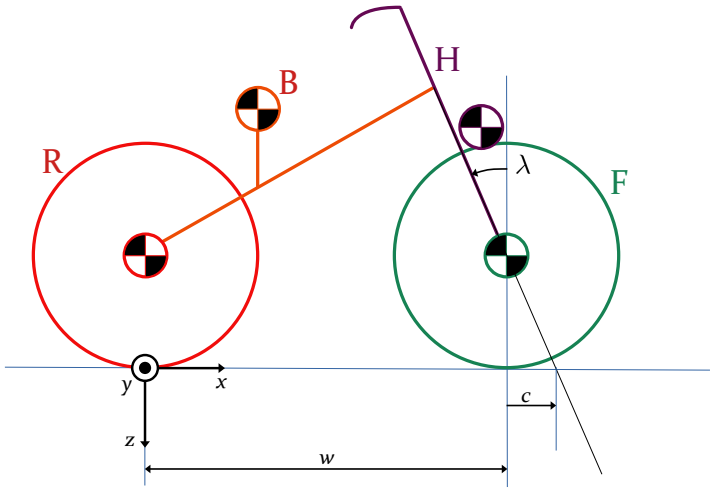

Figure 2-3: The Carvallo-Whipple model, as modelled by [9]. Here R, B, H, and F represent the bodies of the rear wheel, rear frame, front frame, and front wheel respectively. The rear frame includes the mass of the rider. Furthermore, c is the trial defined positive in positive x direction, w is the wheelbase, and λ is the steer axis tilt, being $\pi/2 -$ head angle. The origin of the coordinate frame is at the rear wheel contact point with the ground.

2-2-2 Theoretical System Model of a Bicycle

This thesis uses the Carvallo-Whipple [10, 11] model defined in [9]. See Figure 2-3 for a representation of the model. It consists of four bodies, the rear frame, the front frame, and the rear and front wheel, indicated with B, H, R, and F respectively. The model contains 27 bicycle parameters two of which are gravity and forward velocity and 25 that describe the geometry and inertial properties of the bicycle. The equations of motion have been thoroughly checked and compared to other literature and can describe a wide range of bicycles. It is the simplest model that is able to explain both countersteering and self-stability and has steer and lean torque as control inputs. It is able to closely describe the true bicycle dynamics under normal cycling behaviour [12]. Thus the model is able to accurately differentiate between the dynamic behaviour of two physical bicycles. As such it allows for a controller that lets a human experience a virtually altered bicycle.

The equations of motion of the Carvallo-Whipple model as expressed in [9] are given as

$$\mathbf{M}\ddot{\mathbf{q}} + v\mathbf{C}_1\dot{\mathbf{q}} + \left(g\mathbf{K}_0 + v^2\mathbf{K}_2\right)\mathbf{q} = \mathbf{f} \tag{2-5}$$

Here $\mathbf{M} \in \mathbb{R}^2$ represents the mass matrix, $\mathbf{C}_1 \in \mathbb{R}^2$ represent a damping-like matrix, $\mathbf{K}_0 \in \mathbb{R}^2$ and $\mathbf{K}_2 \in \mathbb{R}^2$ represent two parts of the stiffness matrix. These submatrices are defined according to [9], and contain the bicycle parameters. The indices of the matrices indicate the amount of times the velocity is multiplied with the matrix. Furthermore, $\mathbf{q} = [\,\phi \quad \delta\,]^T$ represents the generalised coordinate vector and contains the lean angle ϕ and steer angle δ, and $\mathbf{f} = [\,T_\phi \quad T_\delta\,]^T$ is the generalised force vector, containing the lean torque T_ϕ and the steer torque T_δ. The lean torque is the torque between the external world and the bicycle around the line between the two ground contact points, and the steer torque is the torque between the rear frame and the front frame around the steer axis.

To transform the equations of motion into a state space representation, the lean rate is defined as $\dot{\phi} \coloneqq \gamma$ and the steer rate as $\dot{\delta} \coloneqq \beta$. The state vector $\mathbf{x} \in \mathbb{R}^4$ then consists of four states, $\mathbf{x} \coloneqq [\,\phi \quad \delta \quad \gamma \quad \beta\,]^T$. The input $\mathbf{u} \in \mathbb{R}^2$ is defined as $\mathbf{u} \coloneqq [\,T_\phi \quad T_\delta\,]^T$. Using these definitions of the state and input vector, (2-5) is rewritten as

$$\begin{aligned}\dot{\mathbf{x}} = \begin{bmatrix}\dot{\phi}\\ \dot{\delta}\\ \dot{\gamma}\\ \dot{\beta}\end{bmatrix} &= \begin{bmatrix}\mathbf{0}_{2\times 2} & \mathbf{I}_2\\ -\mathbf{M}^{-1}(g\mathbf{K}_0 + v^2\mathbf{K}_2) & -\mathbf{M}^{-1}(v\mathbf{C}_1)\end{bmatrix}\begin{bmatrix}\phi\\ \delta\\ \gamma\\ \beta\end{bmatrix} + \begin{bmatrix}\mathbf{0}_{2\times 2}\\ \mathbf{M}^{-1}\end{bmatrix}\begin{bmatrix}T_\phi\\ T_\delta\end{bmatrix}\\ &= \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}.\end{aligned} \tag{2-6}$$

2-2-3 Application of selected Control Method on Theoretical System Model

The solution for the model matching input for bicycles can be found as follows. Let the $\mathbf{A}$ and $\mathbf{B}$ matrix of the Carvallo-Whipple model be redefined as

$$\mathbf{A} = \begin{bmatrix}\mathbf{0}_{2\times 2} & \mathbf{I}_2\\ -\mathbf{M}^{-1}(g\mathbf{K}_0 + v^2\mathbf{K}_2) & -\mathbf{M}^{-1}(v\mathbf{C}_1)\end{bmatrix} \coloneqq \begin{bmatrix}\mathbf{0}_{2\times 2} & \mathbf{I}_2\\ \mathbf{A}_{21} & \mathbf{A}_{22}\end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix}\mathbf{0}_{2\times 2}\\ \mathbf{M}^{-1}\end{bmatrix} \coloneqq \begin{bmatrix}\mathbf{0}_{2\times 2}\\ \mathbf{B}_{21}\end{bmatrix}.$$

Applying these to the solution for the model matching method as given in (2-4), results in a linear matrix equality. This can be seen as a system of four linear equations. Due to the form of the Carvallo-Whipple model, the first two of these linear equations lead to the trivial equation $0 = 0$. As a result, only two equations remain, and the general solution for finding the correct model matching control input is given by

$$\begin{bmatrix}\mathbf{B}_{21,c}\end{bmatrix}\begin{bmatrix}T_{\phi,c}\\ T_{\delta,c}\end{bmatrix} = \begin{bmatrix}\mathbf{A}_{21,r} - \mathbf{A}_{21,c} & \mathbf{A}_{22,r} - \mathbf{A}_{22,c}\end{bmatrix}\begin{bmatrix}\phi\\ \delta\\ \gamma\\ \beta\end{bmatrix} + \begin{bmatrix}\mathbf{B}_{21,r}\end{bmatrix}\begin{bmatrix}T_{\phi,r}\\ T_{\delta,r}\end{bmatrix}. \tag{2-7}$$

Then the solution for the control torques can be found by multiplying both sides with $\mathbf{M} \coloneqq \mathbf{B}_{21}^{-1}$.

2-3 Theoretical Analysis of Model Matching Control Applied on Bicycles

Having derived a general control law for bicycles, this section looks deeper into the practical application of model matching control on bicycles. As the lean torque input is difficult to use for control, this section shows the effects of being able to only control the steer torque, and it compares two different actuation methods to apply this steer torque. Furthermore, this section also investigates the resulting controller design, looking into the stability of the controller, into the existence and uniqueness of the control input, and compares it with a conventional pole placement design method.

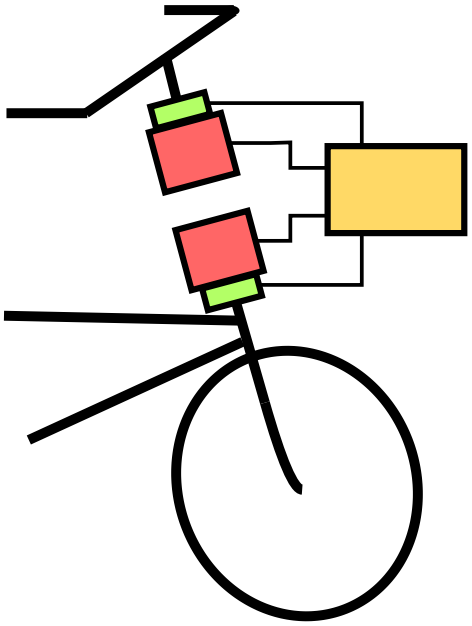

Figure 2-4: Schematic overview of steer-by-wire applied to a bicycle. The controller is indicated in orange, the motor in red, and the rotary encoder in green.

2-3-1 Realistic Application of Model Matching Control on Bicycles

System Model of a Steer-by-Wire bicycle

X-by-wire, which also includes steer-by-wire, is a concept that breaks the mechanical connection between two parts and replaces it by an electrical one. This is done by a sensor measuring the intention of one part, an actuator relaying this measured intention to the other part, and a controller coordinating these actions. Here X stands for the function that is affected. Figure 2-4 shows a schematic of steer-by-wire applied to a bicycle. The head tube connecting the handlebar with the front fork is cut in two. There are two motor and rotary encoder pairs, one attached to the handlebar, and the other to the front fork. The controller allows for communication between these pairs.

It is possible to extend the Carvallo-Whipple model discussed in Section 2-2-2 to include the extra degree of freedom gained by mechanically decoupling the handlebar from the front fork, as is done in [13]. However, for model matching control only the front wheel angle is of concern. as it actively influences the bicycle movement. This is not the case anymore for now the mechanically decoupled handlebar. Therefore the bicycle used in the experiments discussed in the next chapter will be mathematically described by the Carvallo-Whipple model, even though physically it is a steer-by-wire bicycle.

Choice of Steer Actuation: Steer-by-Wire vs. Steer Assist

Ideally, the forces felt by the human driving the controlled bicycle coincide with the forces that would be felt when driving the reference bicycle. This would allow the human to experience

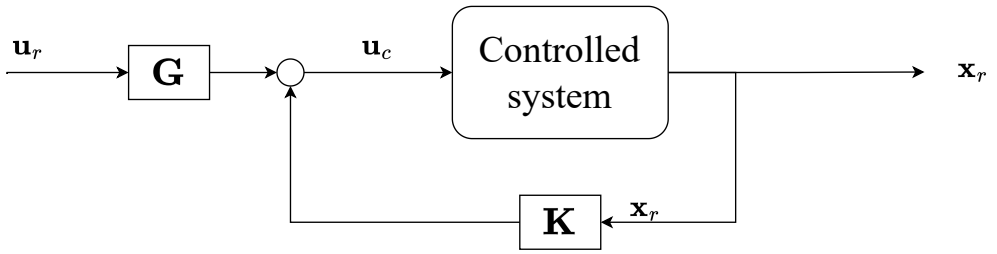

Figure 2-5: A control diagram depicting the structure of the model matching controller. Here $\mathbf{u}_c$ is the control input, $\mathbf{u}_r$ is the reference input, $\mathbf{x}_c$ the state of the controlled system, and $\mathbf{K}$ and $\mathbf{G}$ are respectively the feedforward and feedback matrices.

the controlled bicycle as if it were the reference bicycle. The ability to give the correct force feedback depends on the steer actuation method implemented on the bicycle. Two methods are discussed in this section, being steer assist and steer-by-wire. Steer assist is an actuation method that supplies torque parallel to the user input. Both the user and actuator simultaneously apply torque to the steer.

Consider again the equation for calculating the model matching input given in (2-4). Assuming this has a solution for $\mathbf{u}_c$, the model matching control input then takes the form of

$$\mathbf{u}_c = \mathbf{K}\mathbf{x} + \mathbf{G}\mathbf{u}_r, \tag{2-8}$$

where $\mathbf{K}$ is the matrix containing the state feedback gains, and $\mathbf{G}$ the matrix containing the feedforward gains. Figure 2-5 shows a control diagram of this structure. This form shows that the controlled input differs from the reference input if $\mathbf{G}$ is not the identity matrix. As a result the necessity to implement steer-by-wire as an actuation method becomes dependent on the type of the control and reference input.

This is because the control and reference input are interpreted differently. The reference input $\mathbf{u}_r$ contains the external inputs acting on the bicycle. These are the external inputs that would also control a non-actuated regular bicycle, and act as a reference for the controller. The control input $\mathbf{u}_c$ contains the inputs that act directly on the controlled system. These are the controlled internal inputs from the actuators, but can also include the external inputs when there is a direct mechanical coupling between the application point and the controlled system.

If these inputs refer to a position, e.g. the angle of the front wheel, then model matching is only possible if steer-by-wire is implemented as the actuation method. Take for example a bicycle. Here the controller sets a position for the front wheel to achieve model matching. Because the input is a position, it is not possible for the reference input and the controlled input to be different at the same time if the steer and front wheel are mechanically linked. The operator will have to go against the controlled position when giving its reference position. As a result, the operator will therefore not experience an uncontrolled vehicle. Therefore, steer-by-wire is necessary to allow the human input to be taken into account by the controller while allowing it to be different from the controlled input.

If the inputs refer to a torque, e.g. the steer torque, then model matching is possible with both actuation methods. Look again at the bicycle, and assume the steer and front wheel are mechanically linked. Now the controlled input and the reference input can differ, while

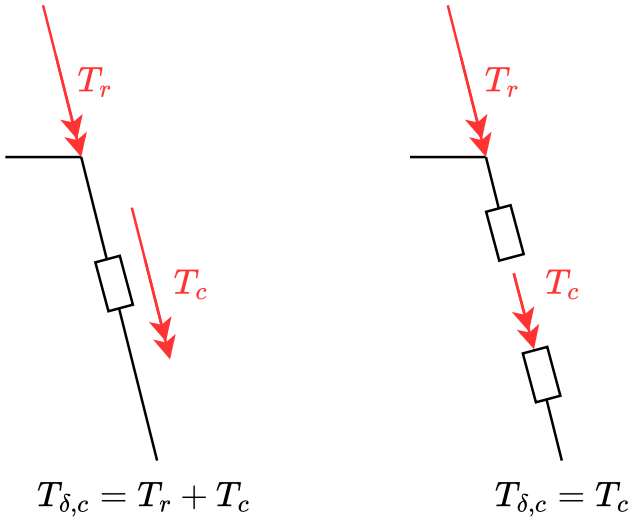

Figure 2-6: Visualization of the applied forces on the steering assembly, showing the need for different control to achieve equal system input u. Left is the steering assembly with a motor parallel to the system, called steer assist. Right is the steer-by-wire steering system. Here T_r is the external human input and T_c is the torque exerted by the controlled motor.

not denying each other's existence. When the different torques are applied at the same time, they will simply be added on top of each other. The resulting torque on the steering system will then give the correct steer input that leads to the same state trajectory as the reference system. The torques felt by the operator would be the resultant torques that the reference system also would have had.

In this thesis, the Carvallo-Whipple model is used for modelling the controlled bicycle. This model uses torque as an input type. Therefore, for bicycles both steer assist and steer-by-wire can be used for model matching. Note that the chosen method does influence the controller, see Figure 2-6. For the steer assist both the operator and the controller contribute to the resultant torque. Therefore, the controller has to take this into account by subtracting the operator torque from its calculated control torque. For steer-by-wire, the controlled torque is the only torque influencing the front wheel and thus can be directly applied without any alterations.

However, steer assist cannot achieve a more complete form of the experience of virtually riding a reference bicycle. So far model matching was applied to match the states of the Carvallo-Whipple model. These are the bicycle's lean and steer angles, and their rates. In a vacuum, matching these states will lead to an equal human experience. However, bicycles move over the ground, having a trajectory that depends on the heading angle and rate. These also contribute to the operator's experience of riding a bicycle. Intuitively it is the difference between taking a corner on a tandem and on a normal bicycle while having the same lean and steer angle, and their rates.

The kinematic equation of a bicycle with steer assist prevents it from matching this cornering

experience. Taken from [9],

$$\dot{\psi} = \frac{v\cos\lambda}{w}\delta + \frac{c\cos\lambda}{w}\dot{\delta} := a\delta + b\dot{\delta}, \tag{2-9}$$

represents the kinematic equations of the heading rate. Here $\dot{\psi}$ is the heading rate, c, λ, and w are trail, steer axis tilt and wheelbase respectively. Note that the number before the steer angle and rate have been abbreviated to a and b respectively. If these geometric parameters c, λ, and w are virtually altered, and model matching of the states is achieved, the heading rate of the reference and control model will still differ. Mathematically this is given by

$$\begin{matrix}\text{(virtual change)}\\ \text{(model matching)}\end{matrix}\begin{cases} a_c \neq a_r & \vee & b_c \neq b_r \\ \delta_c = \delta_r & \wedge & \dot{\delta}_c = \dot{\delta}_r \end{cases} \implies \dot{\psi}_c \neq \dot{\psi}_r \tag{2-10}$$

The inability to have both equal steer angle and rate and equal heading rate is because the heading rate can not be controlled directly.

Future research can investigate if steer-by-wire can create the illusion of both matching the states and the heading. Here the reference lean angle, and lean and heading rate are matched by the use of steer and lean torque. The front wheel will then not match the reference model's steer angle and rate, but control of the mechanically separated handlebar can create the illusion of the correct steer angle and rate. As long as the human does not look at the front wheel, it experiences the bicycle as if it were the reference bicycle.

Limitations of Solely Using Steer Actuation

Practically, it is not trivial to control the lean torque. This torque is between the ground and the bicycle and must be applied while the bicycle is moving. A solution is to implement for example extending training wheels, a reaction wheel, or a controlled moment gyroscope. However, implementing a method to give a controlled lean torque is not part of this thesis. As a result, only the steer torque is available for control. The following paragraphs show how to calculate the model matching control when only the steer torque is a control input and show the effect of this limitation on the choice of the reference model.

Being able to only control the steer torque changes the interpretation of $T_{\phi,c}$. Consider again the different interpretations of the inputs $\mathbf{u}_c$ and $\mathbf{u}_r$. Here $\mathbf{u}_r$ contains the external torques and $\mathbf{u}_c$ contains both the internal and possibly the external torques. As the bicycle used in the experiments cannot generate a controlled lean torque, and the external lean torque directly acts on the controlled system, the resultant lean torque acting on the controlled system is equal to the externally applied lean torque. Therefore $T_{\phi,c} = T_{\phi,r}$.

To calculate the model matching control input for the steer torque, the matrices in (2-7) are split into column vectors of dimension $\mathbb{R}^{2\times 1}$, defined as

$$\begin{aligned} \mathbf{B}_{21,i} &:= \begin{bmatrix}\mathbf{B}_{\phi,i} & \mathbf{B}_{\delta,i}\end{bmatrix} \\ \mathbf{A}_{21,r} - \mathbf{A}_{21,c} &:= \begin{bmatrix}\mathbf{A}_{\phi} & \mathbf{A}_{\delta}\end{bmatrix} \\ \mathbf{A}_{22,r} - \mathbf{A}_{22,c} &:= \begin{bmatrix}\mathbf{A}_{\gamma} & \mathbf{A}_{\beta}\end{bmatrix}. \end{aligned}$$

Rewriting the equation such that only the controlled steer torque remains on the left-hand side results in

$$\mathbf{B}_{\delta,c}T_{\delta,c} = \mathbf{A}_{\phi}\phi + \mathbf{A}_{\delta}\delta + \mathbf{A}_{\gamma}\gamma + \mathbf{A}_{\beta}\beta + (\mathbf{B}_{\phi,r} - \mathbf{B}_{\phi,c})T_{\phi,r} + \mathbf{B}_{\delta,r}T_{\delta,r}. \tag{2-11}$$

In this system of equations, the single controlled variable $T_{\delta,c}$ has to be chosen such that it suffices both the first and second row of the matrix equation. Since the values of ϕ,δ,γ,β, $T_{\phi,r}$, and $T_{\delta,r}$ can be arbitrary, the equation holds if and only if all vectors are linearly dependent on each other. In other words, the ratios between the first element and the second element in the vector should be equal for all vectors. This is given by

$$\begin{aligned} &\frac{\mathbf{A}_{\phi}(1)}{\mathbf{A}_{\phi}(2)} = \frac{\mathbf{B}_{\delta,c}(1)}{\mathbf{B}_{\delta,c}(2)}, \quad \frac{\mathbf{A}_{\delta}(1)}{\mathbf{A}_{\delta}(2)} = \frac{\mathbf{B}_{\delta,c}(1)}{\mathbf{B}_{\delta,c}(2)}, \quad \frac{\mathbf{A}_{\gamma}(1)}{\mathbf{A}_{\gamma}(2)} = \frac{\mathbf{B}_{\delta,c}(1)}{\mathbf{B}_{\delta,c}(2)} \\ &\frac{\mathbf{A}_{\beta}(1)}{\mathbf{A}_{\beta}(2)} = \frac{\mathbf{B}_{\delta,c}(1)}{\mathbf{B}_{\delta,c}(2)}, \quad \frac{\mathbf{B}_{\phi,r}(1)}{\mathbf{B}_{\phi,r}(2)} = \frac{\mathbf{B}_{\delta,c}(1)}{\mathbf{B}_{\delta,c}(2)}, \quad \frac{\mathbf{B}_{\delta,r}(1)}{\mathbf{B}_{\delta,r}(2)} = \frac{\mathbf{B}_{\delta,c}(1)}{\mathbf{B}_{\delta,c}(2)}, \end{aligned} \tag{2-12}$$

where $\mathbf{A}_{\bullet}(i)$ is the i element in the column vector. Here the first element is indicated by a one.

Consequently, it is not possible to freely choose any reference model. The restriction of linear dependency results in a system of six equations given by (2-12). Here the scalar values of the nominators and denominators consist of the elements of the bicycle model's **A** and **B** matrix for both the controlled and reference bicycle. Therefore the nominators and denominators are values consisting of the controlled and reference bicycle's physical parameters. For the system of equations in (2-12) to hold, the values of the physical parameters have to be chosen accordingly. The values of the physical parameters of the controlled system are set, and therefore the parameters of the reference system become the variables to solve for. As there are six equations, only six physical parameters of the reference bicycle have to be chosen as variables to solve for. These will depend on the remaining physical parameters of the reference bicycle and the physical parameters of the controlled bicycle. For this thesis the dependent parameters are I_{Ryy}, I_{Bxx}, I_{Bxz}, I_{Fyy}, and z_b. For this specific set of parameters, the system of equations was not completely linearly independent, resulting in only five dependent parameters being sufficient to solve it.

This restriction can make the reference model physically infeasible, as not every combination of bicycle parameters is physically possible. In most cases, the intended reference model is a physically feasible bicycle design. However, for model matching control to work, the dependent model parameters change to uphold the restriction described above. These changed parameters can make the bicycle infeasible, as physically the parameters depend on each other. For example, changing only the front wheel radius alters the bicycle's head tube angle, trail, and front wheel inertia and mass. Therefore, changing the dependent parameter while keeping the rest constant leads to an infeasible bicycle design.

Finding a feasible reference model via iteration is a tedious problem. It is much easier to find the solution for the limitation above by taking the feasibility into account. Taking the feasibility of the bicycle's physical parameters into account can be achieved by translating the physical relations into constraints and using an optimization to find feasible parameter values, as is done in [14]. However, implementing this method is seen as out of scope and physically infeasible bikes are allowed.

2-3-2 Basic Control Theoretic Analysis of the Model Matching Controller

Stability of the Model Matching Controller

The stability of the error between controlled and reference model states depends on the reference model. The model matching method requires that the states of the reference and controlled model are equal at some point in time. Consider the controlled and reference model, and the model matching control law given in (2-1), (2-2), and (2-4) respectively. Then applying the control law to the controlled system, and using x_c for x leads to

$$\begin{aligned}\dot{\mathbf{x}}_r - \dot{\mathbf{x}}_c &= \mathbf{A}_r\mathbf{x}_r + \mathbf{B}_r\mathbf{u}_r - (\mathbf{A}_c\mathbf{x}_c + \mathbf{B}_c\mathbf{u}_c) \\ &= \mathbf{A}_r\mathbf{x}_r + \mathbf{B}_r\mathbf{u}_r - (\mathbf{A}_c\mathbf{x}_c + ((\mathbf{A}_r - \mathbf{A}_c)\mathbf{x}_c + \mathbf{B}_r\mathbf{u}_r)) \\ &= \mathbf{A}_r(\mathbf{x}_r - \mathbf{x}_c) \\ \dot{\mathbf{e}} &= \mathbf{A}_r\mathbf{e}.\end{aligned}$$

The error will asymptotically go to zero if $\mathbf{A}_r$ is stable. For bicycles, $\mathbf{A}_r$ is dependent on forward velocity and is generally only stable between a specific speed range.

Still, the error will be zero despite a possible unstable $\mathbf{A}_r$. When the controller is turned on, the initial conditions of the reference model are set as the measured states of the controlled system. This is possible as the reference model is only conceptually present. There is no physical system or simulation of it running alongside the controlled bicycle. It is only used to calculate the controller gains. Then, with the initial error equal to zero, it will stay equal to zero.

This is, however, in the ideal case. Disturbances and modelling errors can lead to a nonzero error. While a stable $\mathbf{A}_r$ can counteract a temporary disturbance, lasting disturbance and modelling errors will prevent the controlled system from behaving like the reference system. To accommodate for these errors, either the model should be improved, or the controller should be made more robust as is done in [15].

Existence of Solution and Uniqueness of the Model Matching Controller

Using a reference model that adheres to the system of six equations resulting from (2-11), $T_{\delta,c}$ can be found by using either this first or second row of the matrix equation. This then has the form shown in (2-13), where f_i and g_i are scalar constants. (2-13) is a single linear scalar equation with one variable to solve for, being $T_{\delta,c}$. This type of equation always has a single unique solution.

$$T_{\delta,c} = f_1\phi + f_2\delta + f_3\gamma + f_4\beta + g_1T_{\phi,r} + g_2T_{\delta,r} \tag{2-13}$$

Similarity of Model Matching Control to Pole Placement

The resulting control of the model matching method is in essence equal to a pole placement feedback with a static feedforward gain. The feedback gain that achieves pole placement is not unique for systems with multiple inputs. To make it unique, an extra constraint is necessary. The goal of model matching requires an equal state trajectory between the

controlled system and reference system for equal conditions. Therefore the constraint requires both the eigenvalues and eigenvectors to be matched, which model matching achieves. Thus regarding the autonomous behaviour, model matching is a subset of the possible solutions for pole placement.

In terms of the static feedforward gain, consider the general solution given in (2-4). If $\mathbf{B}_c$ is full rank, a (pseudo-)inverse can be used, resulting in $\mathbf{u}_c = \mathbf{B}_c^{-1}\mathbf{B}_r\mathbf{u}_r$ achieving $\mathbf{B}_c\mathbf{u}_c = \mathbf{B}_r\mathbf{u}_r$. If $\mathbf{B}_c$ is not full rank, the model matching method shows that it is necessary for the columns of $\mathbf{B}_c$ and $\mathbf{B}_r$ to be linearly dependent on each other in order for model matching to be possible. This means a static gain matrix, which scales and swaps the columns of $\mathbf{B}_c$ is enough for $\mathbf{B}_c\mathbf{u}_c = \mathbf{B}_r\mathbf{u}_r$ to hold.

The model matching method above has multiple advantages over using a general pole placement method. If $\mathbf{B}_c$ is rank deficient, not all reference models can be matched, as seen above. As a natural result of using the model matching method, the restrictions on which reference models can be matched will become evident. Furthermore, a generic pole placement method does not guarantee the correct feedback gain. To guarantee this, a requirement of having equal eigenvectors is necessary. The model matching method has this requirement built in.

Chapter 3

Experimental Validation of the Model Matching Controller

3-1 Test setup

This section presents the experimental test setup. It discusses the system models used to create the model matching controller, and the hardware and software needed to perform the experiment.

3-1-1 Bicycle Parameters of the Controlled and Reference System

The model matching controller design uses both the model of the controlled system and the model of the reference system to calculate the controller gains. To acquire these values, I measured the physical parameters of the controlled bicycle, with the exception of I_{Bxx}, I_{Bzz}, I_{Bxz}, I_{Hxx}, I_{Hzz}, and I_{Hxz}. These parameters were estimated using measurement data from [16], which is licensed under Creative Commons Attribution 3.0 Unported License. Table 3-1 shows the measured bicycle parameters in addition to the estimated maximum measurement error. This error is a range in which the measured parameter is highly likely to be in, and is based on an educated guess taking into account the precision of the measurement method. The measurement methods are explained further in Appendix A.

Table 3-1 additionally shows the bicycle parameters of the reference model used in the experiment. To prevent the reference from being a highly unfeasible bicycle, the parameters are chosen to be as similar to the controlled bike as possible, while still causing a noticeable difference in the bicycle's behaviour. The research in [17], which investigated the sensitivity of the stable speed range to the bicycle parameters, gave a selection of bicycle parameters with a large influence on the stable speed range of an uncontrolled bicycle. Simulations of the effect of a change in a single parameter on the bicycle system showed that changing the trail was the most effective for changing the bicycle dynamics. Therefore, only the trail is actively altered. However, the restrictions of only having control over the steer torque cause

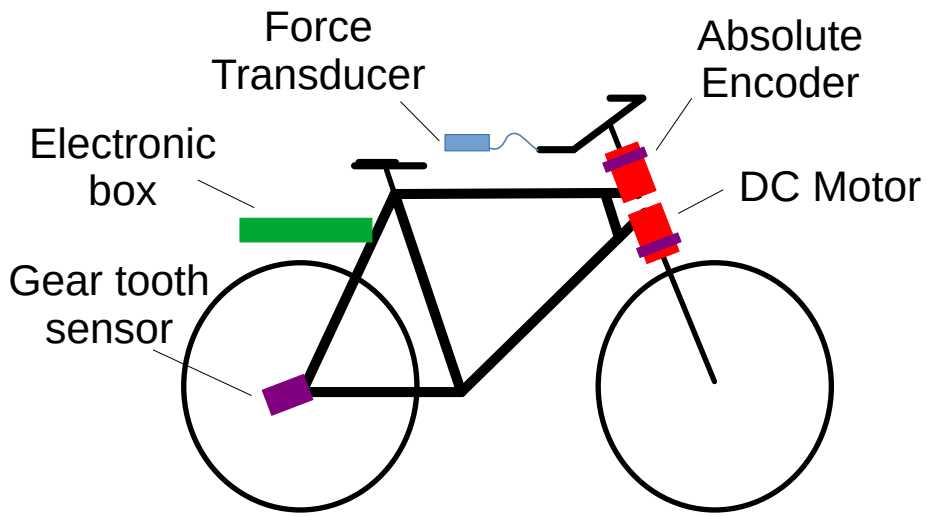

Figure 3-1: Overview of all the important mechatronic systems on the steer-by-wire bicycle

the dependent parameters to be different from the controlled bicycle as well, as discussed in Section 2-3-1. The resulting set of parameters is shown in the table.

3-1-2 Hardware of the Test Setup

The experimental test setup consists of a treadmill and a steer-by-wire bicycle described in [18], who designed and built the bicycle. The treadmill allows for a controlled constant speed with the bicycle riding in place. A steer-by-wire bicycle is used as an electrical steered bicycle on which the model matching controller can be validated. Figure 3-1 shows the placement of all important mechatronic parts on the steer-by-wire bicycle. The steering system is actuated by two Maxon EC45 Flat brushless DC motors attached to the handlebar and front fork, respectively. These are each controlled by a Maxon ESCON 50/5 motor driver. Two RMB20CS absolute encoders measure the steer and front fork angle. On the rear wheel, a GTS35 gear tooth sensor measures the wheel's rotational speed, which is used to calculate the forward velocity. A box for electronics is placed on the luggage carrier. Inside a MPU9250 Inertial Measurement Unit (IMU) measures the roll, pitch, and yaw rate of the bicycle. All sensors and drivers are connected to the Teensy 4.1 microcontroller. The software on the Teensy runs the control loop at a frequency of 100Hz.

A force transducer measures the external steer torque exerted. While riding the bicycle this would be placed on the steer. The microcontroller converts the external pushing force to an external steer torque input. However, the experiments use a riderless bicycle. Therefore the force transducer is not attached to the bicycle to prevent the experimenter's input torque to be a torque that acts between the ground and the front assembly. Such a torque would not qualify as a steer torque input according to the Carvallo-Whipple model, as here it is defined between the rear frame and front assembly, see the left figure in Figure 3-2.

When the transducer would be on the steer and a pure torque is given, it would not affect the rest of the bicycle due to the mechanical decoupling of the steer-by-wire system. However, the decoupled handlebar gives no resistance to movement other than its inertia, thus having large

Table 3-1: Values of the Bicycle parameter for the controlled and reference bicycle. The physical parameters are defined according to [9]. The parameters of the controlled bicycle were measured, with the exception of I_{Bxx}, I_{Bzz}, I_{Bxz}, I_{Hxx}, I_{Hzz}, and I_{Hxz}. These parameters were estimated using measurement data from [16], which is licensed under Creative Commons Attribution 3.0 Unported License. An educated estimation of the maximum measurement error is given with each parameter, see Appendix A for more details.
For the reference bicycle, only the trail is actively changed, the dependent parameters are calculated according to the discussion in Section 2-3-1. The dependent variables are I_{Ryy}, I_{Bxx}, I_{Bxz}, I_{Fyy}, and z_B. The changed variables are highlighted.

Parameter	Controlled Bicycle	Controlled Bicycle's Measurement Error	Reference Bicycle	Unit
w	1.036	$\pm$ 0.01	1.036	m
c	0.0803	$\pm$ 0.01	0.0763	m
λ	18.2	$\pm$ 1	18.2	$^\circ$
g	9.81		9.81	ms^{-2}
v	-		-	ms^{-1}
r_R	0.3498	$\pm$ 0.005	0.3498	m
m_R	10.12	$\pm$ 0.1	10.12	kg
I_{Rxx}	0.1040	$\pm$ 20%	0.1040	kgm^2
I_{Ryy}	0.1641	$\pm$ 20%	0.2618	kgm^2
x_B	0.462	$\pm$ 0.05	0.462	m
z_B	-0.698	$\pm$ 0.05	-0.6980	m
m_B	20.9	$\pm$ 0.6	20.9	kg
I_{Bxx}	1.64	[-0.64, 1.36]	1.640	kgm^2
I_{Bzz}	1.94	[-0.94,1.06]	1.94	kgm^2
I_{Bxz}	0.654	[-1.154, 0.346]	1.040	kgm^2
I_{Byy}	-		-	kgm^2
x_H	0.944	$\pm$ 0.05	0.944	m
z_H	-0.595	$\pm$ 0.05	-0.595	m
m_H	0.6	$\pm$ 0.6	0.6	kg
I_{Hxx}	0.00980	[-0.00980, 0.0902]	0.00980	kgm^2
I_{Hzz}	0.00396	[-0.00396, 0.09604]	0.00396	kgm^2
I_{Hxz}	-0.00044	[-0.00956, 0.01044]	-0.00044	kgm^2
I_{Hyy}	-		-	kgm^2
r_F	0.3498	$\pm$ 0.005	0.3498	m
m_F	1.780	$\pm$ 0.1	1.780	kg
I_{Fxx}	0.0644	$\pm$ 20%	0.0644	kgm^2
I_{Fyy}	0.1289	$\pm$ 20%	0.02594	kgm^2

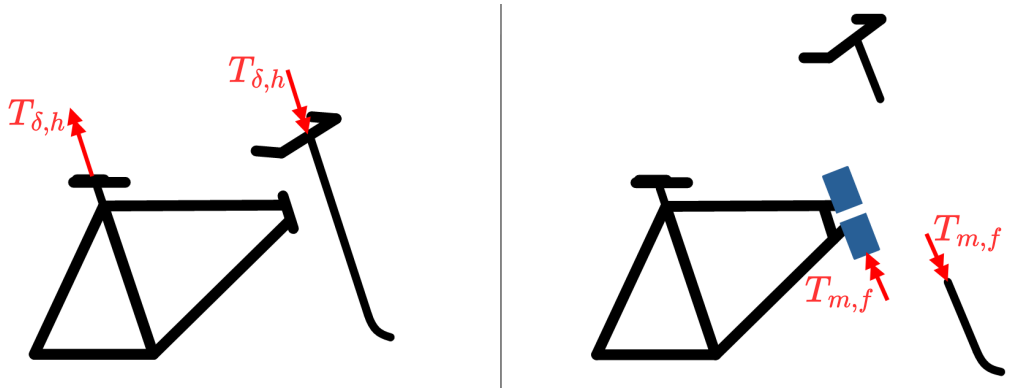

Figure 3-2: Representation of the input steer torque T_δ for a normal bicycle (left) and a steer-by-wire bicycle (right). For the steer-by-wire bicycle the command torque for the fork motor is given via an external input device. The handlebar motor is not used.

angular displacement for small forces. This prevents the experiment from giving a constant torque input or a high torque input. Therefore the transducer is used as an external input device and the experimenter does not apply forces directly on the bicycle. The measured force is sensed and translated into a motor torque command in accordance with the model matching method. This gives an input to the front fork that acts on both the front assembly and the rear frame, see the right figure in Figure 3-2. Additionally, it also prevents other unwanted moments from acting on the bicycle, as it is difficult for the experimenter to give a pure steer torque while standing next to the bicycle.

3-1-3 Software of the Test Setup

Not all variables are measured directly, therefore the microcontroller calculates these online. It calculates the front fork steer rate via first-order numerical differentiation. It calculates the lean angle according to the method described in [19]. This paper uses a filter that in structure is equal to a Kalman filter. It consists of a prediction step using a state transition model and an update step making use of a measurement. In [19] the state transition model describes the integration of the measured roll rate that additionally has a sensor bias. The measurement is a calculation of the lean angle using the measured forward and angular speeds. After these calculations of the lean angle and front fork rate, the bicycle's states and input are transmitted via a USB cable to an external computer, where they are logged in a csv format for future analysis.

Besides the model matching controller, the microcontroller also runs a steer-into-lean controller, described in [20]. This controller, as the name suggests, steers in the direction of the lean, thereby preventing the bicycle from falling. It increases the stable speed range, such that the bicycle is also stable at lower speeds, thereby overlapping with the operating speeds of the treadmill. Therefore this controller allows me to perform experiments while the bicycle is at a stable speed, which is necessary for the weave mode eigenvalue identification described in Section 3-2.

3-2 Experimental procedure for Validating the Model Matching Controller

The purpose of the experiment is to identify the specific dynamics of the controlled bicycle in both the neutral state and in the active state. In the neutral state, the bicycle only runs the steer-into-lean controller, and the bicycle acts according to its real dynamics. In the active state, the bicycle runs the model matching controller in addition to the steer-into-lean controller and should have the same dynamics as the chosen reference model with a steer-into-lean controller. The identification of the dynamics in both the neutral and active state is done with two different experiments. These are a perturbation experiment in the stable speed region identifying the eigenvalues belonging to the weave mode, and an input-to-output experiment identifying the bicycle's Bode gain plots. The Bode gain plots are identified for only a single speed and are from steer torque to the lean rate and steer torque to the steer angle. The measurements taken from these experiments are stored on the external computer that is connected to the bicycle via a USB cable.

At the start of an experiment, the bicycle is placed on the treadmill having a level ground surface. The tyre pressure is brought to 6 bar. A safety harness attaches the bicycle to the ceiling to prevent damage in the case of a crash. No use is made of the handlebar in the experiments, and it is therefore fixated to the rear frame to prevent it from freely swinging. The bicycle is brought up to the target speed while the experimenter holds the bicycle upright. Since the bicycle is almost standing still with respect to the air, there is also no air drag. Once the target speed is reached, the experimenter lets go of the bicycle.

In the case of the eigenvalue identification, the experimenter gives an impulse disturbance by hand at the saddle in the positive or negative lateral direction, respectively being right or left when looking in the direction of velocity. The experimenter catches the bicycle in the event it rides off the side of the treadmill, otherwise the experimenter catches the bicycle after the oscillations from the initial impulse have visibly died out. The experimenter gives this impulse disturbance several times at each speed, and the speeds are increased by steps of 0.5 m/s within the range of 1.5 to 5 m/s.

In the case of the Bode gain identification, the bicycle is riding at a constant speed of 4 m/s. The experimenter gives a single frequency sinusoidal input by hand to a force transducer that is only electronically connected to the bicycle via a wire. The sinusoidal input is given for frequencies between 1 and 3 Hz, where between 1 and 2 Hz increasing steps of 0.1 Hz are used, and between 2 and 3 Hz increasing steps of 0.2 Hz are used. The measured force is translated to a steer torque and used as an input to the bicycle system. For the bicycle in the neutral state, the measured steer torque is relayed one-to-one to the front fork. In the active state, the measured steer torque is used as an input for the model matching controller, which controls the front fork motor accordingly.

In the experiment as well as in the simulation in the next chapter, there are no external forces or torques exerted on the bicycle during riding except for the initial impulse meant to mimic an initial lean rate. There is also no air resistance, as the bicycle is almost standing still with respect to the air. The only forces acting according to the definition of $T_{\delta,r}$ and $T_{\phi,r}$ are the motor torques of the steer-by-wire system. Therefore, for the control input calculation of the model matching controller, it is assumed that there is no lean torque, i.e. $T_{\phi,r} = 0$, and that $T_{\delta,r}$ is equal to the torque given by the human via the force transducer.

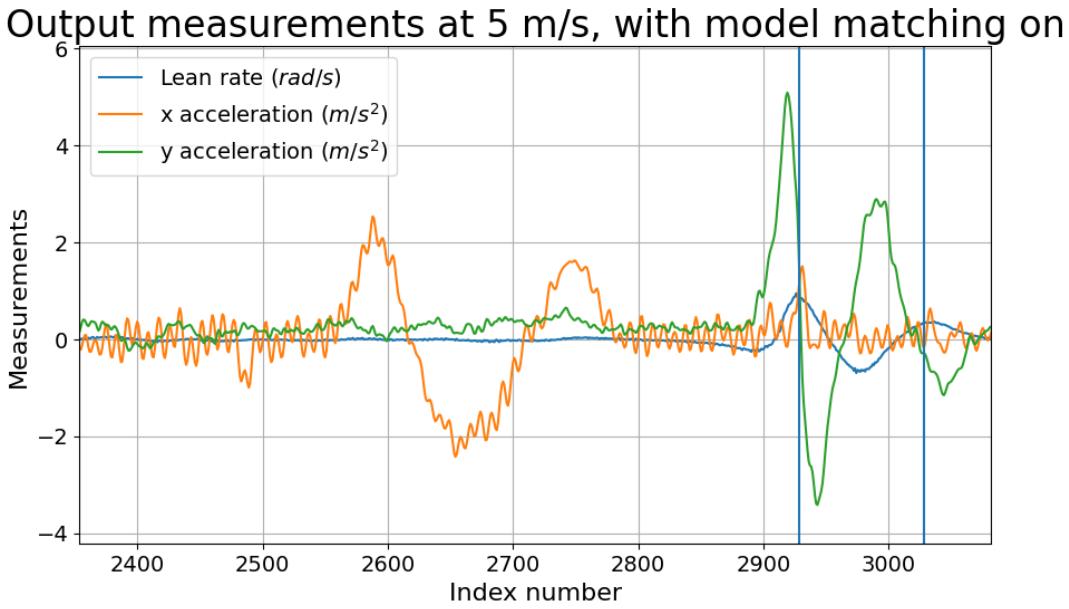

Figure 3-3: The measurements of the impulse response. Here the vertical lines indicate the data set used for further analysis. The time between index numbers is 0.01 s

3-3 Methods Used for Analysing the Experimental Data

The method described in [12] retrieves the weave mode eigenvalues from the measured data. Here they perturb a bicycle with a lateral impulse to the rear frame, while the bicycle is coasting freely at a certain speed. The impulse response is measured and used for the identification of the weave mode eigenvalue. They assume the weave mode is captured by an exponentially decaying or increasing sinusoid described by

$$q(t) = c_1 + e^{\sigma t}(c_2 sin(\omega t) + c_3 cos(\omega t)), \tag{3-1}$$

where σ and ω are the real and imaginary part of the eigenvalue. Additionally, c_1, c_2, and c_3 are parameters that describe the possible signal bias and weight of the sine and cosine functions, respectively. A nonlinear least squares optimization with parameters σ, ω, and c_1 till c_3, fits this function to the impulse response and returns the eigenvalue of the weave mode.

This thesis uses an equal analysis. First, the impulse responses are isolated from the measured data of the experiment. For this, the experimenter moved the bicycle forward and back on the treadmill before each impulse perturbation. The longitudinal and lateral acceleration are then measured to serve as data markers. Here the lateral acceleration indicates the application of the impulse. Figure 3-3 shows the raw data around one single impulse. Here the orange and the green are the longitudinal and lateral acceleration, and the blue line is the measured lean rate. The figure shows that the forward and backward movement is visible in the longitudinal acceleration. It is assumed that the impulse response starts around the point the lateral acceleration switches signs, as this indicates there is no longer an applied force accelerating the bicycle in the lateral direction.

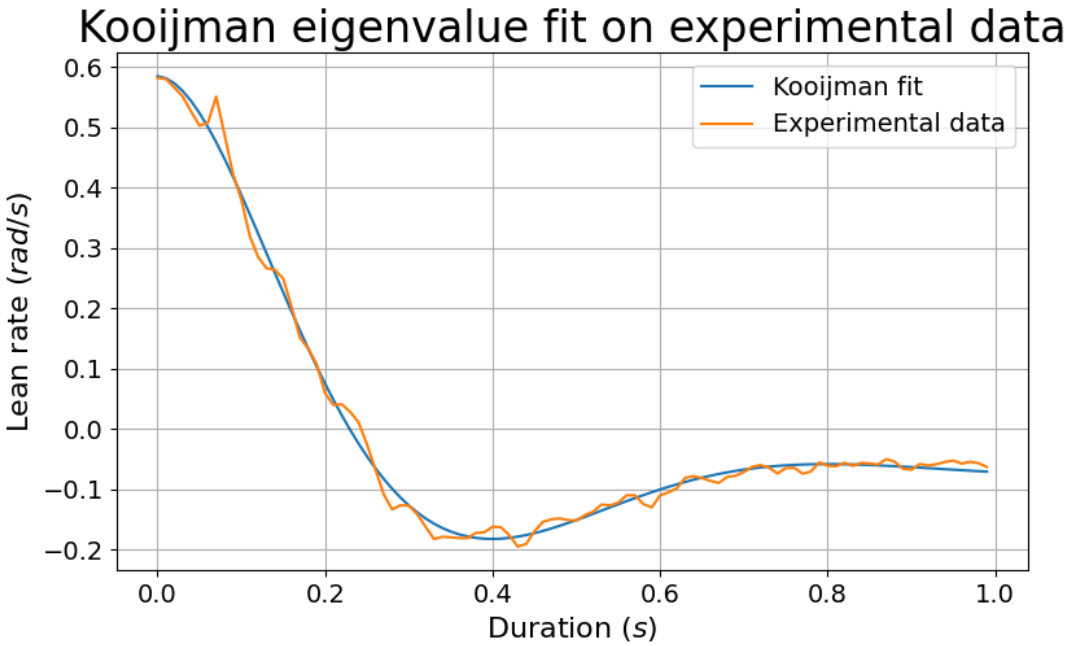

Figure 3-4: Results of the nonlinear fit, identifying the weave mode eigenvalue

Using the data markers, the data is cut to extract the impulse response of the lean rate. The chosen section for analysis is indicated with vertical lines in Figure 3-3. The analysis only uses the first second of the impulse response, as most trials are noticeably affected by unmodelled dynamics after this time period. Then the analysis fits (3-1) to the lean rate to extract the eigenvalues, see Figure 3-4. This is repeated for all disturbances, for all speeds to create the plots seen in Section 3-4.

For the Bode gain identification, the gains are identified by dividing the magnitude of the output signal by that of the input signal. Figure 3-5 shows the raw data of the experiment. Here the green line is the steer torque input, and the blue and orange lines are the lean rate and fork angle outputs respectively. The data is cut to contain an integer amount of periods. This is necessary to get a good transformation from the time domain to the frequency domain. The first few oscillations after the input starts, show a transition from no input to a sinusoidal input. This transition period is left out of the cut data, as it does not accurately represent the gain. The cut section is indicated by the vertical lines.

The fast Fourier transform algorithm transforms the input and output measurements to the frequency domain. Here the frequency of the input is seen as the frequency that has the highest magnitude. Some signals have a maximal magnitude at a frequency of zero, which results from the time domain signal oscillating around an offset. This offset is not the intended input signal. Assuming linearity of the system, the magnitudes at zero frequency are ignored as they do not influence the response to the sinusoidal input. Figure 3-6 shows an example of one cut of the input signal in the frequency domain. Dividing the output magnitude by the input magnitude at the input frequency gives the gain of the system at that frequency. Repeating for different input frequencies gives the Bode gain plot seen in Section 3-4. Here, only the lean rate and fork angle are chosen as the output. This is because the lean angle or

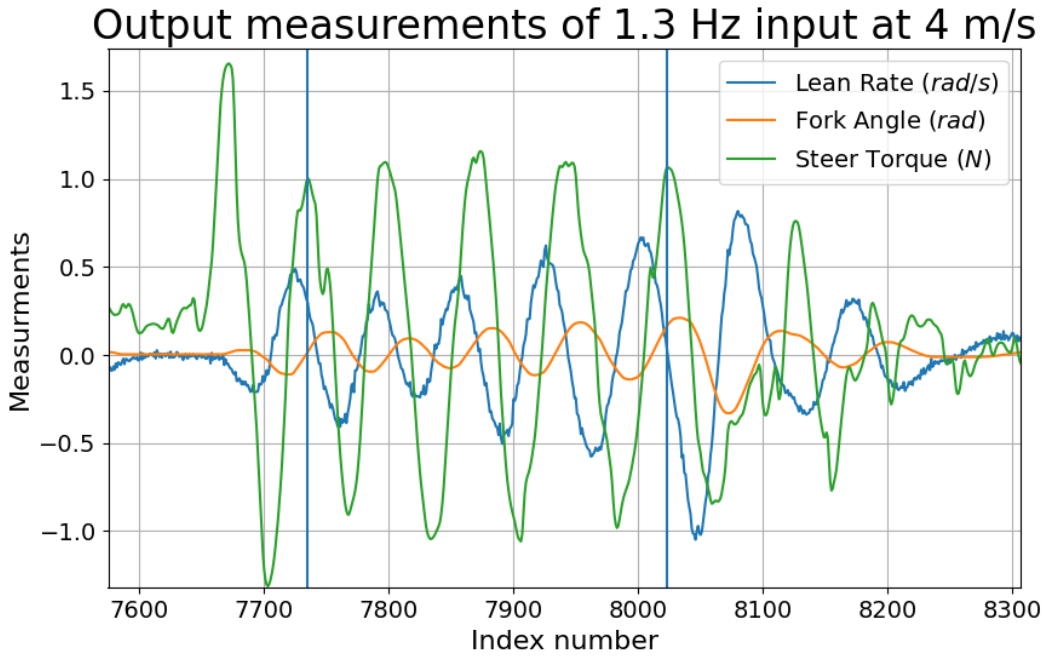

Figure 3-5: The measurements of the sinusoidal steer torque input and the resulting response. Here the vertical lines indicate the data set used for further analysis. The time between index numbers is 0.01 s

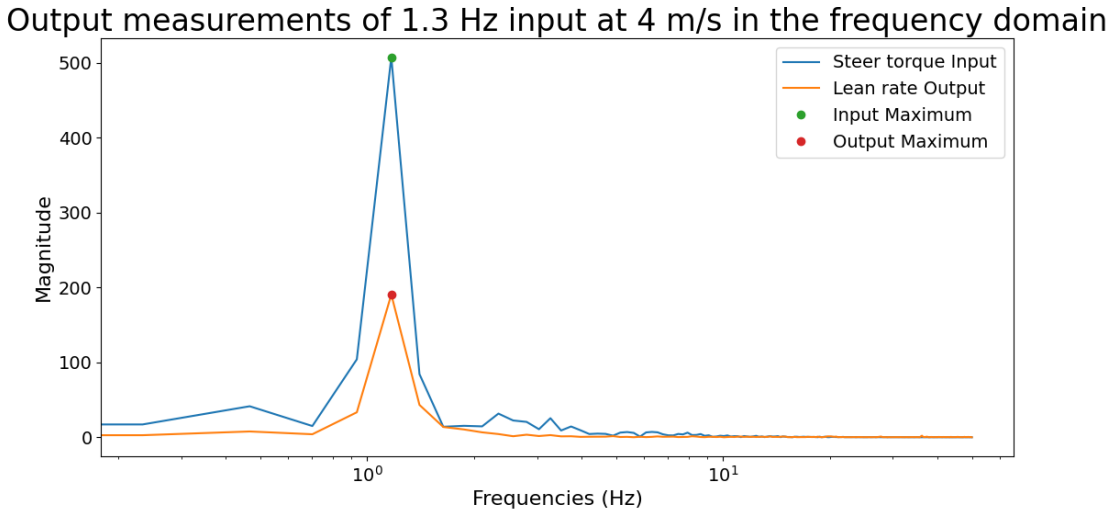

Figure 3-6: Results of applying the fast Fourier transform on the cut data. The dots indicate the maximum values of the input and output used to calculate the Bode gain.

fork rate is respectively the integral or derivative. As the input signal is a single frequency sinusoid, the magnitude is equal and will result in the same plot.

3-4 Results of the Experiment

The analysis of the data is done for both the neutral state, where the model matching controller is 'off', and the active state, where the controller is 'on'. The results are shown in Figure 3-7, 3-8, and 3-9. Here the experiments with the model matching controller 'off' should match the controlled system, and with the model matching controller 'on' should match the reference system. The theoretical values of the controlled and reference system are calculated directly from their respective models. The eigenvalues of the weave mode are calculated using the **A** matrix of the state space model. The Bode gain is calculated using the system's equivalent transfer function.

Figure 3-7 and 3-8 show the Bode gain plot of the bicycle at 4 m/s. For legibility, the identified Bode gains for the model matching controller being 'on' or 'off' are shown separately in the plot on the left and right respectively. The theoretical Bode gains for the controlled and reference system are depicted as the blue solid line and orange dashed line respectively, and are visible on both sides. The markers represent the resulting data points of the different trials. The figures show that for both the model matching controller 'on' and 'off' the data points follow the same trend as the theoretical bode gain plot. The vertical spread on the data points is larger when the model matching controller is 'on'. Figure 3-7 shows that the data points mostly lie on top of their respective theoretical counterpart for steer torque to lean rate. Figure 3-8 shows that the data points are systematically slightly below their respective theoretic counterpart for steer torque to fork angle.

Figure 3-9 depicts a speed-eigenvalue plot, where the x-axis is the speed, and the y-axis the speed-dependent eigenvalues of a bicycle. The theoretical eigenvalues for both the controlled and reference system are depicted with the blue and orange lines. The experimentally identified weave mode eigenvalues are indicated by the triangular markers. Each marker represents a different trial. For both the real and imaginary part of the weave eigenvalue the data points are closely clustered with the exception of the data points at 1.5 m/s. In the real part of the eigenvalues, the data points are mostly on top of the theoretical values when the model matching controller is 'off'. When it is 'on', the data points follow the same trend, but do not perfectly overlap the theoretical line. They are slightly below the theoretical value. For the imaginary part, both for the model matching controller 'on' and 'off' the data points follow the trend of their theoretical counterpart. This is most visible in the slope after 3 m/s. An exception to this is the data points at 1.5 and 2 m/s.

Further analysis into the inconsistent eigenvalues at 1.5 and 2 m/s requires a closer look at the raw experimental data. Figure 3-10 and Figure 3-11 show the time response of the lean rate from which the eigenvalues are identified for both the model matching controller off and on. Here the different colours are the different trials. The plot starts at the moment the peak of the impulse is given. As mentioned in Section 3-3, the eigenvalues are identified using the first second after the impulse is applied. The blue vertical line indicates this cut-off. The black dashed line is the corresponding theoretical impulse response. The figures also show the impulse response at speeds of 3.5 and 5 m/s for comparison. The figure shows that the

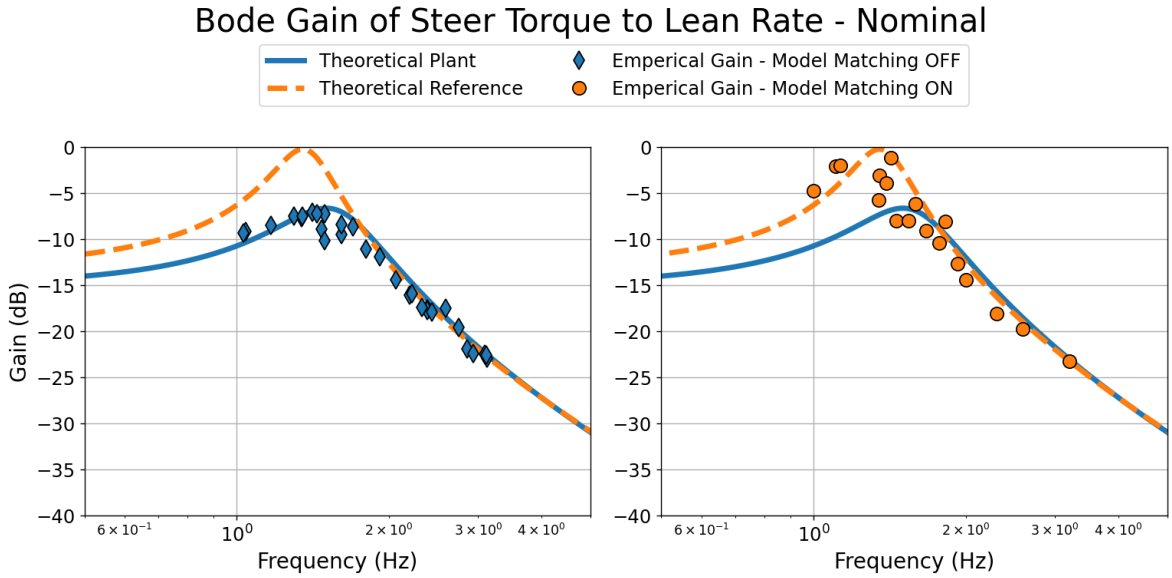

Figure 3-7: Bode gain plot of the controlled system, referred to in the legend as plant, and the reference system. The markers represent the results of the data analysis from the experiment with model matching controller 'off' (Blue diamonds) and 'on' (orange circles). Here the input was the steer torque and the output was the lean rate. The markers should match the lines with the corresponding colour.

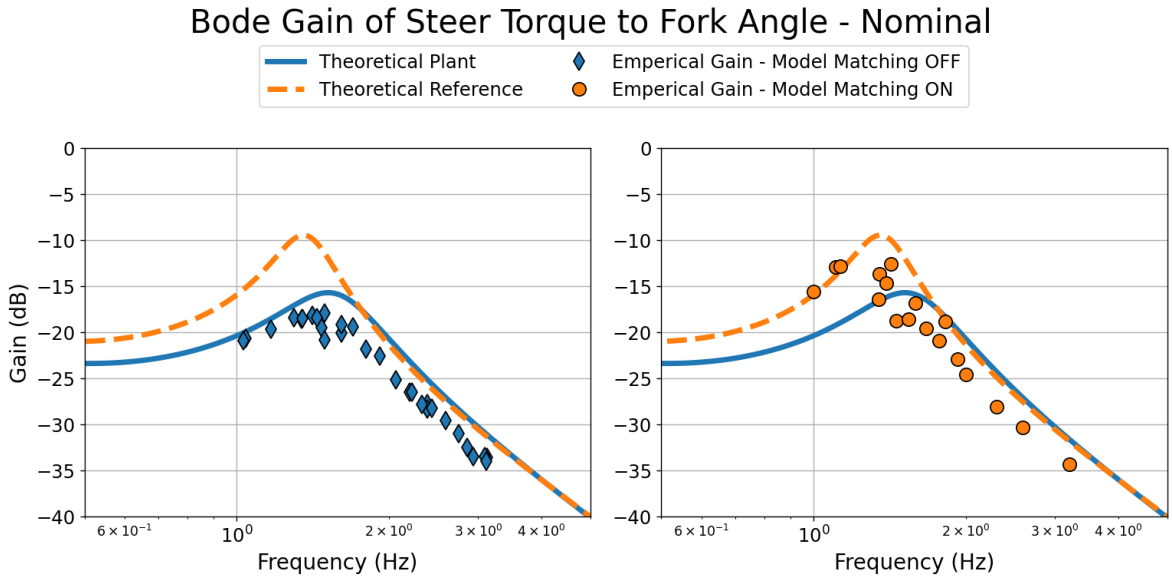

Figure 3-8: Bode gain plot of the controlled system, referred to in the legend as plant, and the reference system. The markers represent the results of the data analysis from the experiment with model matching controller 'off' (blue diamonds) and 'on' (orange circles). Here the input was the steer torque and the output was the fork angle. The markers should match the lines with the corresponding colour.

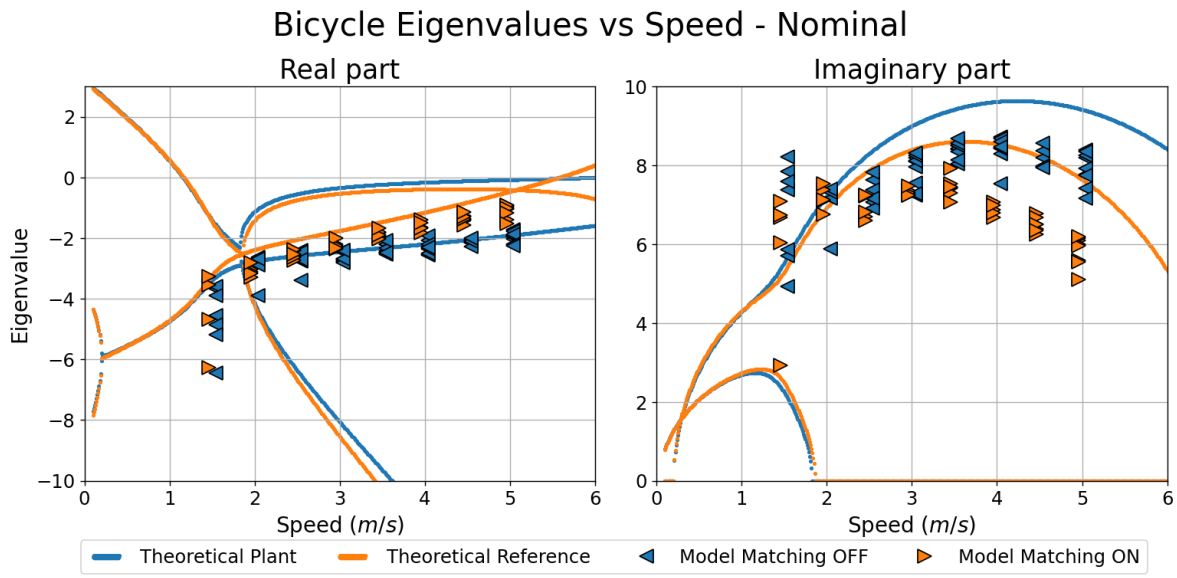

Figure 3-9: Speed-eigenvalue plot of the controlled system, referred to in the legend as plant, and the reference system. The markers represent the results of the data analysis from the experiment with model matching controller 'off' (blue left-facing triangle) and 'on' (orange right-facing triangle). The markers should match the lines with the corresponding colour.

impulse response at 1 m/s does show the general shape of a decaying sinusoide but does not make a full oscillation. This makes it difficult to correctly fit (3-1) to the measured data, and explains the larger spread in the identified values.

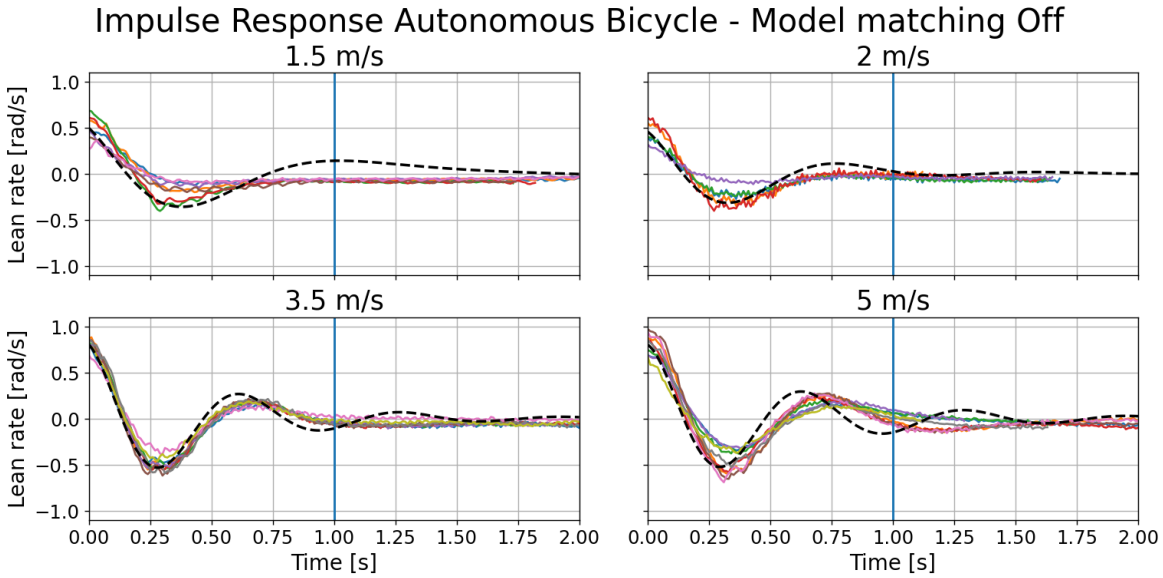

Figure 3-10: time response of the lean rate from which the eigenvalues are identified. Here the model matching controller is off. The different colours represent the different trials. The plot starts at the moment the peak of the impulse is given. The blue vertical line indicates up to where the data is used for the eigenvalue analysis. The dashed black line represents the theoretical impulse response.

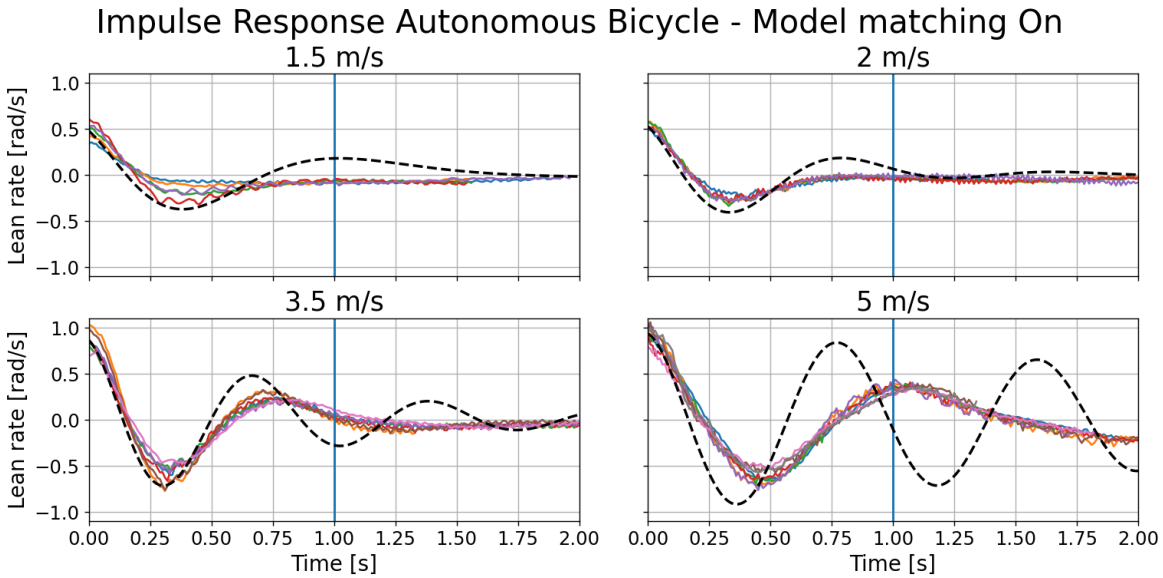

Figure 3-11: time response of the lean rate from which the eigenvalues are identified. Here the model matching controller is on. The different colours represent the different trials. The plot starts at the moment the peak of the impulse is given. The blue vertical line indicates up to where the data is used for the eigenvalue analysis. The dashed black line represents the theoretical impulse response.

Chapter 4

Post-Experimental Analysis of Unmodelled Dynamics

The experiments identified the controlled bicycle with only steer-into-lean control in the neutral case, and the bicycle with steer-into-lean and model matching control in the active case. To predict the identified values for the speed-eigenvalue and Bode gain plot, a model of the controlled bicycle is used for the neutral case. In the active case, the reference model is used, as applying the model matching controller to the bicycle ideally results in the controlled bicycle completely matching the dynamics of the reference model. As seen in the experimental results, the identified values follow their respective theoretically predicted values, but they do not perfectly match them. In reality, there are multiple unmodelled phenomena that govern the real dynamics of a controlled bicycle. These are not included in the model of the controlled bicycle and are not taken into account by the model matching controller.

This chapter tries to explain the observed differences between the experiment and the theory by analysing the effects of eight different unmodelled phenomena. They are analysed independently of each other to identify their respective influence on the bicycle's dynamics. With the exception of one, all unmodelled dynamics do not explain the difference in both the speed-eigenvalue and the Bode gain plot simultaniously. They diminish the difference in one plot but increase it in the other. A more detailed qualitative description of how the difference is altered is given in Appendix C.

4-1 Analysis of Adjustments to the Model

The theory does not predict the neutral case completely, therefore the original model does not describe the true underlying model of the controlled bicycle. Five unmodelled phenomena can be included in the model of the controlled bicycle by adjusting the original model slightly. The new model is analysed to see if it matches the identified values better in the neutral case. In the active case, the reference model can no longer be used because the model matching controller is wrongly designed. This is because the controller gains depend on the model of the

controlled bicycle. The model matching controller gains used in the experiment incorporate the original model of the controlled bicycle, which does not describe the true dynamics. To mimic the experiment, the new reference model represents the model matching controller with the wrongly designed gains applied to the new model, which is assumed to be the true dynamics. Its **A** and **B** matrix are then defined by

$$\mathbf{A}_{r,new} \coloneqq (\mathbf{A}_{c,new} + \mathbf{B}_{c,new}\mathbf{F}_{mm}), \quad \mathbf{B}_{r,new} = \mathbf{B}_{c,new}\mathbf{G}_{mm}. \tag{4-1}$$

Here the subscript r and c respectively stand for reference and controlled system, *new* refers to an adjusted model, and $\mathbf{F}_{mm}$ and $\mathbf{G}_{mm}$ stand for the feedforward and feedback gain of the model matching controller. This new reference model can be analysed to see if it matches the identified values better. Then the theoretical eigenvalues of the weave mode are calculated using the **A** matrix, and the Bode gain is calculated using the model's equivalent transfer function. The subsections below show the individual effect of five different unmodelled phenomena on the match between the experimental and the theoretical values.

4-1-1 Using Alternate Physical Parameters of the Bicycle

The model matching method makes use of both the controlled and reference bicycle's physical parameters. This is needed to calculate the correct controller gains that lead the controlled bicycle to act as the reference bicycle. While the reference parameters are chosen, the physical parameters of the controlled bicycle have to be measured. These measurements have errors, as indicated in Table 3-1, which lead to a degradation of the controller's performance. This is because the model the controller acts on now differs from the model the controller is designed for. Using the wrong physical parameters can therefore be a reason for the difference between the theory and experiment.

I performed a sensitivity analysis, to identify the influence of each measurement error on the performance of the model matching controller, and identify those physical parameters that have the most effect. This analysis uses the 'one-at-a-time' sensitivity analysis method. It calculates the influence of a system's input on its output by perturbing only one input value at a time from its nominal value. For the controller and bicycle system, the inputs are the 27 physical parameters of the Carvallo-Whipple bicycle model discussed in Section 2-2-2. The two outputs quantify the ability of the model matching controller designed for the nominal parameter values, to still match the original reference if applied to the model with one perturbed parameter. This is done by calculating the average absolute error between the original reference model and the perturbed model with the model matching controller. For the first output, this error is calculated for resulting the speed-eigenvalue plot and for the second output it is calculated for the Bode gain plots. Exact details on how the one-at-a-time sensitivity analysis is performed are found in Appendix B.

Figure 4-1, 4-2, and 4-3, show three different methods to use the one-at-a-time method. Figure 4-1 shows the rate of change to the error as a result of changing one parameter. It shows that relatively the trail and the front wheel's inertia parameters have the most influence on the model matching performance. Figure 4-2 shows the averaged absolute error of each physical parameter, when this parameter is perturbed with 20% from its nominal value. Here multiple parameters were shown to be more influential on the performance compared to the others, with most of them being geometrical parameters. Both Figure 4-1 and 4-2 are

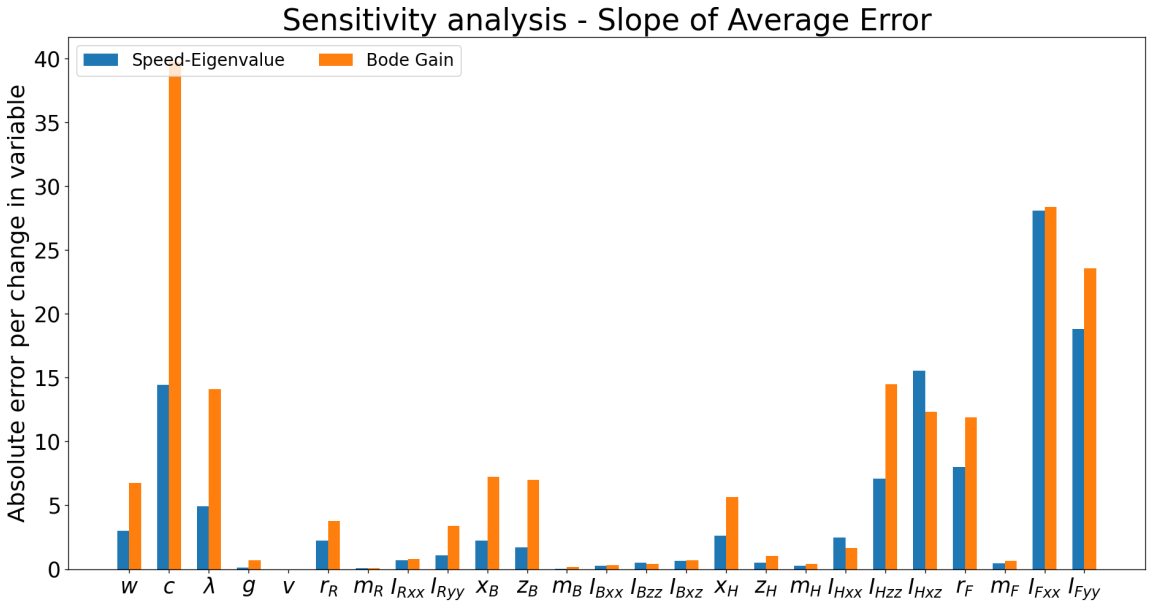

Figure 4-1: Slope of the absolute averaged error between the perturbed system and the reference system. The slope represents the amount of change in the error as a result of a unit change in the respective bicycle parameter. The figure shows both the slopes of the speed-eigenvalue plot (blue) and the bode gain plot (orange).

shown for completeness, further discussion of their implication can be found in Appendix B. Figure 4-3 shows the averaged absolute error of each physical parameter, when this parameter is perturbed from nominal with by its maximum estimated error, indicated in Table 3-1. This figure is the most realistic concerning the experiment, as it takes the actual measurement errors into account. It shows which physical parameters are most likely the cause of the difference between the theoretical and experimental results. These are the inertia moments and products of the rear frame and the front wheel, and the front assembly moment of inertia I_{Hzz}.

From among these parameters I_{Fyy}, I_{Bxx}, and I_{Bzz} were chosen to be altered from nominal. These parameters best diminished the error between the theoretical and experimental results, and additionally where those with the worst accuracy. This is because I_{Bxx}, and I_{Bzz} are only estimations based on a similar bicycle, and physical limitations hindered the accuracy of the measurement of the I_{Fyy} parameter. Trial and error lead to the best fit by changing I_{Bxx}, I_{Bzz}, and I_{Fyy} from their nominal values to 2.0, 2.7, and 0.1189 respectively. The altered parameters are still realistic physical values, and are within their range of expected error. The effect of using these new paramters on the speed-eigenvalue and Bode gain plots are shown in Figure 4-4 till Figure 4-6. Here the blue line represents the corrected bicycle model with the new altered physical bicycle parameters. The orange line is the model matching controller designed for the nominal model applied to the corrected bicycle model, as described above. The markers represent the experimentally identified values.

Using the adjusted model of the controlled bicycle, the theoretical speed-eigenvalue plot and Bode gain plots are calculated again. Their average absolute distance to the experimental values is given in Table 4-1. As seen in the table, the change in physical parameters can best explain the difference between the theoretical and experimental results. Shown in Figure 4-4

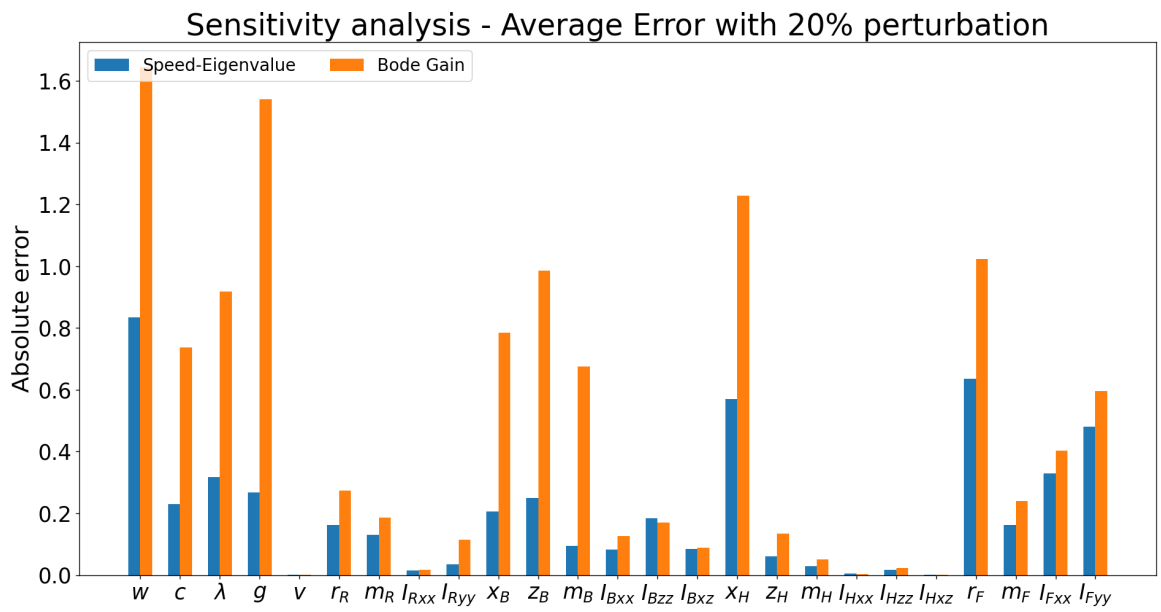

Figure 4-2: Absolute average error between the perturbed system and the reference system when perturbed with 20% of its nominal value. The figure shows both the errors in the speed-eigenvalue plot (blue) and the bode gain plot (orange).

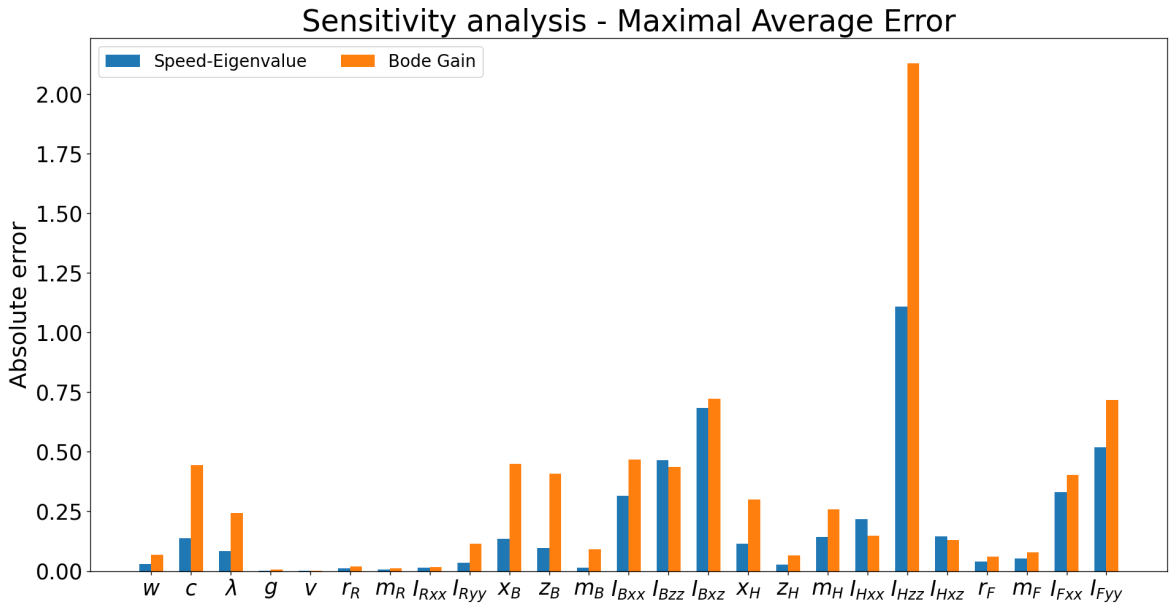

Figure 4-3: Maximum absolute average error between the perturbed system and the reference system. The maximum is the largest resulting error within the range of change for the respective parameter. The figure shows both the errors in the speed-eigenvalue plot (blue) and the bode gain plot (orange).

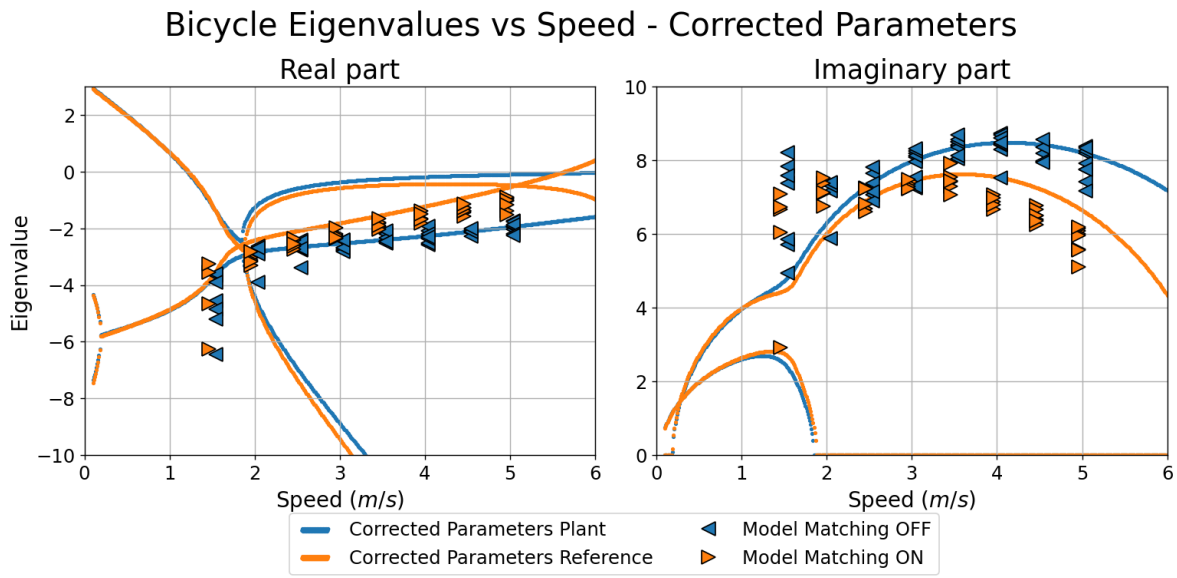

Figure 4-4: Speed-eigenvalue of the bicycle model with corrected model parameters. I_{Bxx}, I_{Bzz}, and I_{Fyy} were altered such that the theoretical model better fitted the experimental data. Here the solid blue line represents the corrected controlled system. The solid orange line represents the model matching controller designed for the nominal system applied to the corrected system. The orange right-facing and blue left-facing triangle represent the experimentally identified eigenvalues for the case where the model matching controller is on and off respectively. The markers should match the lines with the same colour.

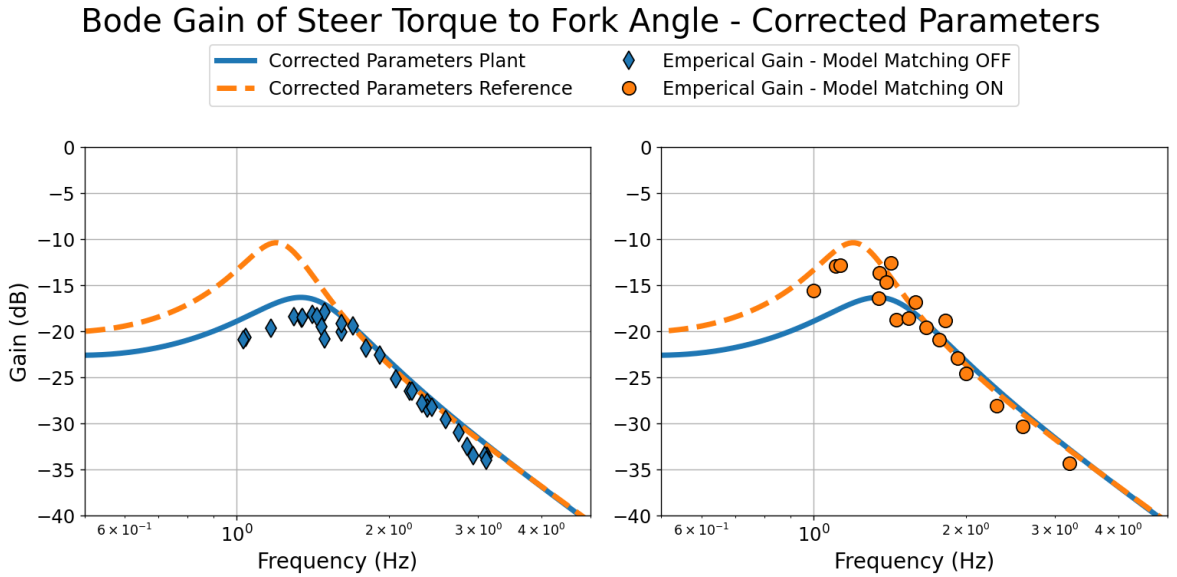

Figure 4-5: Bode gain plot of steer torque to fork angle for the bicycle model with corrected model parameters. I_{Bxx}, I_{Bzz}, and I_{Fyy} were altered such that the theoretical model better fitted the experimental data. Here the solid blue line represents the corrected controlled system. The dashed orange line represents the model matching controller designed for the nominal system applied to the corrected system. The orange circles and blue diamonds represent the experimentally identified bode gain values for the case where the model matching controller is on and off respectively. The markers should match the lines with the same colour.

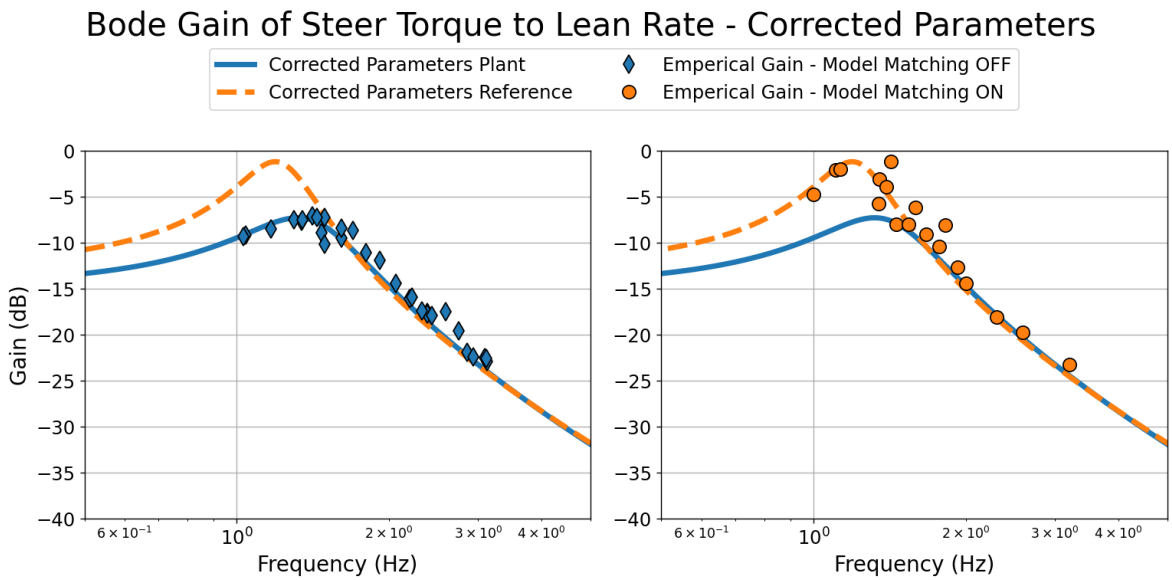

Figure 4-6: Bode gain plot of steer torque to lean rate for the bicycle model with corrected model parameters. I_{Bxx}, I_{Bzz}, and I_{Fyy} were altered such that the theoretical model better fitted the experimental data. Here the solid blue line represents the corrected controlled system. The dashed orange line represents the model matching controller designed for the nominal system applied to the corrected system. The orange circles and blue diamonds represent the experimentally identified bode gain values for the case where the model matching controller is on and off respectively. The markers should match the lines with the same colour.

till Figure 4-6, the new model fits the experimental data better for both the speed-eigenvalue and the bode gain plot, with both the model matching controller being on and off. This is unique, as other phenomena could only reduce the difference in one plot while increasing it in the other.

4-1-2 Including Steering Friction

The Carvallo-Whipple model assumes a frictionless hinge between the rear frame and front fork, which is a good assumption for most bicycles. However, the steer-by-wire bicycle has a non-negligible friction resulting from the drive train between the fork and motor, and the motor itself. A test script in the sympy python library, a computer algebra system presented in [21], creates a nonlinear Carvallo-Whipple model. I adapted this test script such that the nonlinear model includes friction in the steer. By linearising this model, the adjusted model for the controlled bicycle was created. A viscous friction model was applied to keep the linearisation tractable. The friction coefficient was chosen to best fit the experimental data. With this new model, the theoretical plots for the neutral and active case are calculated again and their average absolute distance to the experimental values are calculated. The results are given in Table 4-1. See Appendix C for figures of the speed-eigenvalue and Bode gain plot using the model including friction.

Table 4-1: Average absolute error for all investigated phenomena that had a noticeable influence. The phenomenon that best reduced the difference is highlighted per category.

Phenomenon	Average absolute error			
	Speed-eigenvalue		Bode gain (dB)	
	Controlled	Reference	Controlled	Reference
Nominal	0.733	1.046	1.892	2.810
Altered parameters	0.398	0.715	1.255	1.524
Friction	0.537	0.751	1.152	1.465
Error in speed sensor	0.656	0.977	1.945	2.900
Error in motor torque	0.679	1.021	1.437	1.945
Error in fork encoder	0.733	0.995	1.047	2.078

4-1-3 Using a Wrongly Calibrated Speed Sensor

The measurements of the speed sensor did not coincide with the speed indicated on the treadmill. Assuming the treadmill indicates the true speed, the measured speed is incorrect, and the speed measured by the bicycle and the speed the bicycle travels with are not equal. The gains of both the steer into lean controller and the model matching controller are calculated using the measured speed. Therefore, the controller will apply the wrong gains to the fork motor during the experiment. To mimic the experiment, the speed used in the analysis to calculate the state space matrices of the controlled bicycle model is the true speed. The speed used to calculate the control gains is an adjusted speed. This adjusted speed is calculated using the relation between the treadmill speed and the measured speed. Using this method, the resulting theoretical plots are calculated and their average absolute distance to the experimental values is calculated. Table 4-1 gives the result of this calculation. See Appendix C for the associated theoretical plots, and further details about the relation between the measured speed and the speed indicated by the treadmill.

4-1-4 Using a Wrongly calibrated Motor Torque Command

A calibration test of the handlebar motor showed that the torque physically exerted was only 0.675 times the torque commanded by the microcontroller. Although not measured, it shows there is a possibility that the torque command and the actual torque in the fork motor are not one-to-one. The gain matrices of the model matching controller and steer-into-lean controller are multiplied with a constant scalar value smaller than one to replicate the reduction in exerted force. A scalar value of 0.9 gave the closest match to the experimental values. Then, with the correct torque reduction the theoretical plots are calculated again and their average absolute distance to the experimental values is calculated and given in Table 4-1. See Appendix C for figures of the associated theoretical plots.

4-1-5 Using a Wrongly calibrated Fork Angle Encoder

The absolute encoder used to measure the fork angle consists of a magnet and a PCB. The placement of these parts with respect to each other has to be within the tolerances for the

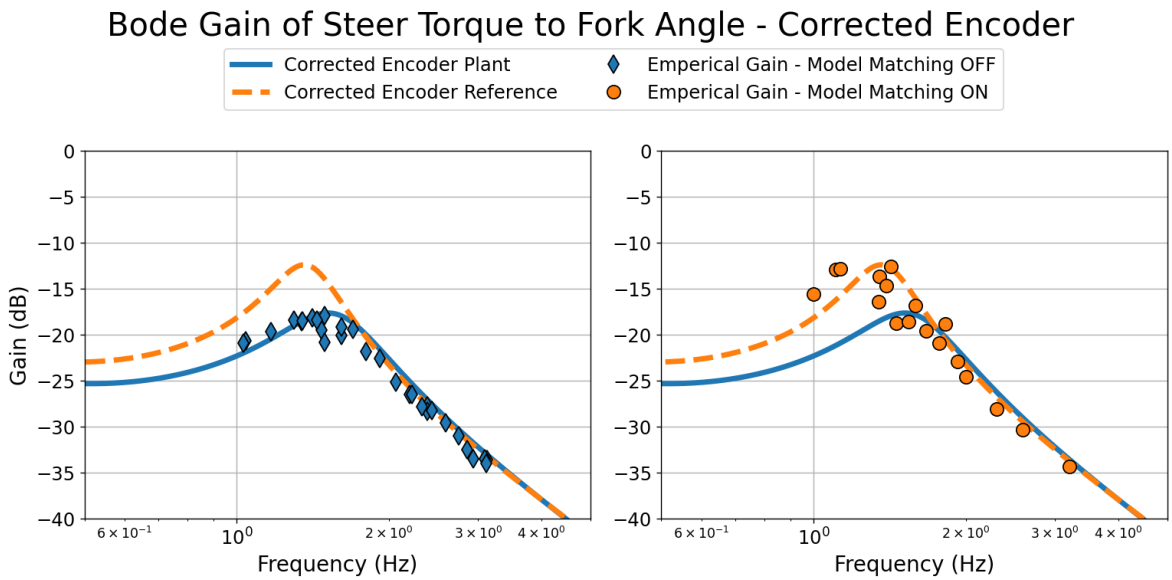

Figure 4-7: Bode gain plot of the controlled system with adjusted fork encoder, referred to in the legend as plant, and the reference system. The markers represent the results of the data analysis from the experiment with model matching controller 'off' (blue diamonds) and 'on' (orange circles). Here the input was the steer torque and the output was the fork angle. The markers should match the lines with the corresponding colour.

sensor to work properly. The implementation of this sensor in the design of the steer-by-wire bicycle made it hard to uphold these positional requirements. During calibration, the zero angle has been correctly assigned, but the measured angle versus the true angle might not have a slope of one. Therefore the fork angle and rate used in the theoretical calculation are multiplied by 0.8. This is done by adjusting the C matrix and the feedback gain of the model matching controller accordingly. Then the theoretical plots are calculated again using the adjusted bicycle model and control gains. Their average absolute distance to the experimental values is calculated and given in Table 4-1. See Appendix C for figures of these theoretical plots.

This adjustment to the model could best explain the downwards shift of the experimental results for the steer torque to fork angle bode plot. See Figure 3-8 for the original Bode plot and Figure 4-7 for the Bode plot with adjusted encoder measurements. The adjusted model both showed a downward shift over the entire frequency and no change for the steer torque to lean rate in the neutral case, which is also seen in the experimental results. The underlying theory is as follows. If the measured angle is 0.8 of the true angle, the magnitude of the measured sinusoidal output is that of the true magnitude divided by 1.25. Because the gain axis in the Bode gain plot is logarithmic, division by a constant becomes subtraction by the logarithm of that constant, and the identified values shift downwards. Furthermore, only the identification of the Bode gain from steer torque to fork angle uses the fork angle measurement.

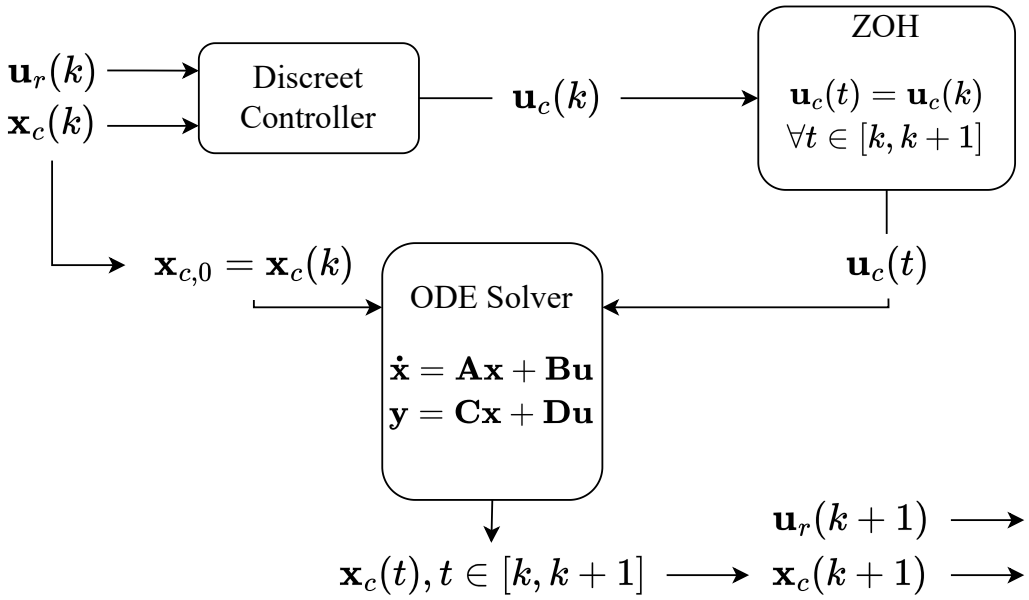

Figure 4-8: Flow chart representing a single iteration of the simulation. The discrete control is made continuous with Zero Order Hold (ZOH). The continuous bicycle system is solved with a ODE solver.

4-2 Analysis through Computer Simulation using an ODE Solver

This section uses a simulation of the controlled bicycle and its controllers to analyse the influence of using estimated state instead of directly measured states. This phenomenon is related to external factors outside the bicycle and its controllers. Therefore only adjusting the model of the controlled bicycle or the applied controllers is not applicable to analyse this phenomenon. The simulation mimics the discrete nature of the controller and the continuous nature of the bicycle. It first calculates the control input based on the current state and the current reference inputs. While keeping the calculated control input constant, a numerical ODE solver then solves the continuous dynamic equations of the bicycle for a set period of time, dt. Upon reaching the end of the time period, the new state is used to calculate the control input for the next iteration, see Figure 4-8. Thus the simulation solves the plant system in a continuous manner, while control is applied at discrete time steps, as is done during the experiment.

The model matching controller is designed to receive full state feedback. However, the steer-by-wire bicycle has no sensors to directly measure the roll angle or steer rate. Instead, they are estimated through an observer and derivative algorithm respectively, as described in Section 3-1-3. According to [19], the observer showed good results with respect to estimating the roll angle. Figure 4-9 presents the error between simulating an impulse response using full state feedback and partially estimated feedback. It shows the difference between the two methods is small and most likely did not cause the difference between the experimental values and the theory.

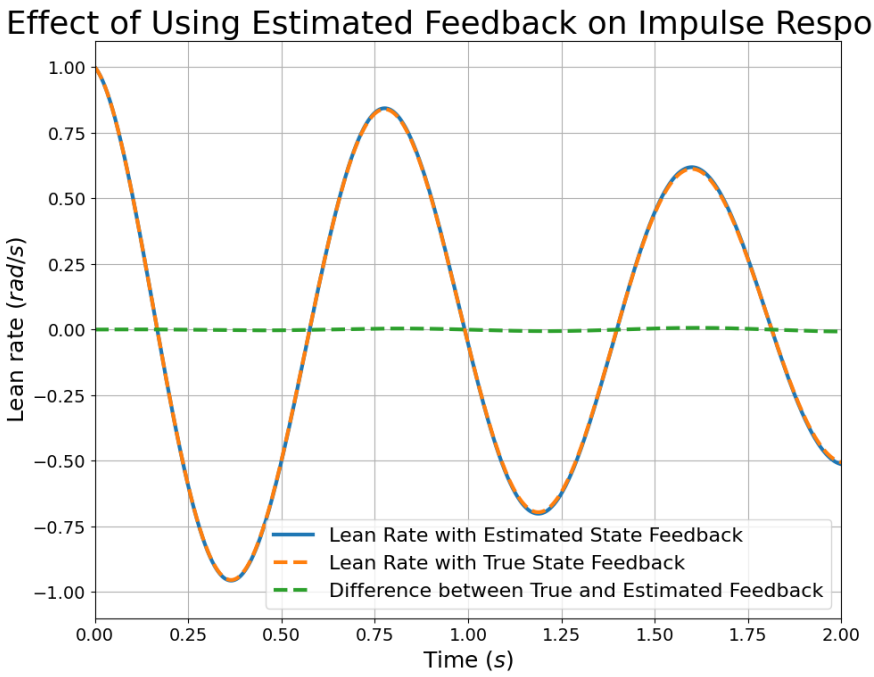

Figure 4-9: The simulated impulse response of the controlled bicycle using full state feedback(dashed orange) and estimated state feedback (solid blue). The error between these two responses is given by the green dashed line.

4-3 The Effect of Post Processing

The final unmodelled phenomenon is the measurement noise, which only influences the measurements. Its effects can be analysed with post-processing methods. The influence of noise proved to be small during the identification of the eigenvalues and bode gain. The lean rate has a visible high-frequency noise, as seen in Figure 3-4. This noise originated from both the electrical noise from the sensors itself, and from vibrations caused by the treadmill. Figure 4-10 and Figure 4-11 show the effect of applying a second-order Butterworth filter to the lean rate, before it is used in the eigenvalue identification. The filter has a cut-off frequency of 10 Hz and a sampling frequency of 100 Hz. The difference between filtered and raw data is small and it suggests that filtering has no impactful effect on the eigenvalue identification. For the bode gain, the identification was done with the use of a Fourier transform. By the nature of the method, high-frequency noise should not affect the identification, as long as the signal-to-noise ratio is high enough. This holds true for both the steer angle, the lean rate, and the force input.

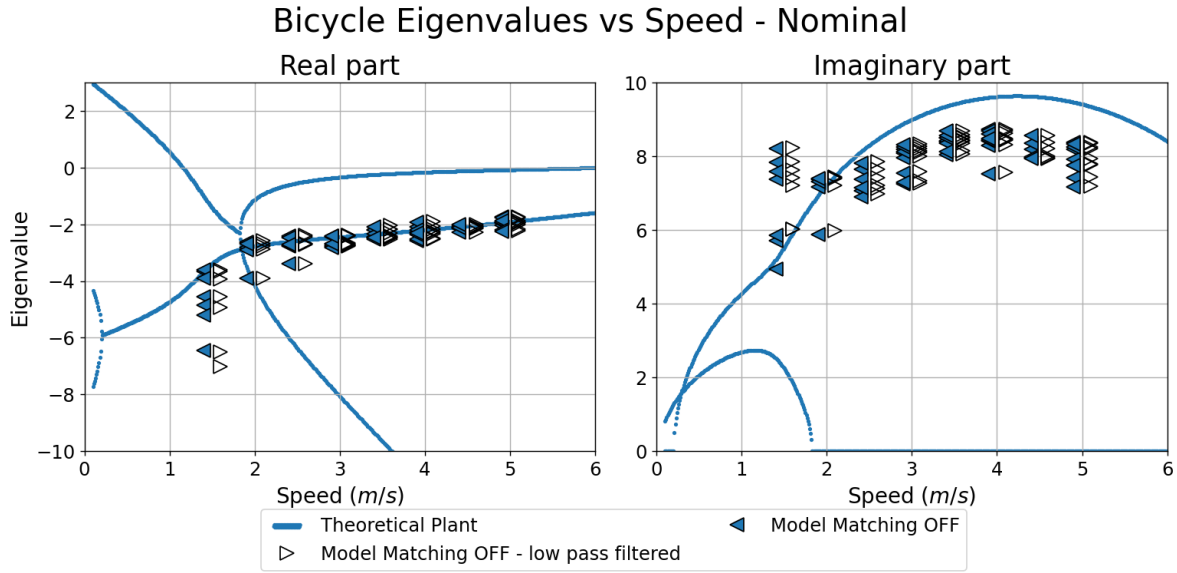

Figure 4-10: Speed-eigenvalue plot showing the theoretical and experimentally identified values of the controller system. Here the open triangles are the eigenvalues identified from the filtered lean rate. The closed triangles use the raw lean rate.

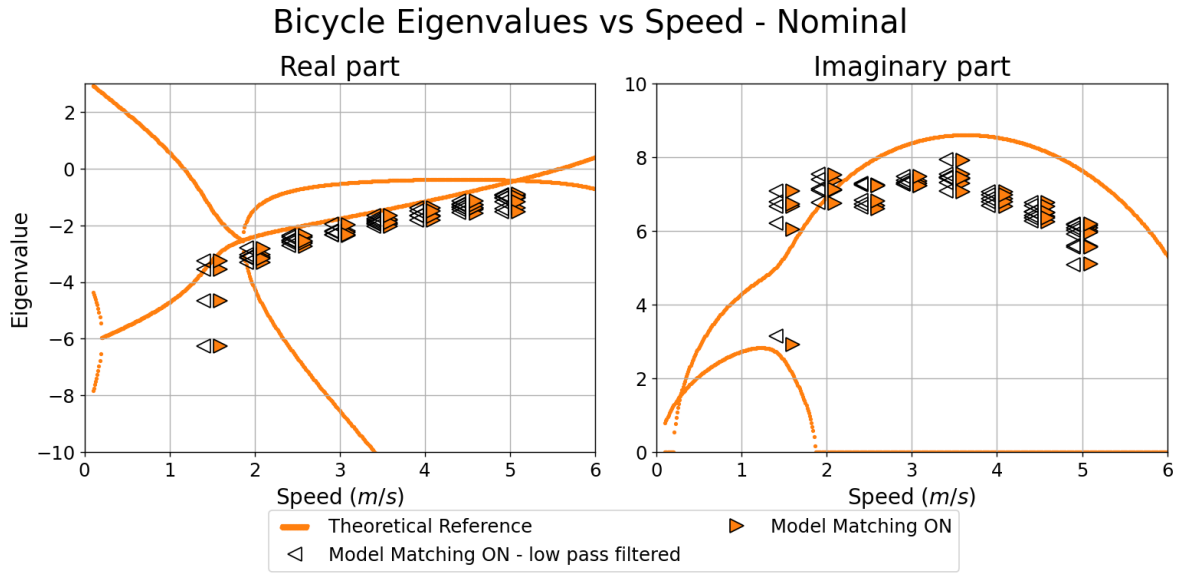

Figure 4-11: Speed-eigenvalue plot showing the theoretical and experimentally identified values of the reference system. Here the open triangles are the eigenvalues identified from the filtered lean rate. The closed triangles use the raw lean rate.

Chapter 5

Discussion

The main research goal of this thesis is to identify the applicability of model matching control to bicycles. Theoretically I have proven it to be possible for the case when both model inputs are available for control and when only the steer torque input is available for control. The sections below discuss the applicability of model matching in practice by discussing the results of the experiment and the post-experiment analysis, and discussing their implication and limitations.

5-1 Discussion of the Results

The figures resulting from the experiment allow for two comparisons. Firstly, it is possible to compare the identified and theoretical dynamics in the neutral case when the model matching controller is off. This makes it possible to assess the correctness of the controlled system model. Figure 3-7 till 3-9 show that the values of the real part of the identified weave mode match the expected theoretical values, with the exception of those at 1.5 meters per second. For the Bode gain plots, the results either match the theoretical values or follow the same trend with an offset. Only the values of the imaginary part of the identified weave mode are noticeably lower than the theoretical values. But they do follow the same curve as the theoretically expected values. The results shown in the figures do not perfectly match the expected values, but they are not arbitrary. It is clear they come from a model very similar to the theoretical one. The model used is a good model but has unmodelled or wrongly modeled dynamics.

Secondly, it is possible to compare the identified and theoretical dynamics in the active case when the model matching controller is on. This allows to identify the capacity of the model matching controller to match the reference model. Figure 3-7 till 3-9 show the same trend as in the neutral case regarding the comparison between the theoretically expected values and the experimentally identified values. The experimentally identified values match the theoretically predicted values well, except for the imaginary part of the weave model eigenvalues. The identified values for the bode gains have more spread, but the slope and peak match those of the theoretically predicted values. The model matching controller is not able to match the

reference model perfectly, but it is again not arbitrary. The experimental results have similar trends as the reference model, and it shows that the model matching controller shifted the system dynamics towards the reference model as expected. Moreover, the results of the active case show the same errors as with the neutral case, inferring the problem may lie with the theoretical model of the bicycle and not the controller.

Altering the bicycle's physical model parameters to fit the experimental results improved the match between the experiment and theory greatly, which is also seen Table 4-1. This new model might not use the true physical values of the controlled bicycle. This is because the bicycle model has 27 parameters, and there is a realistic chance that another set of parameters leads to an equivalent or better match between the theory and experiment. However, the new values are not unrealistic as the parameters that were altered were those that had a high uncertainty, and their values were kept physically feasible. Furthermore, using the experimental results is another method of identifying their values. Therefore I assume this new model can better explain and predict the bicycle in practice.

In the analysis, this new model is used together with the model matching controller designed for the original model to explain the results of the experiment in the active case. Figure 4-4 and Figure 4-6 show that the model matching controller is able to almost completely match the new reference. It shows that if a new model can predict the dynamics of the controlled bicycle in the neutral case, the values expected from theory and the values identified via an experiment match for both the neutral and active case. This gives a strong indication that the model matching controller will work as expected if it is updated to a model better describing the controlled bicycle. However, it is only an indication and does not prove it, as a new experiment should be done to validate this.

With the post-experiment analysis, I investigate other unmodelled phenomena that could explain the differences in theory and practice. The new model shown in Figure 4-4 and Figure 4-6 still showed some differences between theory and experiment, and these investigated phenomena could explain them. Including these phenomena in the original model of the controlled bicycle also improved the model's capability to predict the controlled bicycle's dynamics. Out of the phenomena investigated, friction, and calibration errors in the speed sensor, fork encoder, and motor control showed to have a clear influence. For example, correcting for the fork encoder error can explain the downwards shifted gain in the steer torque to fork angle Bode gain plot, see Figure 3-8. Table 4-1 shows that for all phenomena except the error in the speed sensor, including them into the model quantitatively improves the match between the theory and the experimental results. However, they did not improve the ability to qualitatively explain the theoretical results in both the speed-eigenvalue plots and Bode gain plots at the same time. The eventual real dynamics most likely have a combination of these phenomena. Therefore I recommend measuring these phenomena and taking them into account to improve the next model of the controlled bicycle.

5-2 Implications

Putting this research in perspective, it has achieved the first steps into realising model matching for bicycles. Further research could result in an improvement in safety for commercial bicycles, or a functioning test bench for bicycle related research. As far as I know, there is

no past literature on applying model matching to bicycles, therefore the results are compared to model matching control in automobiles. This is generalisable as model matching itself is a method and is separate from the system it is applied upon. In [7, 8, 4] the authors all use the same model matching method as in this research, and show it can work in practice. They all show that the measured time domain trajectories match that of the reference model. Different from this research, these papers took additional actions on top of the model matching control. While not explicitly mentioned, these additional actions reduce the modelling errors themselves or the amount of influence they have on the system. In [7] the author uses an appropriate model for the steer-by-wire steering system, which includes effects such as friction. As a result, the physical vehicle reacts as the ideal model in terms of steering. In [4] the author uses the output of the model matching controller as a feedforward input and adds an extra feedback loop to compensate for disturbances or modelling errors. The same structure is described in a theoretical case in [15].

5-3 Limitations

The goal of the experiment is to validate the model matching controller and prove it can indeed virtually change the model parameters. Ideally, identifying the **A** and **B** matrix of the controlled bicycle would allow it to be compared to the reference bicycle. The size and duration of performing such identification experiments, however, is out of scope for this master thesis. To validate if the controller matches the dominant autonomous behaviour of the reference bicycle, the performed experiment identifies the controlled bicycle's eigenvalues belonging to the weave mode. For the input-to-output relation, the experiment identifies the bode gain of the steer torque to the lean rate and steer torque to the steer angle for a single speed. These identifications give a characterization of the dynamics of the bicycle system and are a good first assessment of the validity of the controller.

Ideally, the bode plot should be identified by applying an impulse and analysing its frequency domain response. This would give a continuous bode gain for the entire frequency range. However, the bicycle was not able to stay on the treadmill for long enough to measure a sufficient response. Therefore the bode gain was identified discretely, as described in Chapter 3-2. The discrete nature and the fact that it did not cover the entire frequency results in a lower quality comparison. However, the data points are close enough to each other to see a trend. Furthermore, they are taken in the frequency most important during bicycle operation, which is frequencies below three Hz.

The sensitivity method has two issues. Firstly it only informs about a specific controlled and reference system. This is because the model matching controller is dependent on these models. It is not guaranteed that similar results will follow from the analysis if one of the systems is changed. However, in this thesis, the analysis was done with the bicycle's physical parameters used in the experiment. Therefore the results are applicable to the investigated controller.

Secondly, the one-at-a-time method cannot investigate simultaneous variations in the bicycle's physical parameters and therefore does not fully explore the input space. This can be solved by using a variance-based method, but will require more effort to implement. As the sensitivity analysis is not the main scope of this thesis, time constraints led to the choice of the simpler

method. However, the initial goal is to gain insight into the influence of a single parameter on the performance of the controller, and identify which parameters should be altered to better fit the experimental results. The one-at-a-time method satisfies this goal and therefore suffices as the method used.

The theory shows it was not possible to choose any reference model, as mentioned in Chapter 2. Due to the limitation of only being able to control the steer torque, five physical parameters of the reference model were restricted. This is impractical for the purpose of mimicking other real physical bicycles. However, model matching has not been applied to bicycles before, and the first steps have to be made. The research is still able to investigate if it is possible to match a reference model, independent of the practicality of that reference.

Chapter 6

Conclusion and Future Work

6-1 Conclusion

This thesis aimed to apply model matching control to bicycles, both in theory and in practice. Theoretical analysis of the proposed control method shows it is able to match the dynamics of a controlled bicycle model to that of a reference model. Additionally, it shows that model matching is still possible when steer torque is the only control input available, but the choice for the reference model is limited. The steer torque can be given both by a steer-by-wire actuation system or by a motor parallel to the steer.

The model matching controller was successfully able to run on a steer-by-wire bicycle. To validate the controller, I identified the eigenvalues of the weave mode for speeds between 1.5 and 5 m/s, and identified the bode gain relationship of the steer torque to lean rate and to steer angle for frequencies between one and three Hz. Repeated experiments gave consistent results, and characterise the dynamics of the bicycle.

The model matching controller running on the controlled bicycle is able to change the bicycle's dynamics in the expected manner, as it pushes the eigenvalues and bode gain in the right direction. The used model for the controlled bicycle did match the experimental results well in most areas, but had a large error in the imaginary part of the weave mode eigenvalue. Consequently the same holds for the model matching controller's ability to match the reference model. I performed post-experiment simulations to investigate these modelling errors. Adapting the bicycle's physical parameters best explained the error. Applying the model matching controller, which is designed for the original model, to this adapted model could predict the experimental results well.

This thesis has shown that model matching control is applicable to bicycles in a sufficient manner. It is able to match the dynamics of the reference model for the most part, but not entirely. Analysis suggests that improving the model of the bicycle will improve the performance of the model matching controller. With these results, this research made the first step to applying model matching control to bicycles.

6-2 Future Work

To freely choose any reference model it is necessary to control the lean torque as well. As the bicycle should be able to drive freely, this torque should be generated by the bicycle itself. Possibilities are the use of a controlled moment gyroscope, a reaction wheel, or actuated legs attached to the bicycle that can push on the ground. Further research can investigate if these devices can deliver the torque needed by the model matching controller, while still being safe and of practical size.

During the experiments the bicycle experienced a large drift to the side, causing it to ride off the treadmill. It limited the length of the measured data of a single trial, leading to less accurate identifications. The drift can be the effect of a buckled wheel, the front and rear wheels not being aligned, an asymmetric weight distribution in the longitudinal plane, the ground plane not being level, or the interaction between the tyre and the treadmill. To improve the experiment, either the bicycle used can be corrected for the drift, or the drift should be allowable. This can be done by for example perform the experiment in a gym hall, that has a large flat ground and no wind.

To increase the effectiveness of the controller and achieve model matching, either the model of the controlled system should represent the reality better, or the control structure should be made more robust to modelling errors. A better model can be achieved by measuring the bicycle parameters as precisely as possible. I recommend using the methods described in [16]. Additionally, the friction coefficient between the front assembly and the rear frame can be determined experimentally. The controller can then take the friction into account by substituting the controlled plant model with the adapted Carvallo-Whipple model that includes friction. To further minimize the difference between model and reality, the test setup can be made more reliable. This can be done by assuring a known relation between commanded torque and actuated torque, and calibrating the speed sensor and fork encoder against a known reference. Future research can repeat the experiment performed in this thesis with this more accurate model.

Alternatively, the controller can be made robust against modelling errors. The model matching controller currently does not use the error between the virtual states of the reference system and the actual physical bicycle. Both [4] and [15] make use of this error by having a two-degree-of-freedom control structure for their model matching controller. Figure 6-1 shows a control diagram depicting this structure. The feedforward control consists of a dynamic block. This block simulates a model matching controller, as applied in this thesis, that controls a theoretic model of the controlled plant. This is the same model used to create the gains of the model matching controller, hence the controller achieves mimicking of the reference. In this way, the feedforward block gives both the input needed to match the reference if there were no modelling errors, and the state the reference model would be in at that moment. Then the feedforward control is directly applied to the controlled system, and the resulting states are compared to the states of the reference model. The error is given to the feedback controller which is designed to control this error to zero. To increase robustness, the feedback controller can be designed as an H_∞ controller or other controller types that focus on robustness.

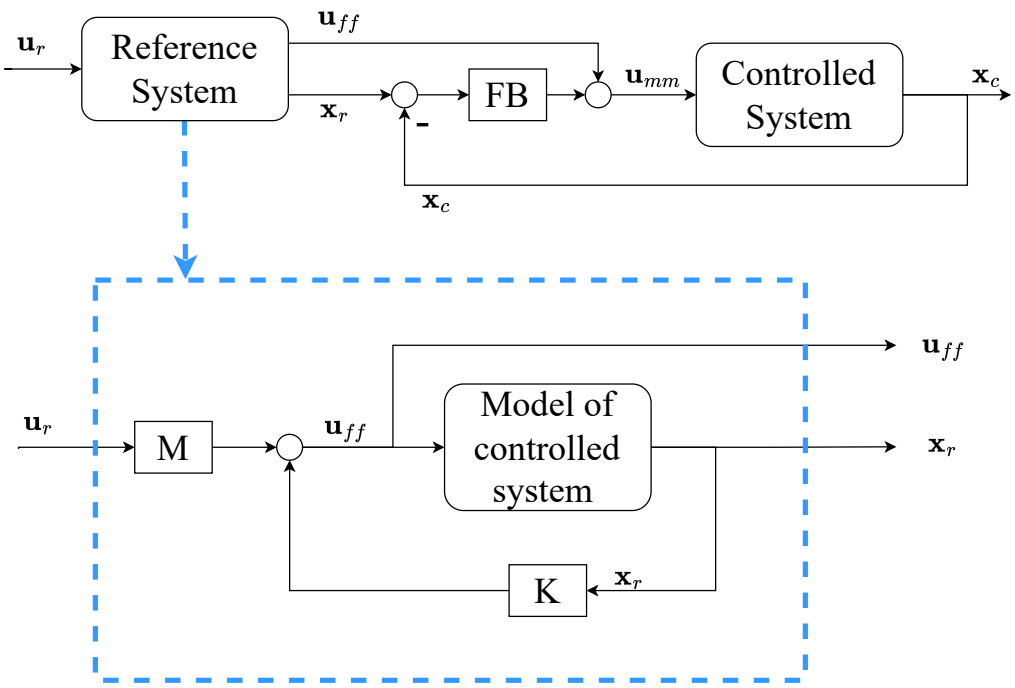

Figure 6-1: Control structure of a two-degree-of-freedom model matching controller. This structure can be used to increase the robustness against modelling errors. Here $\mathbf{u}_r$ is the reference input given by the human, $\mathbf{u}_{ff}$ is the feedforward control that would achieve model matching if there were no modelling errors of external disturbances, and $\mathbf{u}_{ff}$ is the complete model matching control applied to the controlled system. The gain blocks M and K are the model matching gains to let the model of the controlled system match the reference system. 'FB' stands for the feedback controller, and $\mathbf{x}_r$ and $\mathbf{x}_c$ are the states of the reference and controlled system respectively.

Bibliography

[1] P. G. Hamel, "Variable Stability Aircraft and In-Flight Simulators," in *In-Flight Simulators and Fly-by-Wire/Light Demonstrators: A Historical Account of International Aeronautical Research*. Cham: Springer International Publishing, 2017, pp. 33–71. [Online]. Available: https://doi.org/10.1007/978-3-319-53997-3_5

[2] N. C. Weingarten, "History of In-Flight Simulation & Flying Qualities Research at Calspan," *AIAA Atmospheric Flight Mechanics Conference*, vol. 42, no. 2, 2003.

[3] A. Y. Lee, "Matching Vehicle Responses Using the Model-Following Control Method," SAE International, Warrendale, PA, SAE Technical Paper 970561, Feb. 1997. [Online]. Available: https://www.sae.org/publications/technical-papers/content/970561/

[4] H. E. B. Russell and J. C. Gerdes, "Design of Variable Vehicle Handling Characteristics Using Four-Wheel Steer-by-Wire," *IEEE Transactions on Control Systems Technology*, vol. 24, no. 5, pp. 1529–1540, Sep. 2016.

[5] S. Horiuchi, N. Yuhara, and A. Takei, "Two Degree of Freedom/H_∞ Controller Synthesis for Active Four Wheel Steering Vehicles," *Vehicle System Dynamics*, vol. 25, pp. 275–292, Jan. 1996. [Online]. Available: http://www.tandfonline.com/doi/abs/10.1080/00423119608969201

[6] P. I. Ro and H. Kim, "Four Wheel Steering System for Vehicle Handling Improvement: A Robust Model Reference Control Using the Sliding Mode," *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, vol. 210, no. 4, pp. 335–346, Oct. 1996. [Online]. Available: https://doi.org/10.1243/PIME_PROC_1996_210_280_02

[7] P. Yih and J. Gerdes, "Modification of Vehicle Handling Characteristics via Steer-by-Wire," *IEEE Transactions on Control Systems Technology*, vol. 13, no. 6, pp. 965–976, Nov. 2005.

[8] J. W. Brown, R. K. MacLean, S. Laws, C. Gadda, and J. C. Gerdes, "Experimental Vehicle Handling Modification through Steer-by-Wire and Differential Drive," in *2007 American Control Conference*, Jul. 2007, pp. 2302–2307.

[9] J. Meijaard, J. M. Papadopoulos, A. Ruina, and A. Schwab, "Linearized Dynamics Equations for the Balance and Steer of a Bicycle: a Benchmark and Review," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 463, no. 2084, pp. 1955–1982, Jun. 2007. [Online]. Available: https://royalsocietypublishing.org/doi/abs/10.1098/rspa.2007.1857

[10] E. Carvallo, *Théorie du mouvement du monocycle et de la bicyclette*. Paris, France: Gauthier-Villars, 1901.

[11] F. J. Whipple, "The stability of the motion of a bicycle," *Quarterly Journal of Pure and Applied Mathematics*, vol. 30, no. 120, pp. 312–348, 1899.

[12] J. D. G. Kooijman, A. L. Schwab, and J. P. Meijaard, "Experimental Validation of a Model of an Uncontrolled Bicycle," *Multibody System Dynamics*, vol. 19, no. 1, pp. 115–132, Feb. 2008. [Online]. Available: https://doi.org/10.1007/s11044-007-9050-x

[13] A. L. Schwab and N. Appleman, "Dynamics and Control of a Steer-by-Wire Bicycle," in *Bicycle and Motorcycle Dynamics 2013*, November 2013.

[14] J. K. Moore and M. Hubbard, "Expanded Optimization for Discovering Optimal Lateral Handling Bicycles," in *Bicycle and Motorcycle Dynamics 2019*. Symposium on the Dynamics and Control of Single Track Vehicles, Sep. 2019.

[15] P. Hippe and J. Deutscher, "Model-matching Control with Two Degrees of Freedom," in *Design of Observer-based Compensators: From the Time to the Frequency Domain*. London: Springer, 2009, pp. 185–208. [Online]. Available: https://doi.org/10.1007/978-1-84882-537-6_9

[16] J. K. Moore, M. Hubbard, A. L. Schwab, and J. D. G. Kooijman, "Accurate Measurement of Bicycle Parameters," in *Symposium on the Dynamics and Control of Single Track Vehicles*, The Netherlands, 2010.

[17] S. Zhang and T. Tak, "A design sensitivity analysis of bicycle stability and experimental validation," *Journal of Mechanical Science and Technology*, vol. 34, no. 9, pp. 3517–3524, Sep. 2020. [Online]. Available: https://doi.org/10.1007/s12206-020-0803-2

[18] G. Dialynas, R. Happee, and A. Schwab, "Design and implementation of a steer-by-wire bicycle," in *International Cycling Safety Conference*, vol. 11. Barcelona, Spain, 2018.

[19] E. Sanjurjo, M. A. Naya, J. Cuadrado, and A. L. Schwab, "Roll Angle Estimator Based on Angular Rate Measurements for Bicycles," *Vehicle System Dynamics*, vol. 57, no. 11, pp. 1705–1719, Nov. 2019. [Online]. Available: https://www.tandfonline.com/doi/10.1080/00423114.2018.1551554

[20] A. L. Schwab, J. D. G. Kooijman, and J. P. Meijaard, "Some recent developments in bicycle dynamics and control," in *Proceedings of the Fourth European Conference on Structural Control*, St.Petersburg, Russia, Sep. 2008, pp. 695–702.

[21] A. Meurer, C. P. Smith, M. Paprocki, O. Čertík, S. B. Kirpichev, M. Rocklin, A. Kumar, S. Ivanov, J. K. Moore, S. Singh, T. Rathnayake, S. Vig, B. E. Granger, R. P. Muller, F. Bonazzi, H. Gupta, S. Vats, F. Johansson, F. Pedregosa, M. J. Curry,

A. R. Terrel, v. Roučka, A. Saboo, I. Fernando, S. Kulal, R. Cimrman, and A. Scopatz, "Sympy: symbolic computing in python," *PeerJ Computer Science*, vol. 3, p. e103, Jan. 2017. [Online]. Available: https://doi.org/10.7717/peerj-cs.103

Appendix A

Measurement Methods for the Bicycle Parameter

This appendix describes the methods used to obtain 25 physical parameters of the bicycle. These are necessary to create a model describing the controlled bicycle. Only 17 of the parameters will be measured, the six remaining parameters will be copied from similarly shaped bicycles. These values are found in [16] which describes accurate methods to measure the physical bicycle parameters and uses them to measure the parameters of ten bicycles. The copied values are the inertias from the rear frame and front assembly. They are not measured due to the time investment of the measurement themselves and the difficulty of disassembling the steer-by-wire bicycle thereby potentially damaging the system. The measurement techniques were either taken from or inspired by [16]. Some measurements were simplified as the focus was to have a good enough estimation in a short time span. The following sections describe how I measured the seventeen parameters and how I chose the 6 values selected from other bicycles.

A-1 Geometric relations

Wheelbase The wheelbase is measured with a tape ruler. Here the normal vector of the front wheel is put parallel to the normal vector of the rear frame.

Trail The trail is indirectly measured via the 'alternative geometry measurement method' taken from [16]. To measure the fork offset without disassembling the front fork, I attached a wooden plank to the straight part of the fork. This acted as a substitute for a straight table.

Head tube angle The head tube angle is measured via a digital level on a phone. The bicycle is first put in an upright position with zero lean and fork angle. The tyres are pumped up to six bars to mimic the conditions of the experiment. The phone is held against the wooden plank mentioned above, as this plank is parallel to the steering axis.

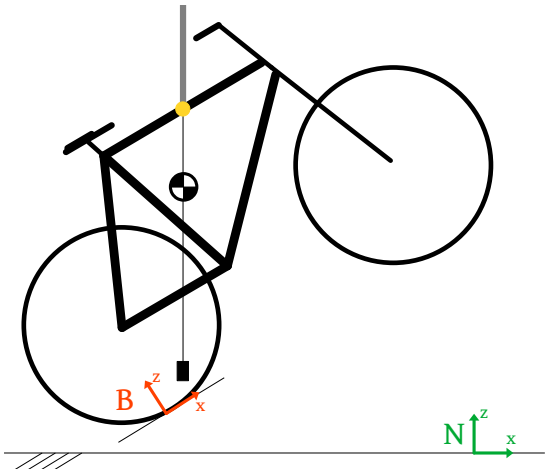

Figure A-1: Schematic representation of the bicycle hung from the ceiling. Here N is the inertial frame, and B the bicycle body fixed frame.

Wheel radius Wheel radius is calculated by measuring the circumference with a tape ruler and dividing the measurement by 2π. Again the tyres are pumped up to six bar.

A-2 Mass

The mass is measured with a digital hanging scale. The front and rear wheels are measured separately. The front and rear frames are measured together, as it was not possible to disassemble them. Steer-by-wire mechanically decoupled the handlebar from the front fork. Therefore the front frame is defined as only the front fork and the part attaching the front fork to the front wheel motor. The handlebar is fixed in place with lashing straps between the handlebar and the rear frame. Therefore the handlebar and part of the head tube that connects it to the handlebar motor is defined as part of the rear frame. To get the separate weights of both the rear frame and front frame, the weight of the front assembly was estimated. Consequently, the weight of the rear frame is the total weight minus the estimated weight of the front assembly.

A-3 Center of mass

The method to measure the center of mass is also taken from [16]. The center of mass is calculated by hanging the bicycle from the ceiling with a cable. A small weight on a string is attached at the point where the bicycle is tied to the cable and creates a vertical line. This represents the line going through the center of mass of the bicycle, see Figure A-1. Here B is the body fixed frame, and N is the inertial frame. Then I measure the distance between the rear axle and the vertical line going through the hinging point with a ruler. The angle between the horizontal ground plane and the head tube angle is measured using a digital level on a phone. Using the geometrical relation as described in [16], it is possible to express the vertical line in the bicycle's body fixed frame.

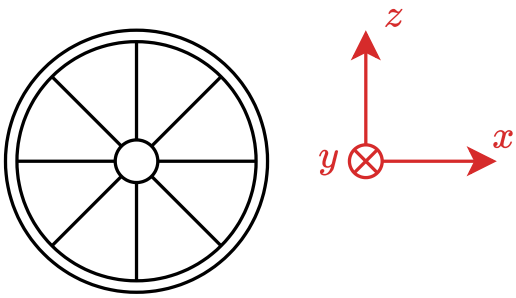

Figure A-2: schematic representation of the wheel defining its axes.

Hanging the bicycle from a different point will result in a different line that goes through the center of mass. The intersection of at least 2 lines will give the location of the center of mass. For the measurement of the steer-by-wire bicycle's center of mass, I did four measurements and calculated four lines. These lines did not intersect in a singular point due to measurement errors. Instead, they give a region in which the center of mass lies. I used the average of the corners of this region to represent the coordinates of the center of mass.

The center of mass of the wheels is assumed to be in the center of the wheel. Due to the difficulties of disassembling the steer-by-wire bicycle, only the front wheel has been removed from the bicycle during measurements. The center of mass of the combined rear and front frame are calculated using the following equation

$$p_a = \frac{p_{a+b}\left(m_a + m_b\right) - p_b m_b}{m_a}. \tag{A-1}$$

Here p represents either the x or z center of mass coordinate, m a body's mass, a and b indicate different bodies, and $a + b$ indicates the combined bodies. All variables on the right side of the equation are assumed to be known.

First, the center of mass of the combined rear and front frame is calculated using (A-1). Here the mass of the combined rear and front frame, the center of mass of the rear wheel, and the rear wheel's weight are known. Then the center of mass of the front frame is estimated by estimating the location of the volumetric center, and assuming equal density. Lastly, the center of mass of the rear frame is calculated using (A-1). Here the mass of the rear frame, the front frame's center of mass, and its mass are known.

A-4 Inertia

The method to measure the inertias is again taken from [16]. Only the inertia tensor of the wheels is measured, the inertia tensors of the front and rear frame are estimated using completed measurements of similar bicycles. The wheel's yy inertia, see Figure A-2, is measured by hanging up the wheel on a level rod. An impulse is given causing the wheel to swing around the yy axis. The period of multiple oscillations is measured with a stopwatch and used to calculate the average period of one oscillation. Here the wheel is assumed to be a

compound pendulum. Therefore the yy inertia is calculated using the following equation

$$I_{yy} = \left(\frac{\bar{T}}{2\pi}\right) mgl - ml^2, \tag{A-2}$$

taken from [16]. Here I_{yy} is the yy inertia, $\bar{T}$ is the average period of one oscillation, m the wheel's mass, g the gravitational constant, and l the distance between the wheel's center of mass and the contact point of the rod and rim of the wheel.

The xx inertia is measured by using a large metal rod functioning as a torsional spring. The rod is clamped vertically from the ceiling, and the wheel is clamped to this torsional spring, such that the torsional spring is in line with the xx direction of the wheel. both the xx direction and the torsional spring are in parallel to the gravity vector. An impulse is given in the y direction, and the wheel's period of oscillation is measured. The xx inertia is calculated using the following formula

$$I_{xx} = \frac{k\bar{T}^2}{4\pi^2}, \tag{A-3}$$

taken from [16]. Here I_{xx} is the xx inertia, $\bar{T}$ is the average period of one oscillation, and k is the stiffness of the torsion spring. This stiffness is measured by applying the same experiment with an object with known inertia.

I take the inertial tensor of the 'Davis instrumented bicycle' from [16] to represent the inertial tensor of the rear frame. The instrumented bicycle like the steer-by-wire has a battery at the seat tube and electronics on the luggage carrier. To represent the inertial tensor of the front fork, I take the inertial tensor of the 'Bianchi Pista' from [16], as this most resembles the thin front fork with no fender of the steer-by-wire. However, the inertia tensor of the Bianchi Pista is of both front fork, head tube, and steer. Therefore it is divided by ten before being used as the inertial tensor of the steer-by-wire bicycle. This factor seems reasonable as the steer and part of the head tube are a large portion of the weight and volumetric space of the front frame. The inertia data taken from [16] is licensed under Creative Commons Attribution 3.0 Unported License

A-5 Values of the Measured Bicycle Parameters

The values in the 'Controlled Bicycle's Measurement Error' column of Table 3-1 indicate the estimated error during measurement. This estimate creates a range around the measured value, that is taken such that there is only a small chance that the true value is outside this range. The size is an educated guess that depends on how accurate the measurement method was and takes into account finite resolution and human errors. The measurement error is only used during the sensitivity analysis. Therefore an estimation is sufficient, as it is only used to compare the accuracy between the parameters itself.

Appendix B

Elaboration on the Applied Sensitivity Analysis

B-1 Detailed Explanation of the Sensitivity Analysis

In this thesis, the method used for the sensitivity analysis is the 'one-at-a-time' method. It calculates the influence of a system's input on its output by perturbing only one input value at a time from its nominal value. Applying this method to the model matching controller, the input space is the 27 bicycle parameters of the Carvallo-Whipple model. One output is calculated by making use of the speed-eigenvalue plot. The gains for the model matching controller are calculated to match a nominal model to the reference model, which will be an exact match when there are no modelling errors, see Chapter 2. The nominal model uses the values shown in Table 3-1. This controller is then applied on a perturbed system instead, which has only one parameter different from the nominal values. This results in Figure B-1. The analysis calculates, for all speeds and for all four eigenvalues, the absolute error between the equivalent eigenvalues of the reference system and controlled perturbed system. These are summed and averaged over the speed dimension and over the number of eigenvalues to create the output. In case the bode gain plot is used instead, the output is calculated by taking the absolute error between the reference system and controlled perturbed system per frequency, summing over all frequencies, and averaging it over the the frequency dimension and the number of input to output pairs. These use the plot shown in Figure B-2.

This process is done for all parameters using multiple perturbed values per parameter. The range depends on the estimated measurement error of the parameter, where large errors result in larger ranges. It creates a plot with the deviation from nominal on the x-axis and the output representing the controller performance degradation on the y-axis. The slope of this plot gives the sensitivity of controller performance to the bicycle parameter, and is constant over the x-axis for all parameters as a result of taking the absolute of the eigenvalue or bode gain error. Combined with the larger range for larger measurement errors, the analysis shows which parameters causes the most performance degradation for the current set of measured

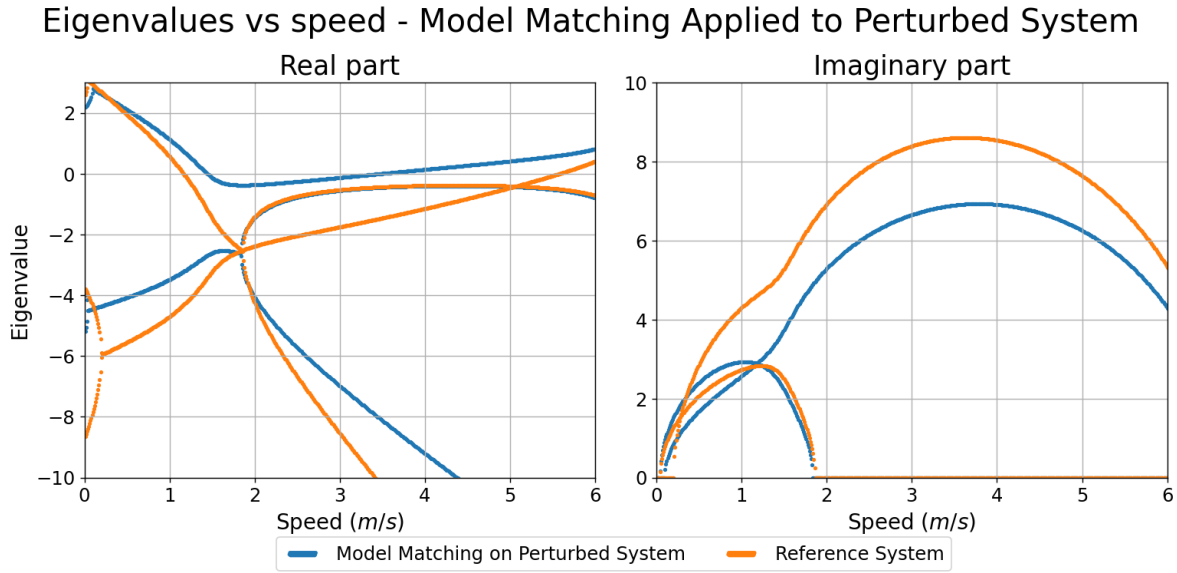

Figure B-1: Speed eigenvalue plot of the reference model and the perturbed model with a model matching controller designed for the nominal model. Here the variable I_{Hzz} is perturbed 2325% from its nominal value.

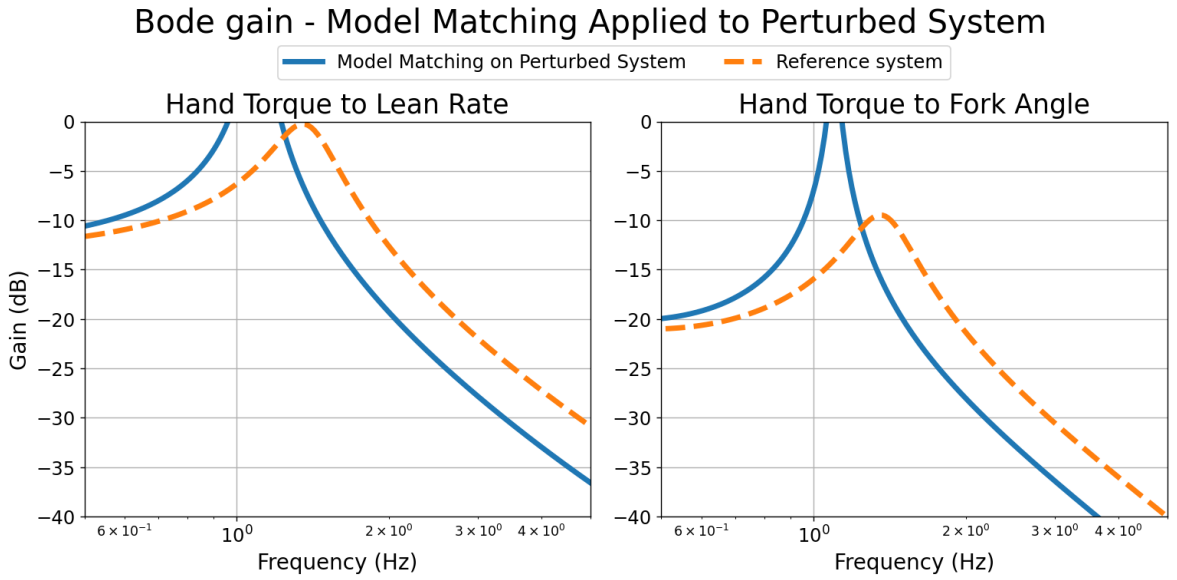

Figure B-2: Bode gain plot of the reference model and the perturbed model with a model matching controller designed for the nominal model. Here the variable I_{Hzz} is perturbed 2325% from its nominal value.

nominal values. Note that the eigenvalue and bode gain error are not comparable with each other as their output are calculated slightly differently. There is also no physical reference for the absolute value errors, and are compared relative to each other.

B-2 Interpretation of Sensitivity Analysis Results

An imprecise measurement technique can lead to large performance degradation. The controller is more sensitive to some parameter than other. Taking the parameter measurement performed in this thesis into account, the performance degradation is dominated by the measurement uncertainty. This is most visible by comparing Figure 4-1 and Figure 4-3. Here the parameters with the highest performance degradation are not necessarily those with the largest sensitivity.

The sensitivity analysis show that the model matching controller is not robust in regards to errors in the modelling parameters. An error in the model parameters prevents the model matching controller from obtaining the reference system's behaviour as is visible from Figure B-1 and Figure B-2. To contrast this statement, not all parameters lead to a large deviation from the reference. Additionally these figures show the worst case for each respective parameter. This is not necessarily the actual physical error, which can be close to the nominal value even if the uncertainty is high. However, it is already apparent that an error in a single parameter can be detrimental. With the bicycle having 27 model parameters, the total performance degradation can be noticeable, even if all parameters are not at their maximum estimated error.

Appendix C

Additional Figures of the Effect of Modelling errors

C-1 The Effect of an Erroneous Fork Encoder on the Bicycle Dynamics

Figure C-1 till Figure C-3 shows the effect of implementing the error in the fork angle encoder into the model. It does not noticeable influence the speed-eigenvalue plot, and the bode gain plot for steer torque to lean rate. However, it match the theory better for the steer torque to fork angle bode gain. Thus an error in the speed encoder can only partially explain the difference between the theoretically predicted and experimentally measured values.

C-2 The Effect of an Erroneous Speed sensor on the Bicycle Dynamics

Figure C-4 shows the speed measured by the sensor against the speed indicated by the treadmill. These different speed indications are slightly different as they have a different slope. Figure C-5 till Figure C-7 shows the effect of applying the control gains calculated at the wrong speed. The effect is small for lower speeds. For speeds above four meters per second, the imaginary part start to gain a more negative slope for larger speeds.

C-3 The Effect of an Erroneous Motor command on the Bicycle Dynamics

Figure C-8 till Figure C-10 show a simulation where the applied torque is 0.9 times the commanded torque, and compares it to the nominal case. The reduction in torque improves

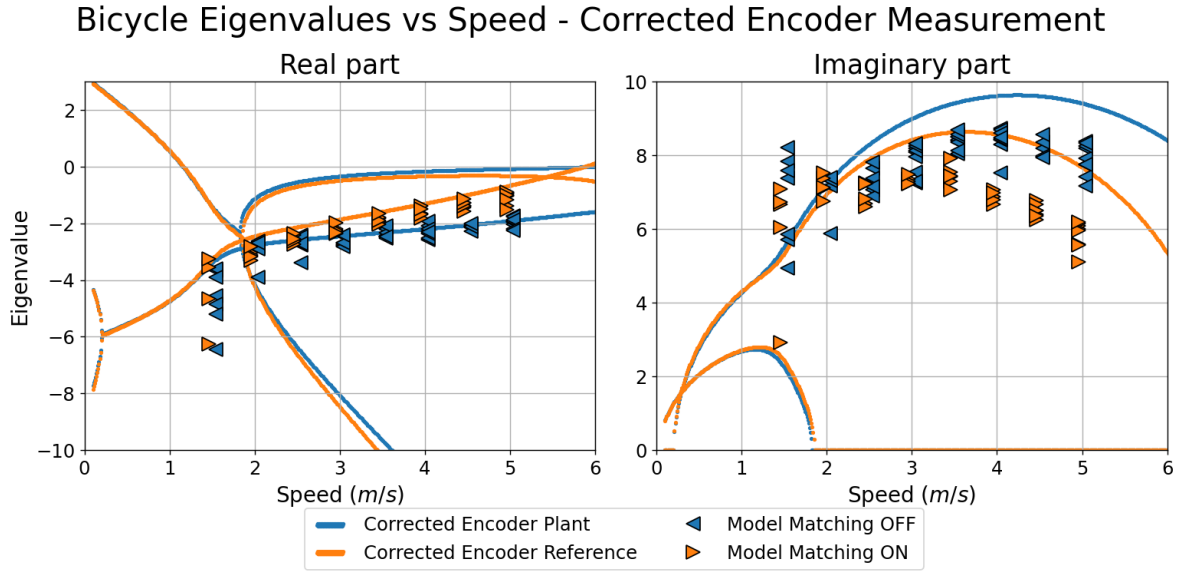

Figure C-1: Speed-eigenvalue of the bicycle model with corrected fork encoder. Here the solid blue line represents the corrected controlled system. The solid orange line represents the model matching controller designed for the nominal system applied to the corrected system. The orange right-facing and blue left-facing triangle represent the experimentally identified eigenvalues for the case where the model matching controller is on and off respectively. The markers should match the lines with the same colour.

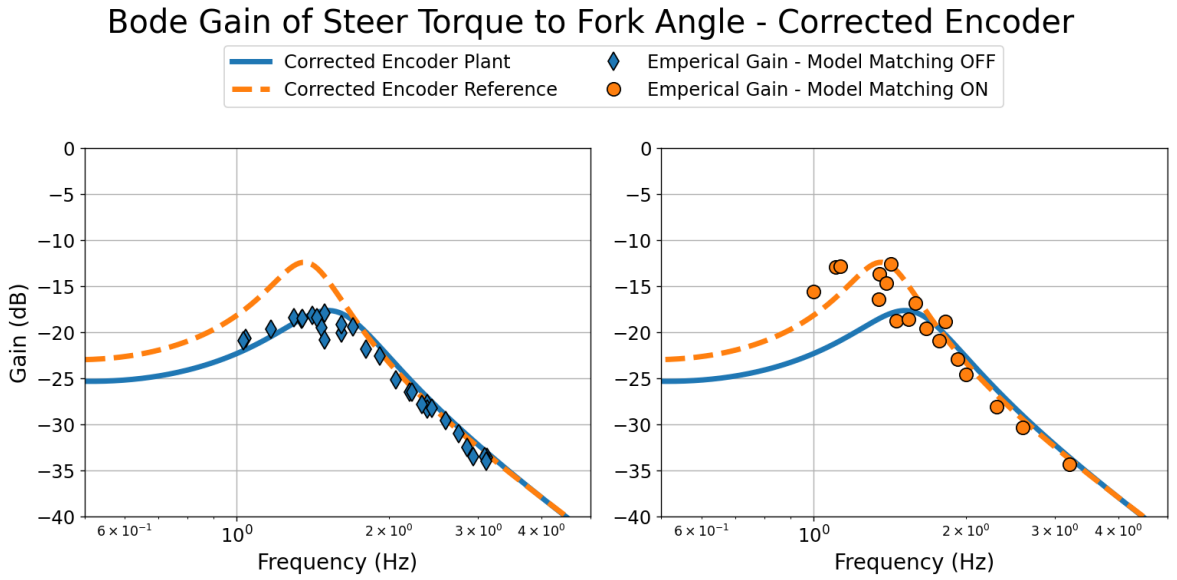

Figure C-2: Bode gain plot of steer torque to fork angle for the bicycle model with with corrected fork encoder. Here the solid blue line represents the corrected controlled system. The dashed orange line represents the model matching controller designed for the nominal system applied to the corrected system. The orange circles and blue diamonds represent the experimentally identified bode gain values for the case where the model matching controller is on and off respectively. The markers should match the lines with the same colour.

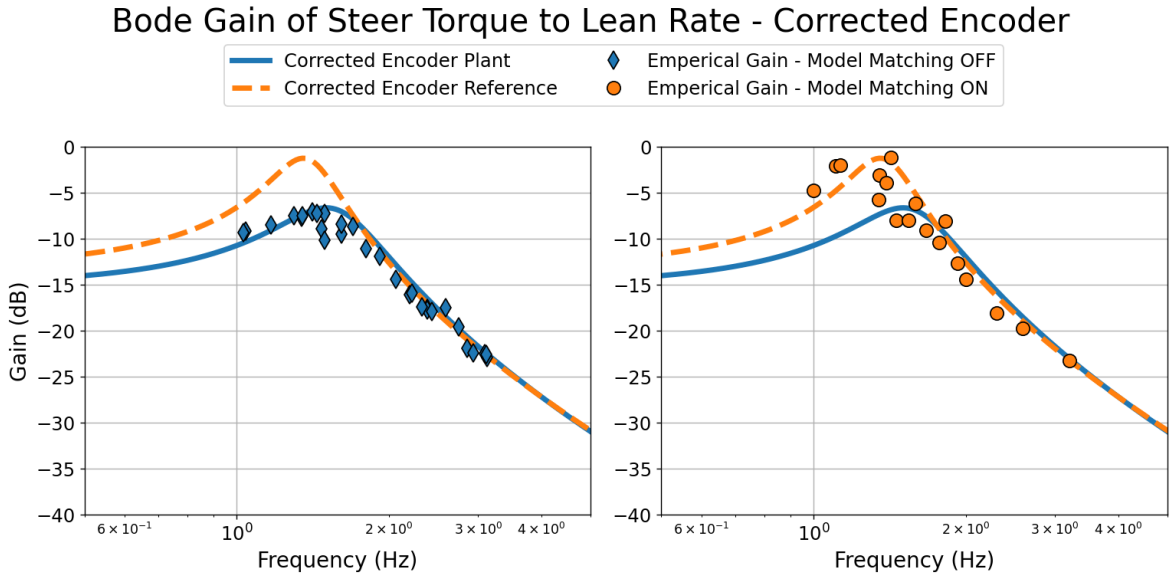

Figure C-3: Bode gain plot of steer torque to lean rate for the bicycle model with with corrected fork encoder. Here the solid blue line represents the corrected controlled system. The dashed orange line represents the model matching controller designed for the nominal system applied to the corrected system. The orange circles and blue diamonds represent the experimentally identified bode gain values for the case where the model matching controller is on and off respectively. The markers should match the lines with the same colour.

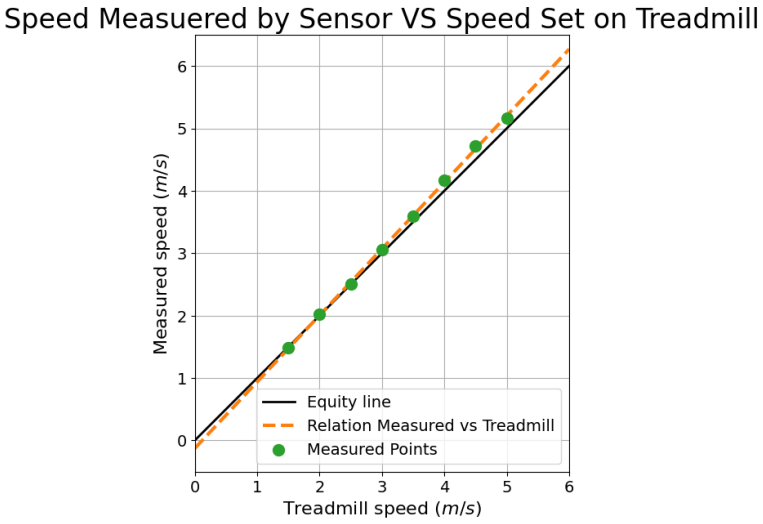

Figure C-4: Plot showing the relation between the speed measured by the speed sensor on the bicycle, and the speed indicated by the treadmill. The dots represent the discrete measurement points, and the dotted line is the result of a linear regression. The solid line is a one-to-one relationship and serves as a reference.

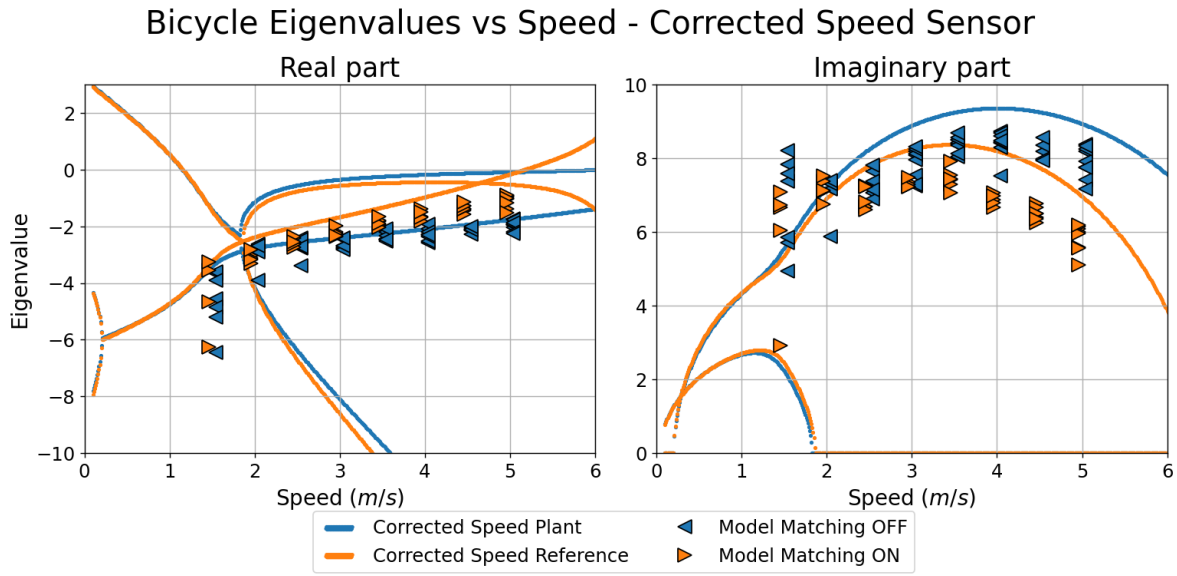

Figure C-5: Speed-eigenvalue of the bicycle model with corrected speed sensor. Here the solid blue line represents the corrected controlled system. The solid orange line represents the model matching controller designed for the nominal system applied to the corrected system. The orange right-facing and blue left-facing triangle represent the experimentally identified eigenvalues for the case where the model matching controller is on and off respectively. The markers should match the lines with the same colour.

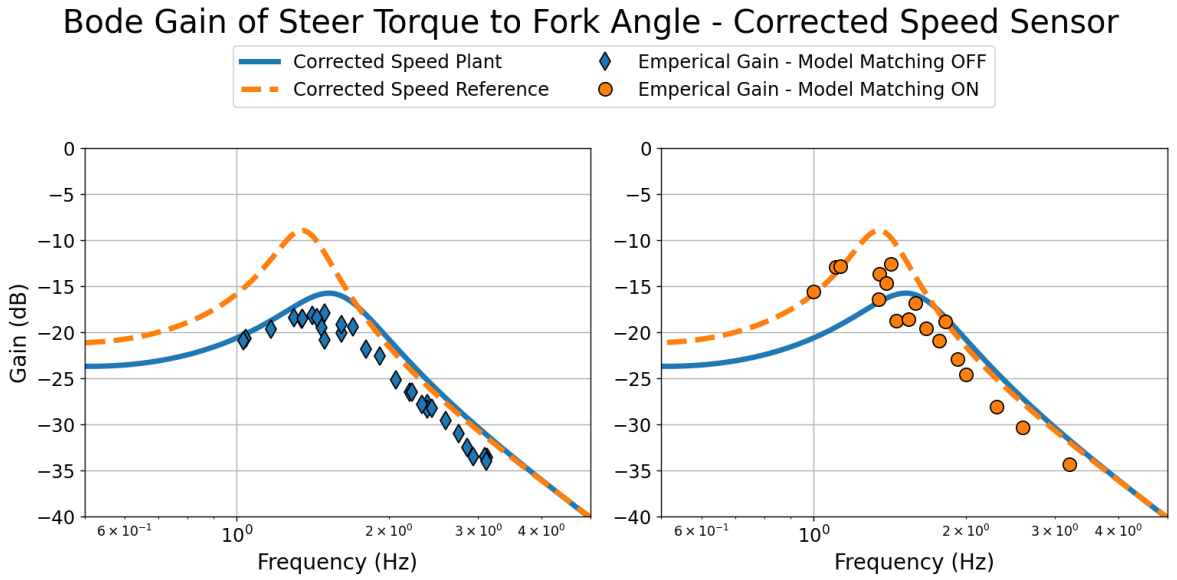

Figure C-6: Bode gain plot of steer torque to lean rate for the bicycle model with with corrected speed sensor. Here the solid blue line represents the corrected controlled system. The dashed orange line represents the model matching controller designed for the nominal system applied to the corrected system. The orange circles and blue diamonds represent the experimentally identified bode gain values for the case where the model matching controller is on and off respectively. The markers should match the lines with the same colour.

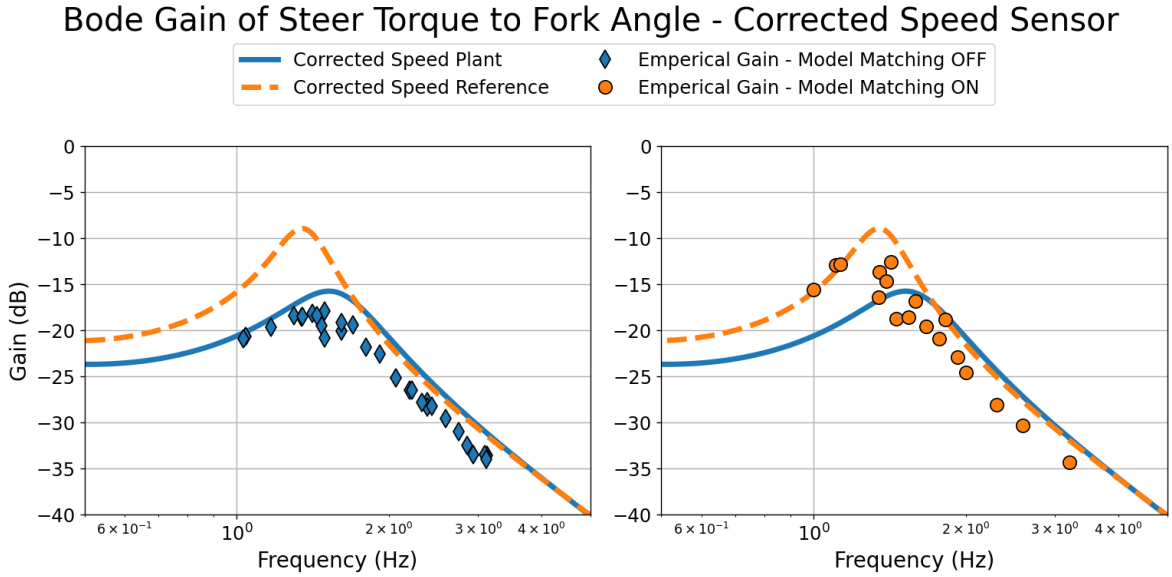

Figure C-7: Bode gain plot of steer torque to fork angle for the bicycle model with with corrected speed sensor. Here the solid blue line represents the corrected controlled system. The dashed orange line represents the model matching controller designed for the nominal system applied to the corrected system. The orange circles and blue diamonds represent the experimentally identified bode gain values for the case where the model matching controller is on and off respectively. The markers should match the lines with the same colour.

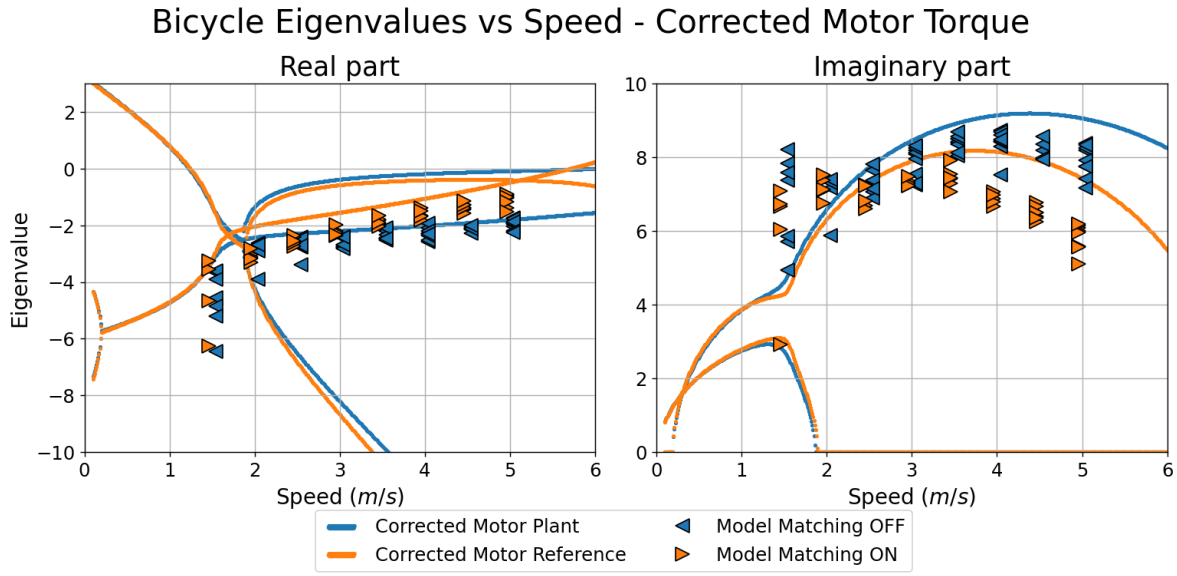

Figure C-8: Speed-eigenvalue of the bicycle model with corrected speed sensor. Here the solid blue line represents the corrected controlled system. The solid orange line represents the model matching controller designed for the nominal system applied to the corrected system. The orange right-facing and blue left-facing triangle represent the experimentally identified eigenvalues for the case where the model matching controller is on and off respectively. The markers should match the lines with the same colour.

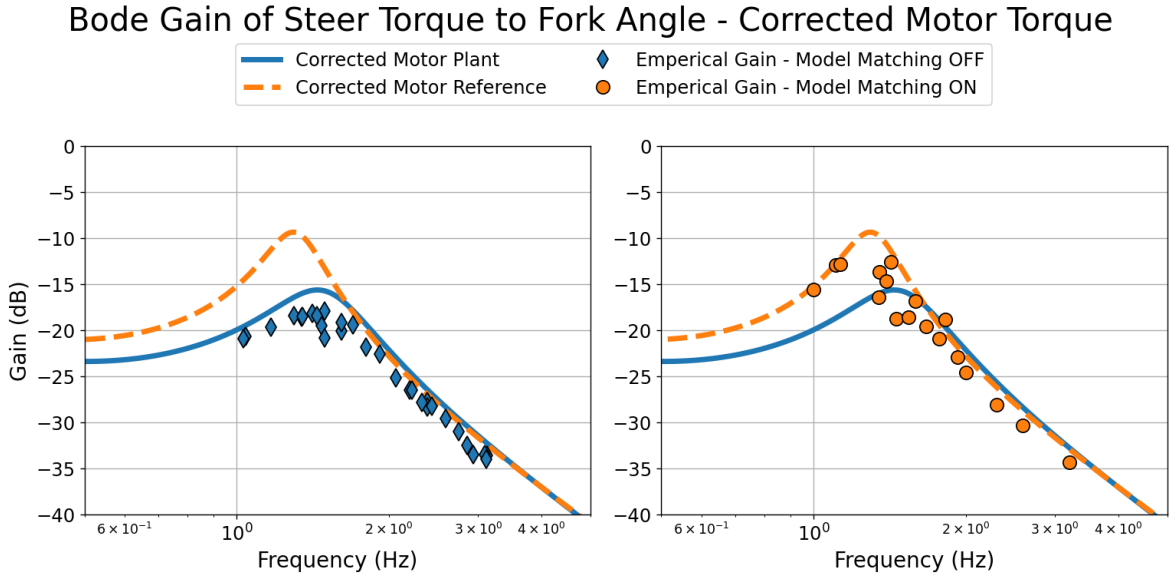

Figure C-9: Bode gain plot of steer torque to lean rate for the bicycle model with with corrected speed sensor. Here the solid blue line represents the corrected controlled system. The dashed orange line represents the model matching controller designed for the nominal system applied to the corrected system. The orange circles and blue diamonds represent the experimentally identified bode gain values for the case where the model matching controller is on and off respectively. The markers should match the lines with the same colour.

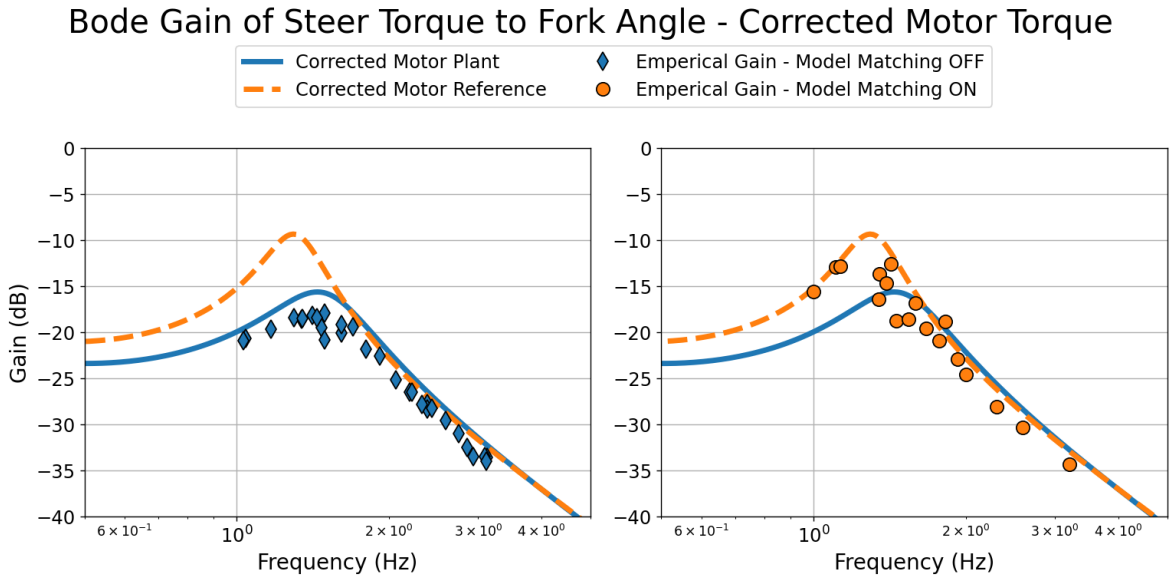

Figure C-10: Bode gain plot of steer torque to fork angle for the bicycle model with with corrected speed sensor. Here the solid blue line represents the corrected controlled system. The dashed orange line represents the model matching controller designed for the nominal system applied to the corrected system. The orange circles and blue diamonds represent the experimentally identified bode gain values for the case where the model matching controller is on and off respectively. The markers should match the lines with the same colour.

the match between measured and theoretical values for the bode gain plot. For the eigenvalue identification, the simulated plot aligns less with the measured values, especially when the model matching controller is on. A mismatch in the torques can thus only partially explain the difference between measured and theoretical eigenvalues.

C-4 The Effect of Steer Friction on the Bicycle Dynamics

A test script in the sympy python library, a computer algebra system presented in [21], creates a nonlinear Carvallo-Whipple model. This script was adapted such that the nonlinear model includes friction in the steer. A viscous friction model was applied to keep the linearisation tractable, where the friction coefficient was chosen to best fit the experimental data. Figure C-11 till Figure C-13 show the theoretical speed-eigenvalue and bode gain plot that resulted from using the new model, and show the results of applying the model matching controller, designed for the controlled system without friction, on the model with friction. In this way the simulation mimics the experiment, as the experiment had friction but did not take it into account. All figures show that friction lowers the theoretical values in both the speed eigenvalue and bode gain plot. Both the bode gain plots when the model matching controller is on and off match the experimental results better. On the other hand, the effect of friction worsens the match for the speed-eigenvalue plot. When the model matching controller is off, the experimental data no longer match their theoretical counterpart. Moreover, their trend is also no longer similar, which is also the case when the model matching controller is on. Thus, friction in the steer can lead to a closer match between experimental results and theory in some parts, but it cannot fully explain the difference.

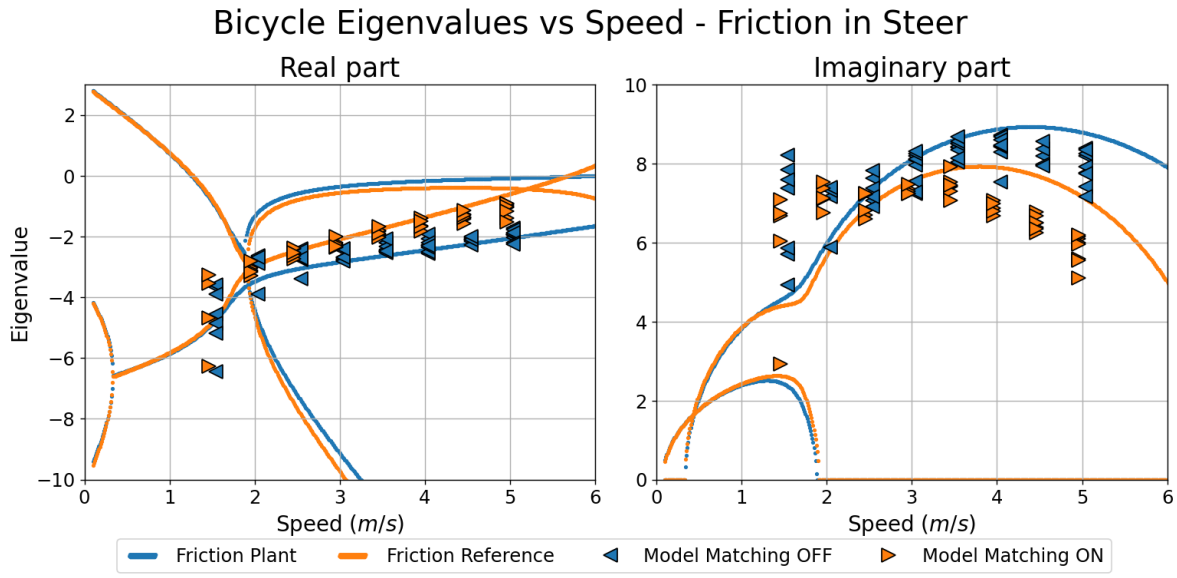

Figure C-11: Speed-eigenvalue of the bicycle model that includes friction. Here the solid blue line represents the corrected controlled system. The solid orange line represents the model matching controller designed for the nominal system applied to the corrected system. The orange right-facing and blue left-facing triangle represent the experimentally identified eigenvalues for the case where the model matching controller is on and off respectively. The markers should match the lines with the same colour.

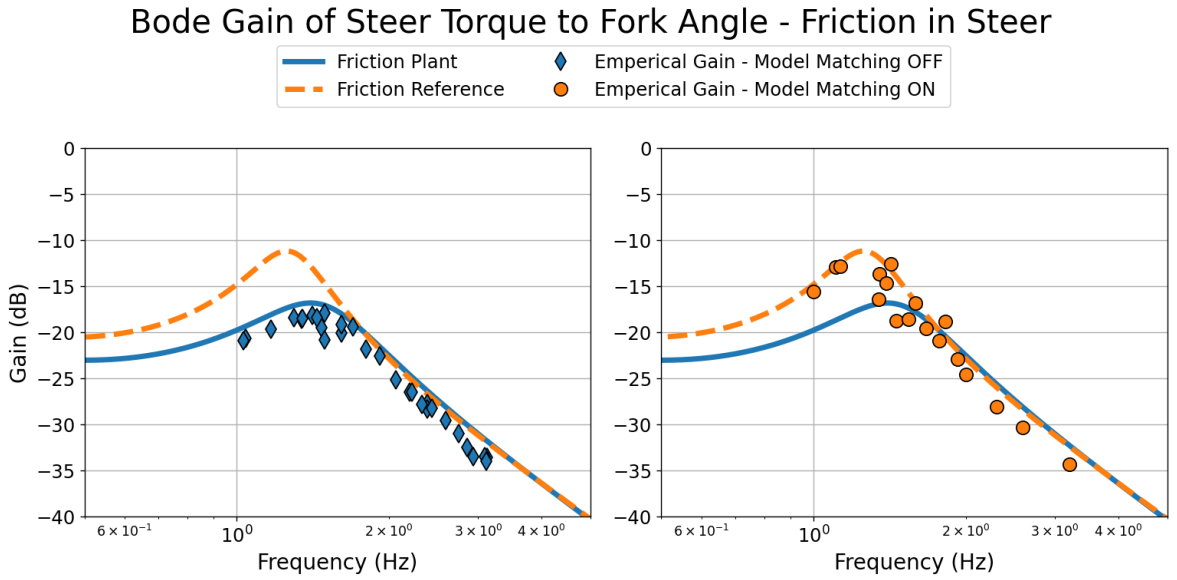

Figure C-12: Bode gain plot of steer torque to lean rate for the bicycle model that includes friction. Here the solid blue line represents the corrected controlled system. The dashed orange line represents the model matching controller designed for the nominal system applied to the corrected system. The orange circles and blue diamonds represent the experimentally identified bode gain values for the case where the model matching controller is on and off respectively. The markers should match the lines with the same colour.

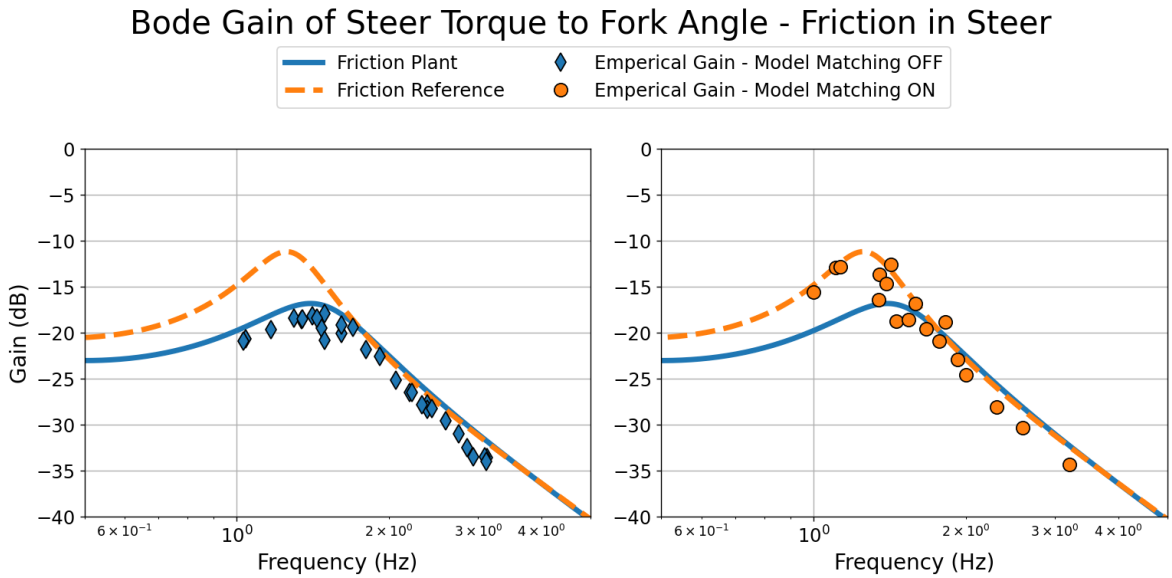

Figure C-13: Bode gain plot of steer torque to fork angle for the bicycle model that includes friction sensor. Here the solid blue line represents the corrected controlled system. The dashed orange line represents the model matching controller designed for the nominal system applied to the corrected system. The orange circles and blue diamonds represent the experimentally identified bode gain values for the case where the model matching controller is on and off respectively. The markers should match the lines with the same colour.