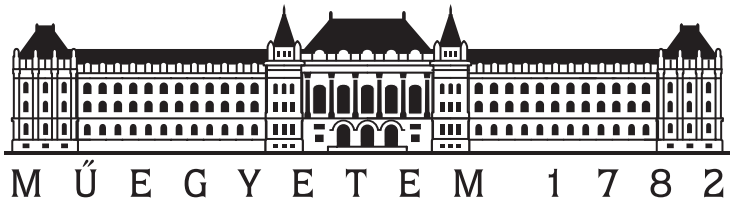
BUDAPEST UNIVERSITY OF TECHNOLOGY AND ECONOMICS
FACULTY OF MECHANICAL ENGINEERING
DEPARTMENT OF APPLIED MECHANICS



Utilizing the memory effect of elastic tires in the steering of a self-balancing e-scooter

Bendegúz Máté Györök
MSc 2nd semester

Supervisor:
Dr. Dénes Takács
associate professor

Budapest, November 2023

Contents

Abstract

The main inspiration of this paper was the self-balancing motorcycle developed by Honda [1], which was first unveiled in 2017. The motor is able to stabilize itself when standing still (with no longitudinal motion) using the Honda Riding Assist technology. The motorcycle is balanced by the steering assist system, instead of using gyroscopes, as other companies did previously. This industrial development area inspired the topic of this work, because there is an ongoing research at BME Department of Applied Mechanics to propose similar control strategies for a self-balancing vehicle.

To test the theoretical control laws, first, an experimental 2-wheeled vehicle needs to be developed. In this paper, we design a power steering system for a Xiaomi Mi Electric Scooter 3 [2], by which the fork-handlebar assembly of the scooter can be steered using a brush-less DC (BLDC) motor. Therefore, it can be used for self-balancing tasks.

In our former work [3], we focused on estimating the coefficient of friction at the steered wheel. Knowledge about the frictional relation between the road surface and the elastic tire would provide important information for the steering control, which is necessary for balancing the electric scooter. The above-specified experimental set-up made it possible to design measurements and experiments to further enhance the knowledge about the effect of friction forces on the accuracy of steering control.

Based on measurement data, we aim to enhance the accuracy of our delayed tire model, which describes the relation between the ground and the elastic tire. The simulation results are compared to the acquired experimental data. Hence, we lay out the foundation of a further improved friction estimation method.

Kivonat

A legnagyobb inspirációt a dolgozat témájához a Honda által 2017-ben piacra dobott önegyensúlyozó motorkerékpár [1] nyújtotta. Ez a motor képes arra, hogy stabilizálja magát egy helyben állás közben az ún. Honda Riding System használatával, ami annyit takar, hogy csupán az első kerék kormányzásával képes egyensúlyozni magát, a korábbi giroszkópos megoldásokkal ellentétben. Ez az ipari fejlesztés inspirálta ezen dolgozatot, hiszen a BME Műszaki Mechanikai Tanszéken egy jelenleg is futó kutatási projekt célja egy olyan szabályozási metódus kidolgozása, amely ugyanígy képes a jármű egyensúlyozására.

Ahhoz, hogy a kidolgozott elméleti irányítási sémákat valós berendezéseken is tesztelni lehessen, először egy kísérleti kétkerekű jármű fejlesztése szükséges. Munkánk során egy speciális berendezést terveztünk, amelyet egy Xiaomi Mi Electric Scooter 3 típusú elektromos rollerre [2] szereltünk fel. Ezzel a berendezéssel a jármű első villája elfordítható egy kefenélküli DC (BLDC) motor használatával, vagyis az önegyensúlyozáshoz szükséges mozgás létrehozható.

Egy korábbi munkánkban [3] a súrlódási együttható becslésére irányuló eljárásokat dolgoztunk ki. Az útburkolat és a gumikerék közötti súrlódási viszonyok ismerete fontos információt nyújtana a kormányrendszer szabályozásához, ami szükséges az önegyensúlyozó mozgás megvalósításához. A korábbiakban megemlített kísérleti berendezés lehetővé tette, hogy méréseket és kísérleteket végezzünk annak érdekében, hogy tovább növeljük ismereteinket a súrlódás hatásáról a kormányrendszer pontosságára.

A dolgozat során a fő céljaink voltak, hogy mért adatok alapján pontosítsuk a talaj és gumikerék között lezajló folyamatok leírását. Az így kapott szimulációs modellt felhasználva, összehasonlíthatóak a szimulációs és a kísérleti eredmények, ezzel megalapozva egy még pontosabb becslési eljárás kidolgozásának lehetőségét.

1 Introduction

The main inspiration of this work was the self-balancing motorcycle developed by Honda [1] (illustrated in Figure 1.1), which was first unveiled in 2017. The motor can stabilize itself when standing still (with no longitudinal motion) using the Honda Moto Riding Assist technology. Knowledge about the frictional relation between the road surface and the elastic tire would provide important information for such steering control tasks.

Figure 1.1: Self-balancing motorcycle developed by Honda[1].

For other various vehicle motion control tasks, accurate real-time information about the tire-ground relationship would be also essential. However, the friction properties are hard to measure, therefore it requires special methods to estimate the coefficient of friction. In the literature, several methods can be found, but none of them provides an accurate, fast, and robust technique, thus the development of an algorithm, that fulfills the above-mentioned criterion, is an important topic. The self-aligning torque induced in the contact patch of a steered tire is in close connection with the coefficient of friction. This phenomenon offers a possibility to estimate the friction properties based on the measured steering torque.

The self-balancing control of motorcycles would also benefit greatly from an accurate friction estimation since the self-aligning torque of the steered tire could be utilized to stabilize the vehicle. The balancing of electric scooters presents similar dynamical problems as with motorcycles, but the purchasing of an experimental electric scooter is much more cost-effective, therefore it is ideal for research purposes.

An accurate simulation environment is also essential to model the dynamic behavior

[1]Source: `https://www.bikesrepublic.com/` (accessed: 2023-11-05)

of the contact patch. In the future, various balancing control algorithms will need to be tested, and as the industrial pattern shows, testing in a simulation environment is more cost- and time-effective, because the faults of the control algorithms can be eliminated before testing on the real-world equipment. For accurate simulation of the tire dynamics, the tire model should be carefully selected, especially for cases with zero longitudinal velocity.

This work aims to investigate the relationship between the tire and ground to provide important information for various applications, especially focusing on the self-balancing e-scooter. The main goals of this work are

(i) enhancing the accuracy of modeling and simulation of the contact-patch dynamics considering zero longitudinal velocity,

(ii) develop and test estimation methods, that aim to provide accurate information about the friction properties,

(iii) to develop and build an experimental tool for a self-balancing electric scooter.

1.1 Structure of the paper

The paper is categorized into different sections, as follows. In Section 1, the goals and motivations of the work are defined and the existing scientific literature is summarized in the topic of friction estimation.

Section 2 presents the applied tire models. Based on previous measurements, the model parameters are identified and tested for the brush tire model. A modification is presented for the brush model, which makes it possible to consider different static and dynamic coefficients of friction. The models are evaluated and compared with open-loop simulations and the consequences are drawn.

In Section 3, we deal with the estimation of the coefficient of friction. First, a previously designed analytic method is discussed with its advantages and drawbacks, and then a machine learning-based condition is developed. If the condition is fulfilled, the analytic method provides the estimated friction value to the controller, or else the estimation is neglected. This method is suitable for the self-balancing type of applications when the friction properties do not change so abruptly (the main causes of changes are temperature dependency and tire wear). And lastly, a purely numerical estimation method is presented, which is based on the brush model. The models are evaluated on measurement data and compared based on different aspects.

In Section 4, the design process of the experimental electric scooter is discussed. From the schematic idea of the equipment to real-world experiments, the development steps are detailed in this Section.

Lastly, in Section 5, we summarize the achieved results, draw the conclusions, and frame the future development plans.

1.2 Overview of the literature

Just as the mentioned self-balancing motorcycle, other problems in vehicle dynamics would benefit greatly with accurate estimation of the frictional properties in the contact region of road surface and elastic tires. Knowledge about the coefficient of friction makes vehicle control systems more robust and accurate, therefore several studies have been proposed, in order to develop an estimation method for the friction parameters of tires. In the literature, several estimation approaches can be found, that can be categorised into two groups various groups based on their working principles. Different methods require different sensors and calculation methods, thus each estimation process has different drawbacks and advantages, which are summarized in Table1.1.

Working principle	Advantages	Drawbacks
Image recognition	Can predict friction properties ahead	Bad lighting decreases reliability
Based on ambient temperature	Easy to implement	Only gives a warning
Vibration model	No special sensors needed	High frequency noises
Vehicle dynamical calculations	Easy implementation to ADAS	No known drawbacks
Based on self-aligning torque	Easy implementation to EPAS	No known drawbacks

Table 1.1: Summarized estimation methods for the coefficient of friction, that can be found in the literature.

In general, vision-based predictions have several possibilities in store for the future, with rapidly improving vision recognition softwares, but currently, their most important disadvantage is that the accuracy of the prediction is enormously worse during bad lighting conditions [4]. Sound measurements can be also applied to improve the reliability of vision-based systems, but it still produced several drawbacks [5].

The road conditions can also be estimated based on the ambient temperature [6]. This technique does not provide an exact value for the coefficient of friction, but can be used as a warning system for the driver. The advantages of the image-, sound- or temperature-based methods are that the friction properties of the road surface can be predicted ahead, thus, important information can be supplied to the control systems of the vehicle (e.g. ABS control or even the steering assistant control system). Another advantage of these techniques is that the coefficient of friction can be estimated without physical excitation,

but they are not taking into account all the factors to provide an accurate estimation for the coefficient of friction, for example, knowledge about the tire: tire type, tire pressure, and so many more aspects are not utilized.

A popular approach for friction estimation is to apply a vibration-model, which monitors the vertical position of the tire as a function of the longitudinal velocity [7]. These methods can be suitable for rolling tires, but for the self-balancing application, the longitudinal dynamics do not need to be considered. Apart from that, this method suffers from high frequency noises.

Calculations based on vehicle dynamics can also provide an estimation of the coefficient of friction [8]. For example, larger slip angle occurs on slippery roads (e.g. on ice), than during normal operation. Estimation techniques based on vehicle dynamics usually measure yaw rate, lateral acceleration, and vehicle speed. Most researchers focus on describing the lateral dynamics of the vehicle by using some kind of vehicle model, while the friction forces are taken into account by using a tire model. Although vehicle dynamics-based estimation approaches are interesting and promising fields of research, in this work, the goal is to provide an estimation algorithm for vehicles with zero longitudinal velocity, therefore these methods could not be applied.

Along with vehicle dynamical measurement signals, the self-aligning torque of tires is recently used in the control schemes of some driver assistant systems. Electronic power-assisted steering (EPAS) made the real-time calculation of the self-aligning torque possible. Previously both Mitsubishi [9] and Toyota [10] released publications about calculating the self-aligning torque based on available signals of the EPAS system. However friction estimation methods based on the self-aligning torque often assume moving vehicles and the states are estimated with the help of a vehicle model.

The found methods either used signals, that were unavailable in the investigated case or were not derived for cases with zero longitudinal velocity, therefore it was necessary to develop a new estimation process. In a previous work [3], we proposed an analytic estimation method based on the brush tire model, with zero longitudinal velocity assumed. This dynamical system is similar to the self-balancing application, thus the results of this work can be useful for future application. However, the previously proposed method had a few drawbacks that need to be addressed before applying to real-world problems, thus we aim to find the solutions to these problems in this paper.

2 Modeling and simulation of the tire deformation in the contact patch

The largest challenge for developing a self-balancing electric scooter is that the scooter should be able to balance itself with zero longitudinal velocity. The tire deformation causes forces and moments that can be utilized to stabilize the system, but it is especially hard to model these phenomena. The coefficient of friction between the scooter's tire and the ground affects the tire forces largely, therefore estimating its value could enhance the performance of the controller. Another aspect is that a simple control law (that does not deal with the estimated coefficient of friction) could be tested with a simulation environment that considers a realistic tire-ground connection. Hence, in this section, the modeling and simulation of the tire forces and moments are discussed.

2.1 The brush tire model

There are different approaches to model the tire-road relationship. Starting from the 1940s, the vehicle industry showed more and more interest in tire modeling, therefore the first mechanical models were introduced in that period [11]. Later on, researchers focused more on finding empirical formulas that describe the tire forces. Nowadays, with the advanced computation efforts, combined tire models are being introduced, using both physical models and empirical formulas, moreover, finite element methods are used to validate these modern theories.

For the investigations, the brush tire model was selected [12], which is a popular analytical approach to model torques and forces resulting from the deformation of tires. The main advantage of analytic approaches is that they are based on physical principles, therefore no experimental data is needed to describe the system. The schematic view of the brush tire model can be seen in Figure 2.1. The model consists of tread elements (also called bristles) connected to the carcass. The carcass is represented as a rigid disk. The thread elements are modeled as linear springs and their deformation is independent from each other, therefore outside of the contact patch, zero deformation can be assumed.

An in-plane model of the brush model can be defined, where the deformation of the elastic bristles is calculated in the $x-y$ plane, as shown in Figure 2.1 and 2.2. The x axis is aligned to the longitudinal, and y is to the lateral direction of the tire, thus the $x-y$ coordinate system rotates with the carcass. Inside the contact patch, the deformation is limited by the available friction between the treads and the road surface. Assuming parabolic normal force distribution (which assumption was proven by measurements in [13]) in the contact region, the limit of the deflection (q_{cr}) is also parabolic

$$q_{\mathrm{cr}}^{\pm}(x) = \pm \frac{3F_{\mathrm{N}}\mu}{4a^3k}\left(a^2 - x^2\right), \tag{2.1}$$

where a is the contact patch half-length, F_{N} is the normal force, μ is the coefficient of

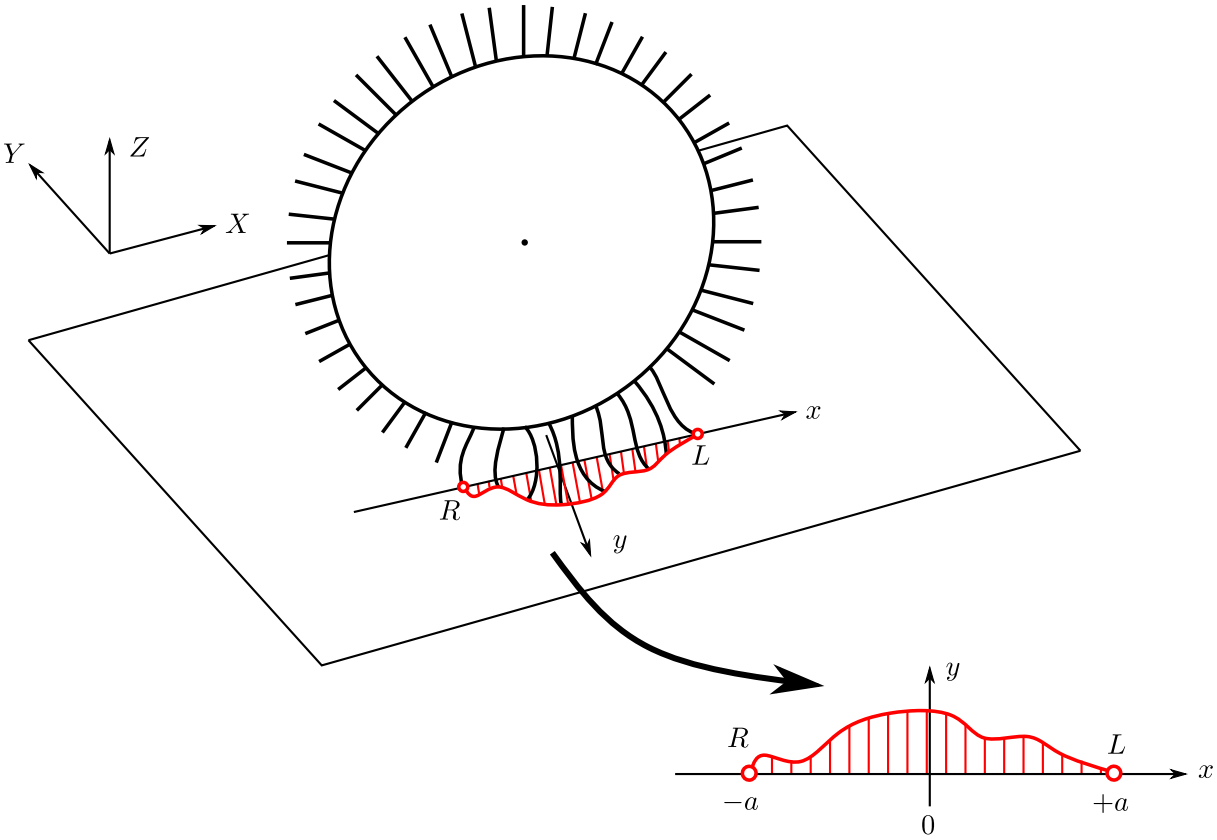


Figure 2.1: Schematic view of the brush tire model. The leading edge is noted by L and the rear edge by R. The in-plane deformation within the contact patch is also illustrated.

friction, and k [N/m^2] is the distributed lateral stiffness of the elastic bristles (it is assumed to be homogeneous around the carcass).

Originally, the brush tire model was developed to model tire forces in case of non-zero longitudinal velocity. In this work, the concept is used to model tire deformations in case of zero longitudinal speed, to model the dynamical situation of the balancing task. In Figure 2.2, the deformation of an elastic tire is illustrated by the original idea with non-zero longitudinal speed. The sliding and sticking regions within the contact patch are also shown in Figure 2.2. Tire elements enter the contact patch at the leading edge L with zero deformation. In the quasi-stationary case, the brush elements start to be deformed according to the side slip angle α, which measures the angle of the wheel center point velocity relative to the wheel center plane. When the deformation of an element reaches the limit of deflection, it enters the sliding zone, and its deformation is limited by the previously defined q_{cr}. In an arbitrary position x, the deformation of the tread elements (q) can be expressed as

$$q(x) = \begin{cases} q_{\mathrm{cr}}^{\pm}(x), & \text{when sliding,} \\ q_{\mathrm{r}}(x), & \text{when sticking.} \end{cases} \tag{2.2}$$

The deformation of the elements causes a torque, namely the self-aligning torque, that acts against the steering torque. Assuming the deformation function is known for $t \in$

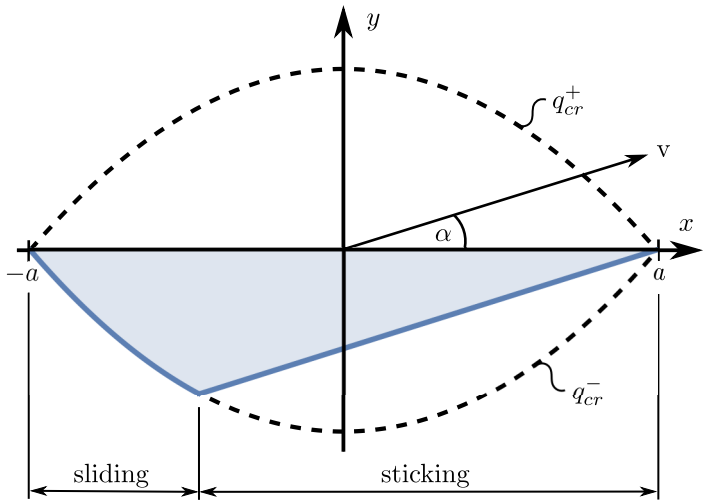


Figure 2.2: In-plane model of the tire brush model, in case of non-zero longitudinal velocity.

$[t_0, t_{\text{end}}]$, the self aligning torque can be expressed as

$$M_z(t) = \int_{-a}^{a} k \cdot q(x,t) \cdot (x - e)\ \mathrm{d}x, \tag{2.3}$$

where e is the caster length (also called as trail).

2.1.1 Numerical simulations based on the brush model

For the investigation of the tire-ground relation, a simulation environment was set up in [3] based on the brush model. Based on Figure 2.3, let us consider an arbitrary point P in the sticking region of the contact patch. The wheel is towed by the rigid caster with constant (zero) velocity in the X direction. The lateral deformation of the tire can be described with $q(x,t)$ function. It is known that the kinematic constraint of sticking requires the absolute velocity to be zero, therefore $\mathbf{v}_P = \mathbf{0}$.

Using the notations of relative kinematics

$$\mathbf{v}_{P20} = \mathbf{v}_{P10} + \mathbf{v}_{P21}, \tag{2.4}$$

where $\mathbf{v}_{P20}$ is the absolute velocity of point P, $\mathbf{v}_{P10}$ is the carrier velocity and $\mathbf{v}_{P21}$ is the relative velocity.

The absolute velocity of the carcass, or in other words, the carrier velocity of the tire, can be expressed as

$$\mathbf{v}_{P10} = \mathbf{v}_{A10} + \omega_{10} \times \mathbf{r}_{AP}. \tag{2.5}$$

Since the towing speed is zero in our study, therefore $\mathbf{v}_{A10} = \mathbf{0}$. As a result, $\mathbf{v}_{P10}$ can be expressed in the $(x,\ y,\ z)$ coordinate system as

$$\mathbf{v}_{P10} = \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} \times \begin{bmatrix} -(e-x) \\ q(x,t) \\ 0 \end{bmatrix} = \begin{bmatrix} -q(x,t) \cdot \dot{\psi} \\ -(e-x) \cdot \dot{\psi} \\ 0 \end{bmatrix} \tag{2.6}$$

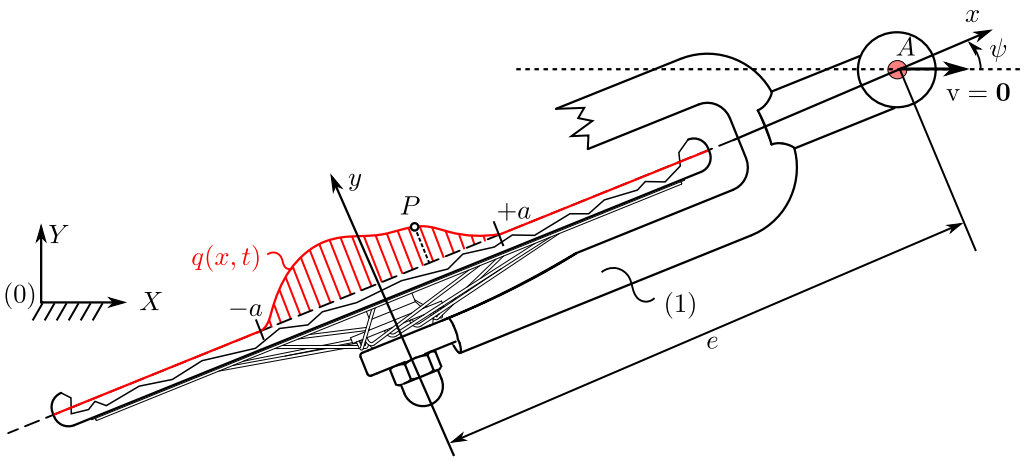


Figure 2.3: Illustration of the mechanical system for deriving the kinematic constraint of sticking.

The relative velocity between the tire and the carcass can be attained by the time derivation of the relative position vector $\mathbf{r}_{P21}$, as

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{r}_{P21} = \frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix} x \\ q(x,t) \\ 0 \end{bmatrix} = \begin{bmatrix} \dot{x} \\ \dot{q}(x,t) + q'(x,t)\cdot\dot{x} \\ 0 \end{bmatrix}, \tag{2.7}$$

where $\dot{\square}$ notes the partial derivative with respect to time and $\square'$ notes the partial derivative with respect to coordinate x. Thus the time derivative of $q(x,t)$:

$$\frac{\mathrm{d}}{\mathrm{d}t}q(x,t) = \underbrace{\frac{\partial}{\partial t}q(x,t)}_{\dot{q}(x,t)} + \underbrace{\frac{\partial q(x,t)}{\partial x}}_{q'(x,t)} \cdot \underbrace{\frac{\partial x}{\partial t}}_{\dot{x}}. \tag{2.8}$$

From (2.7), the kinematic constraint equations of sticking can be expressed as

$$-\,q(x,t)\cdot\dot{\psi} + \dot{x} = 0, \tag{2.9}$$

$$-\,(e-x)\cdot\dot{\psi} + \dot{q}(x,t) + q'(x,t)\cdot\dot{x} = 0, \tag{2.10}$$

where $x \in [-a,\, a]$, assuming the whole contact region is sticking.

From (2.9), the translational velocity can be determined:

$$\dot{x} = q(x,t)\cdot\dot{\psi}(t). \tag{2.11}$$

From (2.10), the differential equation for the lateral deformations of the elastic tire elements can be expressed as

$$\dot{q}(x,t) = (e-x)\cdot\dot{\psi}(t) - q'(x,t)\cdot q(x,t)\cdot\dot{\psi}(t), \tag{2.12}$$

where the last term is nonlinear. Considering small deformations and motions, it represents a third-order small term that can be neglected in the analysis.

The equation of motion for the caster-wheel system can be expressed using Newton's second law, as

$$J_A\ddot{\psi}(t) = -b_t\dot{\psi}(t) - M_z(t) + M(t), \tag{2.13}$$

where J_A is the mass moment of inertia with respect to point A, b_t is the torsional damping of the kingpin, M_z is the self-aligning torque of the tire and M is the steering torque.

The governing equation of the system consists of the integro-differential equation (2.13), where the self-aligning moment can be calculated by (2.3), while the lateral deformation is described by the partial differential equation (2.12). Moreover, the sliding in the contact patch is also considered since the deformation is maximized within the contact patch based on (2.1). The equations necessary for the numerical simulation can be summarized as

$$J_A\ddot{\psi}(t) = -b_t\dot{\psi}(t) - \int_{-a}^{a} k \cdot q(x,t) \cdot (x-e)\,\mathrm{d}x + M(t), \tag{2.14}$$

$$\dot{q}(x,t) \approx (e-x) \cdot \dot{\psi}(t), \tag{2.15}$$

$$|q(x,t)| \leqslant \frac{3F_N\mu}{4a^3k}\left(a^2 - x^2\right), \forall\, t \in [t_0,\, t_{\mathrm{end}}]\,. \tag{2.16}$$

A simulation environment has been programmed based on the derived equations, which can be used with arbitrary steering torque input, and various steering control methods can be tested as well.

2.2 Experimental validation of the brush tire model

In our previous work [3] we constructed experiments to validate various estimation methods. Although the acquired measurement data was used for testing the proposed estimation method which is based on the brush tire model, the model itself hasn't been validated. Hence in this section, we compare the measured signals with simulations that were discussed in Section 2.1.1.

At the Department of Applied Mechanics of BME, an experimental rig has been developed to investigate the dynamical properties of towed wheels [14]. This already established experimental set-up was used with some modifications that can be seen in Figure 2.4. The investigated bicycle wheel is mounted in a way, that allows the modifications of caster length e. The caster can be rotated about the rigid kingpin, where oil lubrication ensures that the effect of dry friction can be neglected, and low torsional damping is expected. The length of the contact patch can be modified by changing the vertical distance between the kingpin and the "road" surface. The vertical load F_z and the self-aligning torque M_z were measured with a Kistler 9129AA Multicomponent Dynamometer using a Kistler Type 5080A charge amplifier. The yaw angle $\psi(t)$ was measured with a magnetic rotation sensor placed at the joint (AMS AS5048A magnetic rotary encoder, 14-bit resolution). For detecting and digitizing the measured angle signals, a PIC microcontroller was used (part of the experimental rig). The measurement data was gathered using a

National Instruments USB-6216 (BNC) data acquisition device, but the measured signals were processed in MATLAB.

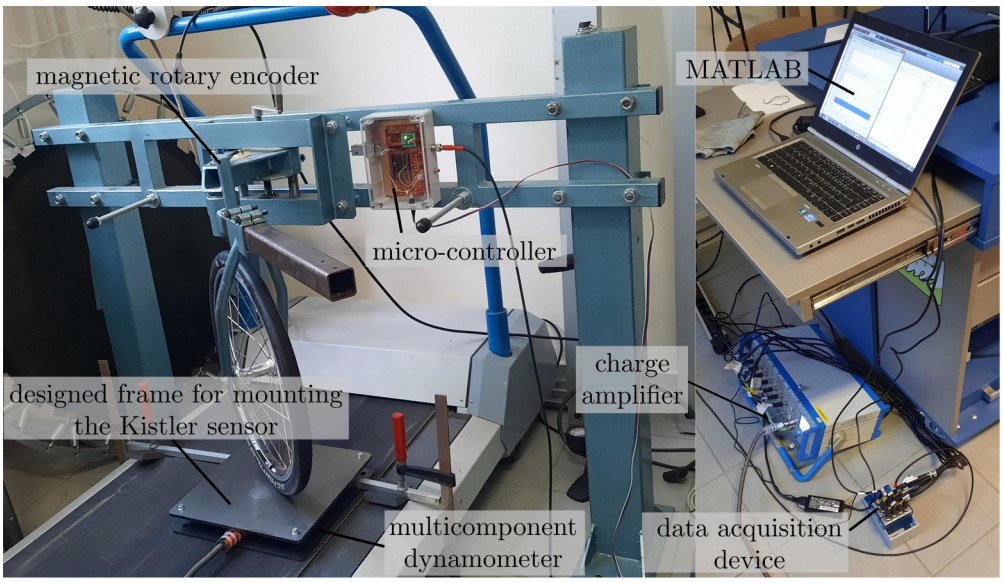


Figure 2.4: Experimental rig with the used devices.

Before the validation measurements, a few parameters of the system had to be identified. The towing length can be measured on the experimental rig, but for all measurements, $e = 0.009$ m was set. The length of the contact patch was measured with a vernier caliper. The yaw angle, vertical load, and self-aligning torque were gathered with the dynamometer as previously discussed.

The only remaining parameter of the system is the distributed lateral stiffness. In practical applications, this would be hard to measure directly, but since this value is dependent on the tire pressure, a good estimation could be given by the manufacturer. In this work, we identified k with a simple experiment that is detailed in [3].

The model depends also on the coefficient of friction between the tire and the ground, thus this parameter also had to be identified. A simple experiment was applied: a wooden object with rubber material glued onto the surfaces was towed on the measured plates while measuring the forces with the Kistler Multicomponent Dynamometer. The rubber material was chosen to be similar to the elastic rubber of the tire. The experiment was carried out with different normal forces to estimate the coefficient of friction. In the brush tire model, the static and dynamic friction properties are assumed to be the same, thus the measured data is fitted as a function expressed as

$$F = \mu N, \tag{2.17}$$

where $F = \sqrt{{F_x}^2 + {F_y}^2}$ and the normal force $N = F_z$.

The identified parameters of the system are summarized in Table 2.1. An open-loop simulation has been developed, which means that ψ, $\dot{\psi}$ and F_z are determined from the measurements. With the constants summarized in Table 2.1 only the deformation function

and the self-aligning torque are calculated from the model. The results are shown in Figure 2.5.

Parameter	Notation	Value	Dimension
Coefficient of friction	μ	1.9003	1
Contact patch half-length	a	0.04	m
Distributed lateral stiffness	k	$2.438 \cdot 10^6$	$\mathrm{N/m^2}$
Mass moment of inertia with respect to A	J_A	0.1518	$\mathrm{kgm^2}$
Torsional damping	b_t	6.8509	Nms
Towing length	e	0.009	m

Table 2.1: System parameters during the experiments.

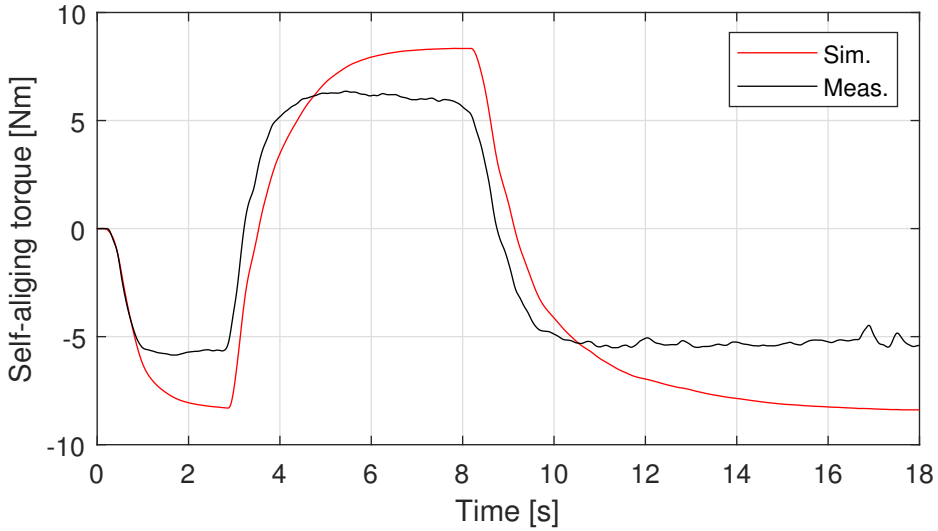


Figure 2.5: The results of the measurement and the open-loop simulation comparing the self-aligning torque.

It is visible in Figure 2.5, that the measured self-aligning torque saturates around ± 6 Nm, but the simulated signal does not show such characteristics. It is more visible in Figure 2.6, where the measured and simulated values are shown in the $\psi - M_z$ plane.

It is more visible in 2.6 that the measured and simulated signals show similar characteristics, but above a specific value, the measured self-aligning torque does not grow further compared to the simulation. One of the possible reasons is that this version of the brush tire model does not take into account that the slipping coefficient of friction is different from the sticking one. At large steering torque inputs the majority of the contact patch is sliding, thus the model becomes more inaccurate.

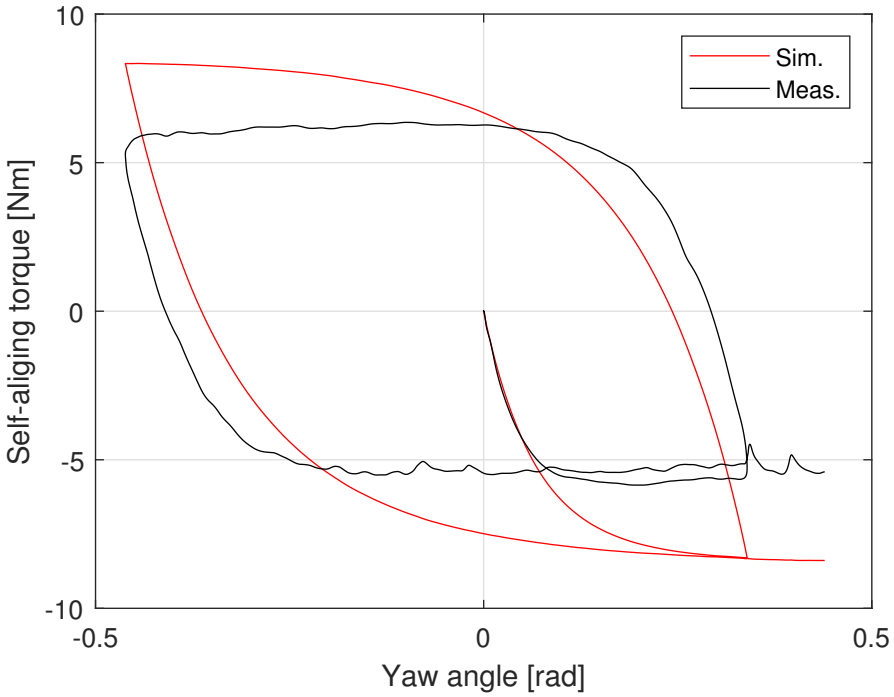


Figure 2.6: The measured and simulated self-aligning torque is compared in the $\psi - M_z$ plane.

2.3 Improved brush tire model

As discussed in Section 2.2, one of the main disadvantages of the used brush tire model, is that the static and dynamic coefficient of friction is assumed to be equal. The static coefficient of friction affects the maximum displacement $q_{\mathrm{cr}}^{\pm}$ for which the thread elements start sliding, while the dynamic coefficient of friction affects the maximum displacement $q_{\mathrm{slip}}^{\pm}$ for the thread elements that are sliding. Noting the dynamic coefficient of friction with μ_{d}, the maximum displacement for the sliding thread elements can be expressed as

$$q_{\mathrm{slip}}^{\pm}(x) = \pm \frac{3F_{\mathrm{N}}\mu_{\mathrm{d}}}{4a^3k}\left(a^2 - x^2\right). \tag{2.18}$$

A few examples can be found in the literature, when the static and dynamic coefficients of friction are different in the brush model, for example in [13]. Although this improved tire model has been published, it has not been tested for zero longitudinal velocity. For a previously used example – when the longitudinal velocity is not zero – the proposed improvement can be illustrated as in Figure 2.7

Let us note the solution of the differential equation (2.12) with $\hat{q}(x,t)$ for any $t \in [t_0,\, t_{\mathrm{end}}]$. In general, for any arbitrary t, the following statements are true, thus the time dependency is not noted. It has previously shown that for any x if $|\hat{q}(x)| \geq q_{\mathrm{cr}}(x)$ then the tire is sliding at x. In other words

$$q(x) = \begin{cases} \hat{q}(x), & \text{if } |\hat{q}(x)| \leq q_{\mathrm{cr}}(x), \\ q_{\mathrm{cr}}^{\pm}(x), & \text{otherwise.} \end{cases} \tag{2.19}$$

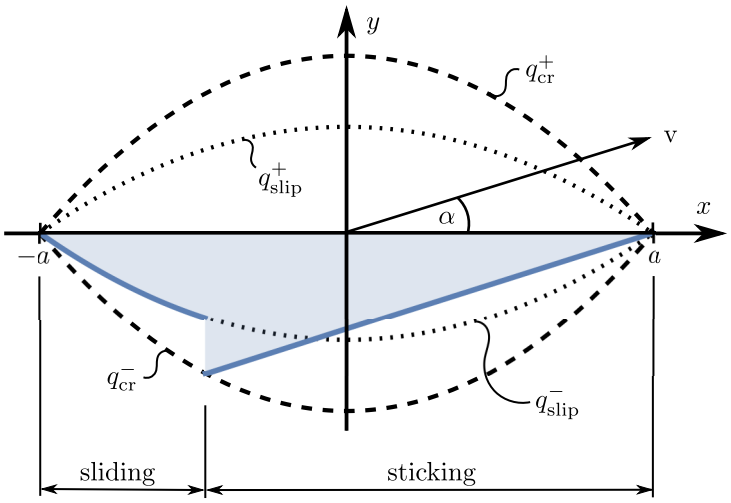


Figure 2.7: In-plane model of the improved brush model, in case of non-zero longitudinal velocity.

With considering different static and dynamic friction properties (2.19) can be reformulated as

$$q(x) = \begin{cases} \hat{q}(x), & \text{if } |\hat{q}(x)| \leq q_{\mathrm{cr}}(x), \\ q_{\mathrm{slip}}^{\pm}(x), & \text{otherwise.} \end{cases} \tag{2.20}$$

This formulation results in a non-smooth deformation function, which is heavily influenced by the resolution of the spatial discretization. All in all, with this improved model, more accurate results can be achieved by comparing to the measurement data.

The problem is that with the identification measurements, only the static coefficient of friction has been determined, thus the dynamic one still needs to be identified. Fortunately, three different measurements were made with the parameters in Table 2.1, thus one of them can be used to fit the dynamic coefficient of friction. The previously discussed measurements have been performed with two different surfaces The identified static and dynamic coefficients of frictions are summarized in Table 2.2.

Surface	μ_{s} [1]	μ_{d} [1]
Metal plate	1.674	1.467
Red emery paper	1.9003	0.676

Table 2.2: The measured static coefficient of friction noted with μ_{s} and the identified dynamic coefficient of friction μ_{d}.

With the same parameters as in Table 2.1 and the same measurement signals used for the open-loop simulation as in Figure 2.5 and 2.6, the improved model response can be seen in Figure 2.8.

The improved model can be also illustrated in the $\psi - M_z$ plane in Figure 2.8, where it is more visible, that different dynamic coefficient of friction results in qualitatively more accurate representation of the system.

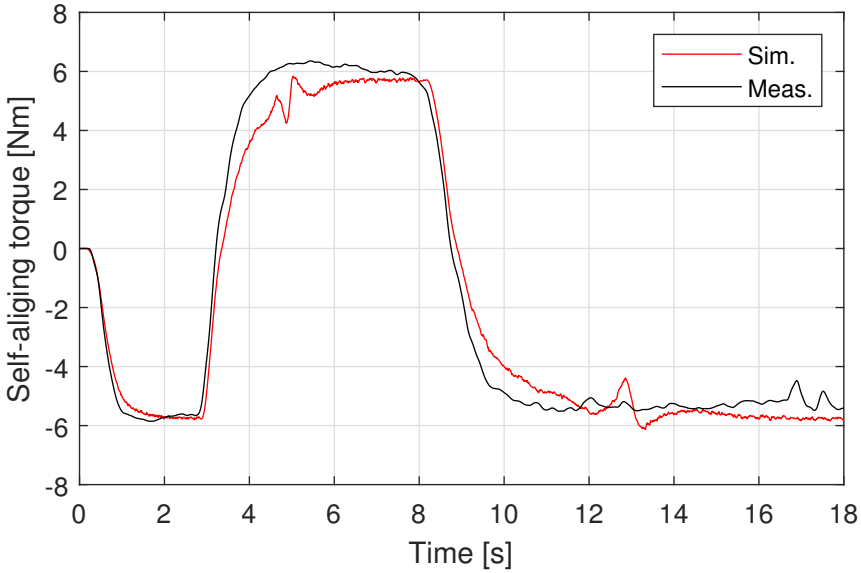


Figure 2.8: The results of the measurement and the open-loop simulation with the improved brush tire model. The self-aligning torque can be compared in the time domain.

With both of these surfaces (shown in Table 2.2) multiple measurements have been made, varying the vertical load, thus the contact patch length. With the metal plate 6, with the red emery paper 7 different measurement signals are available for testing the method., but one of each data-series have been used for fitting the dynamic friction. The accuracy of the simulation is characterized by the root-mean-squared error (RMSE) metric for the self-aligning torque:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(M_z[i] - \hat{M}_z[i]\right)^2}, \tag{2.21}$$

where $M_z[i]$ is the i^{th} sample of the measured self-aligning torque and $\hat{M}_z[i]$ is i^{th} simulated value with N samples available. The improved model is compared with the original brush model in Table 2.3, and also in Figure 2.10, with boxplots.

Plate	Original RMSE	Improved RMSE
Metal plate	1.7499	1.5049
Red emery paper	2.8574	0.9744

Table 2.3: Average RMSE of the open-loop simulation using the original brush tire model and the improved friction modeling.

It is visible, that the improved model results in more accurate representation of the tire-ground relation in both cases. Although the improvement for the metal plate is less than the improvement for the emery paper, it is logical, since the original model proved to be already quite accurate in that case. Furthermore the identified dynamic coefficient of friction is closer to the static coefficient in case of the metal plate, therefore no drastic change is expected.

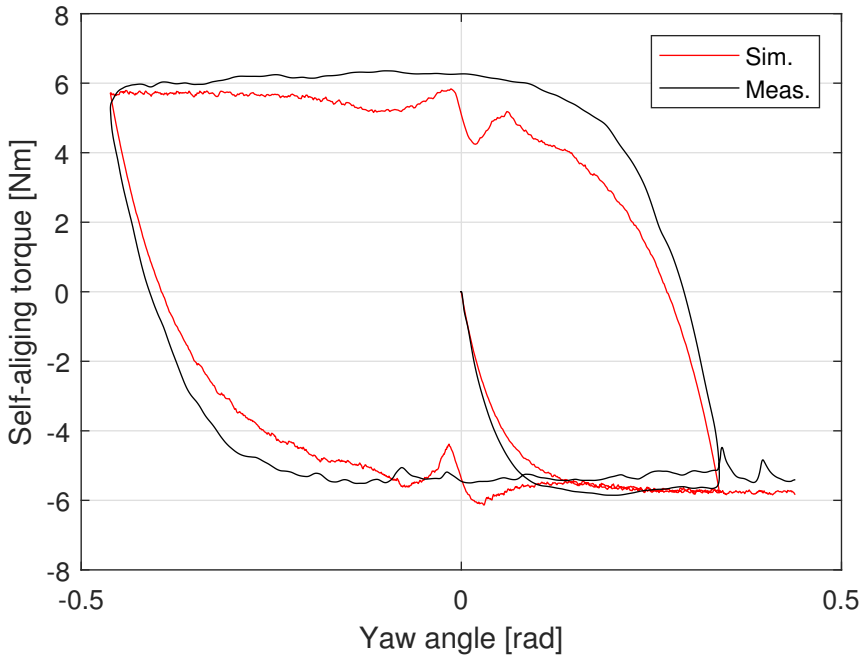


Figure 2.9: The measured and simulated self-aligning torque is compared in the $\psi - M_z$ plane in case of the imporved brush tire model.

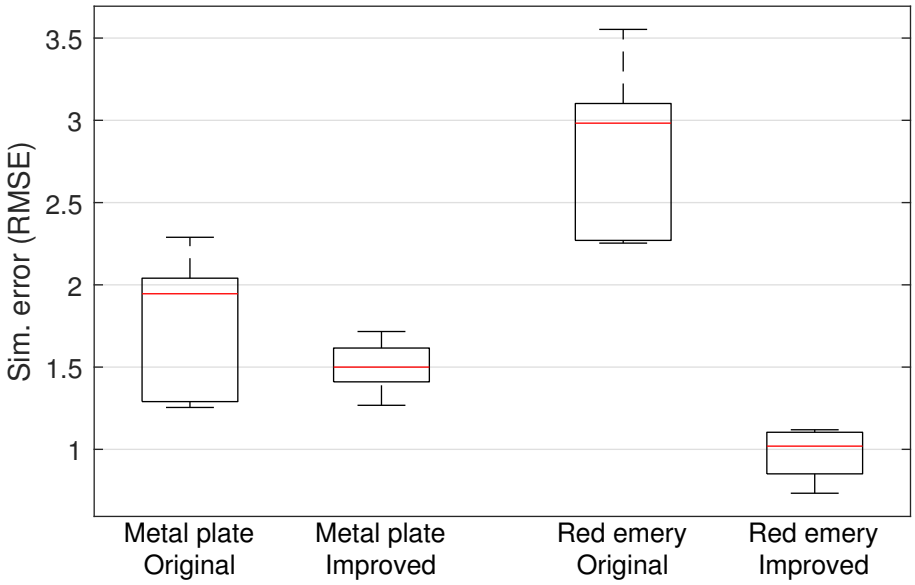


Figure 2.10: Boxplots representing the RMSE error of the self-aligning torque. The median values are illustrated with red lines.

3 Estimating the coefficient of friction

As previously mentioned, accurate information about the tire-ground relation would enhance the performance of the self-balancing control applied for an electric scooter. Hence, the development of an accurate friction estimation method is an important topic. In our previous work [3], an analytic estimation method has been proposed based on the brush tire model. In this section, we summarize the previous estimation algorithm and discuss its problems, then two possible solutions are presented. Furthermore, the proposed methods are evaluated on measurement data, and conclusions are drawn.

3.1 Analytic estimation of the coefficient of friction

The principle of the estimation method is based on (2.14). Assuming $\dot{\psi}(t)$, $\ddot{\psi}(t)$, and $M(t)$ can be measured and J_A, k, b_t, a, e constants are known, the only unknown in the equation of motion is the deformation function of the contact line. It was previously discussed that the deflection of the elastic elements is dependent on the coefficient of friction between the ground and the tire. Assuming $q(x,t)$ can be expressed as a function of the friction coefficient and known or measured physical properties, μ could be calculated theoretically.

As it turns out the deformation function can be expressed using analytic functions according to the in-plane version of the brush tire model. For example, when three deformation zones appear in the contact patch (illustrated in Figure 3.1), $q(x,t)$ can be expressed as

$$q(x) = \begin{cases} \operatorname{sgn}(\psi)\,\dfrac{3\,F_\mathrm{N}\,\mu}{4\,a^3\,k}\cdot(a^2-x^2)\,, & \text{if } x\in[-a,\,x_\mathrm{R}),\\ -\tan\psi\cdot(x-e)\,, & \text{if } x\in[x_\mathrm{R},\,x_\mathrm{L}),\\ -\operatorname{sgn}(\psi)\,\dfrac{3\,F_\mathrm{N}\,\mu}{4\,a^3\,k}\cdot(a^2-x^2)\,, & \text{if } x\in[x_\mathrm{L},\,a], \end{cases} \tag{3.1}$$

where sgn notes the signum function.

Of course, more than three deflection zones can occur in the contact line of an elastic tire. For example, when the direction of rotation is changed, the thread elements, that are in the sliding region can revolve back into the sticking region. As a result, in the deformation function, parabolic sections appear, which are revolved around point A. This phenomena makes the analytic expression of the deformation harder, but not impossible. In [3] the analytic deformation function has been derived for more complex cases, but the estimation method is similar for all cases.

After deriving the analytical deformation function, all terms are known in the equation of motion. The expressions for the deformation contain known or observable physical quantities, only the coefficient of friction remains unknown. The definite integral in the equation of motion can be evaluated analytically because the deformation function is known if the yaw angle is measured and the history of measurements is stored (the "memory effect" of an elastic tire is considered), an algebraic equation can be formed for

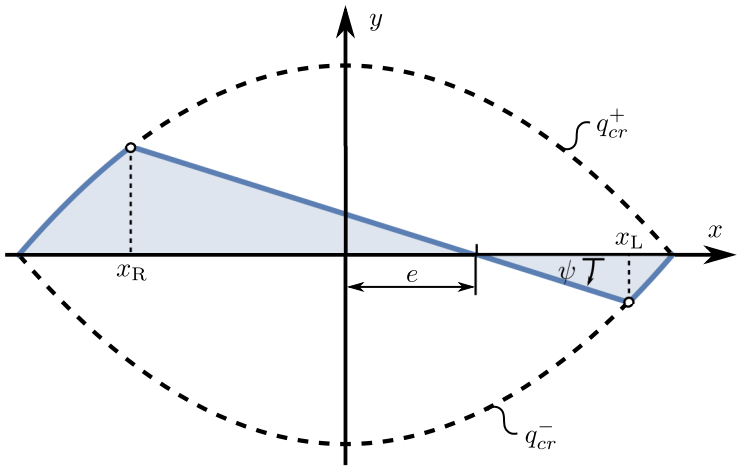


Figure 3.1: In-plane brush model representation of the tire elements deformation. Three distinct zones can be visibly identified, therefore three different equations are needed to describe the deformation function.

the coefficient of friction, as

$$f(\mu) := -J_A\ddot{\psi}(t) - b_{\mathrm{t}}\dot{\psi}(t) + M(t) - \underbrace{\int_{-a}^{a} kq(x,t)\,(x-e)\,\mathrm{d}x}_{=M_z} = 0, \tag{3.2}$$

where $M(t)$ is the steering torque and $f(\mu)$ is the algebraic expression, that needs to be solved to acquire the value of the coefficient of friction. Although this function cannot be solved analytically, it is possible to solve it with numerical schemes. The chosen numerical solver was the built-in `fsolve` function of Matlab [15], using a trust region algorithm.

The high-level steps of the estimation algorithm based on the analytic brush tire model are shown in Algorithm 1. This method proved to be accurate using simulated data and also provided great results for measured data as well. The main advantage of the method is that the current estimation does not depend on the accuracy of the past estimated values. The most important disadvantage of the algorithm is that for measured data, the estimated coefficient of friction showed huge oscillations at certain points caused by numerical instability of the solver. These oscillations affect the average accuracy of the method, shown in Figure 3.2.

Algorithm 1 Analytic friction estimation

Require: J_A, b_{t}, k, e constants to be known

1: **loop**
2: get the measured ψ, $\dot{\psi}$, $\ddot{\psi}$ and M
3: calculate M_z from the equation of motion
4: express the analytic deformation function
5: solve $f(\mu)$ numerically
6: **end loop**

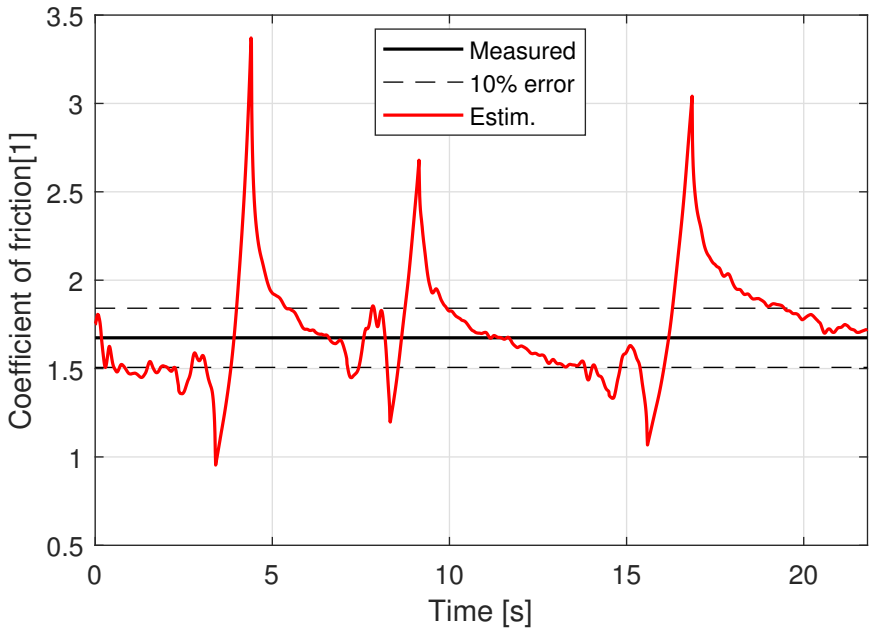


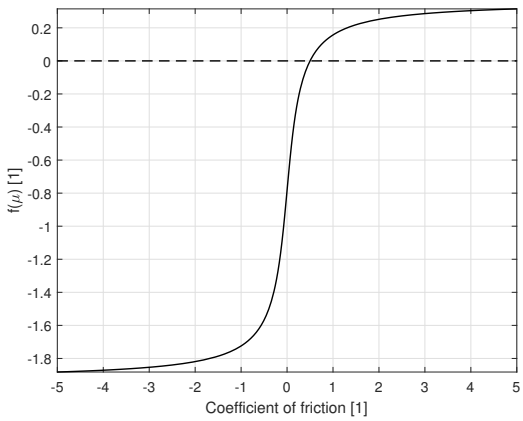
Figure 3.2: Analytic estimation results for the coefficient of friction based on previous results. Big oscillations caused by numerical instability are visible, but apart from these regions, the estimation is quite accurate.

3.2 Enhancing the accuracy of the analytic estimation approach

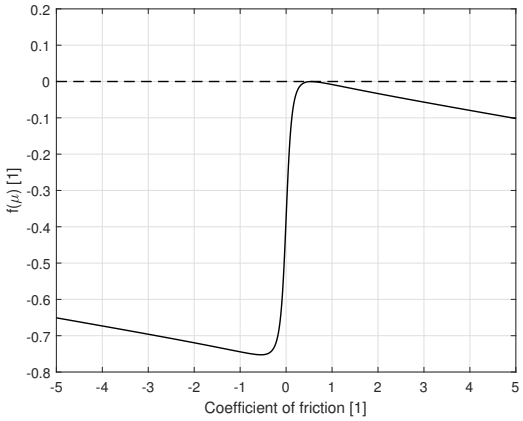
The mentioned results suggest that some numerical inaccuracies cause the problems with the previously proposed estimation method. In Figure 3.3, $f(\mu)$ is illustrated with the sought value of $f(\mu) = 0$. On the left, a "normal" case is visible, where the numerical solver had no problem finding the appropriate solution. On the right side of the figure, a "problematic" case is visible. The function never reaches the value of zero, because of the numerical inaccuracies, therefore the solver approximates the solution with the point of the curve, which is nearest to the x-axis. This approximation results in the previously discussed oscillations in the estimation.

Another possible explanation for the numerical problems is that the analytic deformation function for 5 or more fragmentation zones (illustrated in Figure 3.4) results in an equation for the coefficient of friction, which does not have a unique solution. At small steering torque values, where the self-aligning torque is around zero, very small and very large friction values can be both expected because the estimation process depends on the self-aligning torque. With zero (or near zero) self-aligning torque, the estimation algorithm can provide false results, which makes the average accuracy of the estimation worse.

The main application of the estimation method is to provide important information to improve the self-balancing control of an electric scooter. While traditional friction estimation methods aim to provide information at every time step, during the self-balancing task with no longitudinal motion, the frictional properties do not change that abruptly.



(a) Solving $f(\mu) = 0$ numerically, "normal" case.



(b) Solving $f(\mu) = 0$ numerically, "problematic" case.

Figure 3.3: Illustrating $f(\mu)$ at two different time instances. The dashed line represents the sought $f(\mu) = 0$ value.

Hence, it is sufficient to provide the estimated coefficient of friction only when we know it is accurate. Therefore we aim to develop a method, that can predict the possible error of the analytic estimation, before calculating the estimated value, this way the time-cost of the algorithm can be also improved. The high-level steps of the idea are summarized in Algorithm 2.

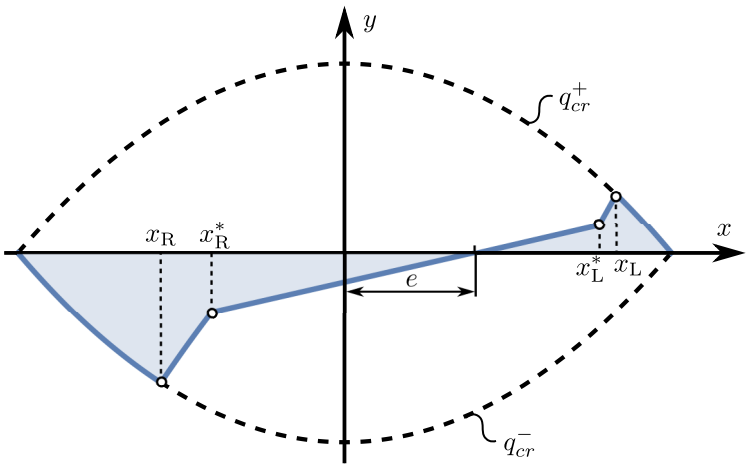


Figure 3.4: In plane model of the contact patch showing five deformation zones. The inner parabolic sections represent the memory effect of the tire.

The main causes of the estimation inaccuracies are the suspected numerical and theoretical errors of the algorithm. The exact reasons are not known, therefore an analytical condition can not be formulated for the estimation. Hence we aim to solve the problem with the use of machine learning. Nowadays, machine learning algorithms and artificial neural networks (ANNs, or just shortly NNs) are well-known terms for the majority of academic society. The idea of creating computational systems based on neurons in the human brain was first introduced about 60 years ago, but in the last 35-40 years, the

usage of previously mentioned algorithms spread worldwide [16]. The most important property of ANNs is the capability to learn complicated connections between patterns from (measured) data. The main application fields are classification or regression tasks, moreover, predictions of time-series events are even possible [17].

Algorithm 2 Condition-based analytic friction estimation

Require: J_A, b_t, k, e constants to be known

1: **loop**
2: get the measured ψ, $\dot{\psi}$, $\ddot{\psi}$ and M
3: calculate M_z from the equation of motion
4: **if** the result is expected to be accurate **then**
5: express the analytic deformation function
6: solve $f(\mu)$ numerically
7: **end if**
8: **end loop**

The basic building blocks of neural networks are so-called neurons, which form layers within the network. The layers can be classified into three main classes, based on their positions in the network. The first layer is called the input layer, the last is the output layer, and all the other ones are called hidden layers. The connection between distinct layers is calculated with the help of weight vectors and activation functions. Let us consider an ANN with L layers, then the output of an arbitrary k^{th} layer ($1 < k < L$) can be expressed as

$$\mathbf{y}_\mathrm{k} = \mathbf{h}_{\mathrm{act,k}} \left(\mathbf{W}_\mathrm{k} \cdot \mathbf{y}_{\mathrm{k-1}} + \mathbf{b}_\mathrm{k} \right), \tag{3.3}$$

where $\mathbf{h}_{\mathrm{act,k}}$ is the activation function of the given layer, $\mathbf{W}_\mathrm{k}$ is the weight matrix of the layer, $\mathbf{y}_{\mathrm{k-1}}$ is the output of the $(k-1)^{\mathrm{th}}$ layer and $\mathbf{b}_\mathrm{k}$ is the bias vector of the layer. Assuming that the k^{th} layer has N_k neurons, similarly the $(k-1)^{\mathrm{th}}$ layer consists of $N_{\mathrm{k-1}}$ neurons, the output vectors $\mathbf{y}_\mathrm{k} \in \mathbb{R}^{N_\mathrm{k}}$ ans $\mathbf{y}_{\mathrm{k-1}} \in \mathbb{R}^{N_{\mathrm{k-1}}}$. Also the weight matrix and the bias vector $\mathbf{W}_\mathrm{k} \in \mathbb{R}^{N_\mathrm{k} \times N_{\mathrm{k-1}}}$, $\mathbf{b}_k \in \mathbb{R}^{N_\mathrm{k}}$. Thus, the output of the entire neural network can be expressed as a nested function

$$\mathbf{y} = \mathbf{y}_\mathrm{L} = \mathbf{h}_{\mathrm{act,L}} \circ \cdots \circ \mathbf{h}_{\mathrm{act,2}} \circ \mathbf{h}_{\mathrm{act,1}} \left(\mathbf{x} \right), \tag{3.4}$$

where $\mathbf{x} \in \mathbb{R}^{N_0}$ is the input vector of the network [18]. In the learning phase, the aim is to find the optimal elements of the weight matrices and the bias vectors to minimize the difference between the calculated output of the system and the expected, pre-defined output. For activation functions, usually nonlinear functions are chosen. Typical examples are rectified linear, sigmoid (in other words logistic), or hyperbolic tangent (tanh) functions.

The neural network was programmed in Python, using the built-in functions of powerful machine learning libraries, such as Tensorflow [19] and Keras [20]. The goal of the network is to approximate the relative error of the friction estimation method based on the yaw

angle ψ, yaw rate $\dot{\psi}$, yaw acceleration $\ddot{\psi}$ and the self-aligning torque M_z. From the previously discussed 13 measurements (7 with the emery paper and 6 with the metal plate), 11 of them were used for training and 1 measurement for each plate was used for testing. The hyper-parameters of the network are summarized in Table 3.1

Activation fun.	Optimizer	Batch size	Regularization const.
tanh	Adam*	512	0.001

Table 3.1: Hyper-parameters of the ANN, that were not modified. *The Adam optimizer was used with a fixed learning rate of 0.001.

The network was trained for 1000...5000 epochs, depending on how the training losses occurred. 20% of the training data was used for validation, and an L1 regularization term was also applied to prevent overfitting. The number of hidden layers and the number of neurons per layer were varied to find the best possible structure. The training losses are illustrated for an arbitrarily chosen structure in Figure 3.5. After training the model structure was restored to match the model with the best validation error.

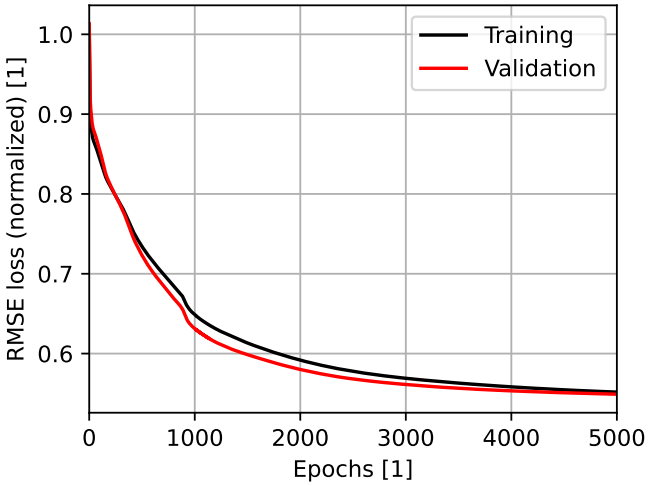


Figure 3.5: Training and validation losses defined by the RMSE metric for one of the ANN structures.

The number of hidden layers was varied from 1 to 3, and the number of neurons between 16 to 128. The RMSE between the predicted relative error and the measured error was monitored and this way, the optimal architecture of the ANN was found to be 1 hidden layer with 16 nodes. The exact test RMSE values are summarized in Figure 3.6.

Thus, the selected ANN architecture is summarized as:

- input dense layer with 16 neurons, tanh activation function, and an input size of $\mathbf{x} \in \mathbb{R}^4$,
- one hidden, dense layer with 16 neurons, and tanh activation function,
- output dense layer with 1 neuron (since the output $\mathbf{y} \in \mathbb{R}$), linear bypass.

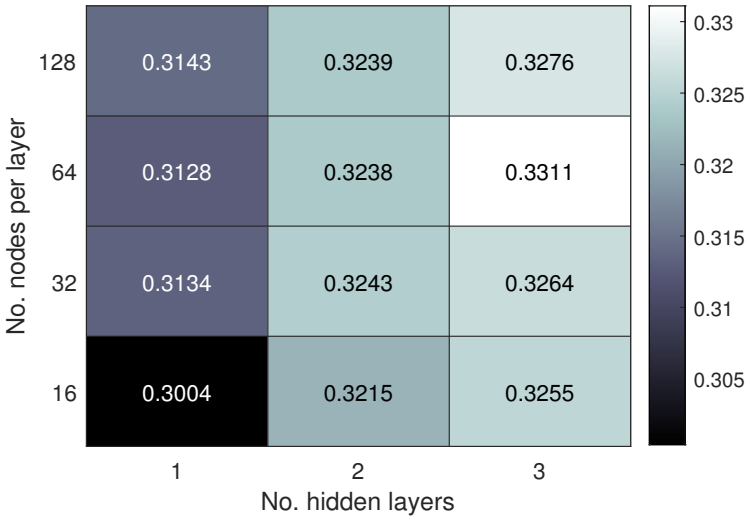


Figure 3.6: Heatmap showing the test errors (with the RMSE metric) for different ANN architectures.

The ANN structure is evaluated on the test data set, which is shown in Figure 3.7. The condition for the estimation algorithm is that the predicted relative error should be below 30%. The results of the method are shown in Figure 3.8 for one of the test cases.

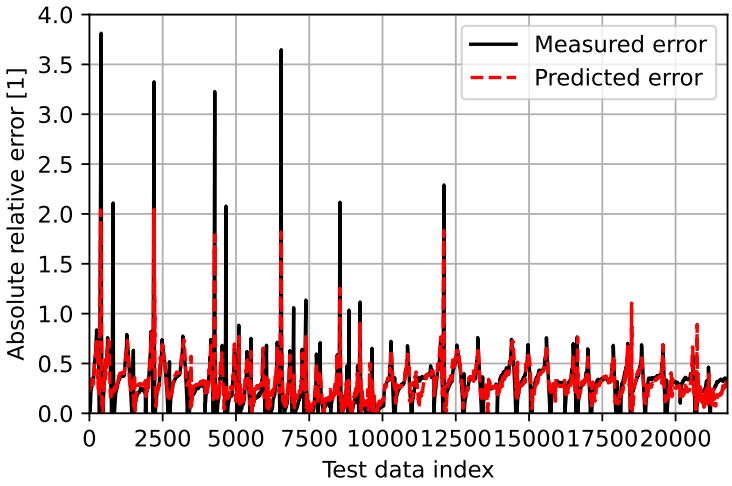


Figure 3.7: The best ANN evaluated on the test data set. The error between the analytic estimation and the measured coefficient of friction is shown in black, and the prediction for the error (by the ANN) is shown with dashed red line.

3.3 Estimating the coefficient of friction considering different static and dynamic friction properties

We showed in Section 2.3, that considering different static and dynamic friction properties immensely affects the accuracy of the brush tire model. Thus it is a reasonable expectation that an improved estimation method, which considers different static and dynamic

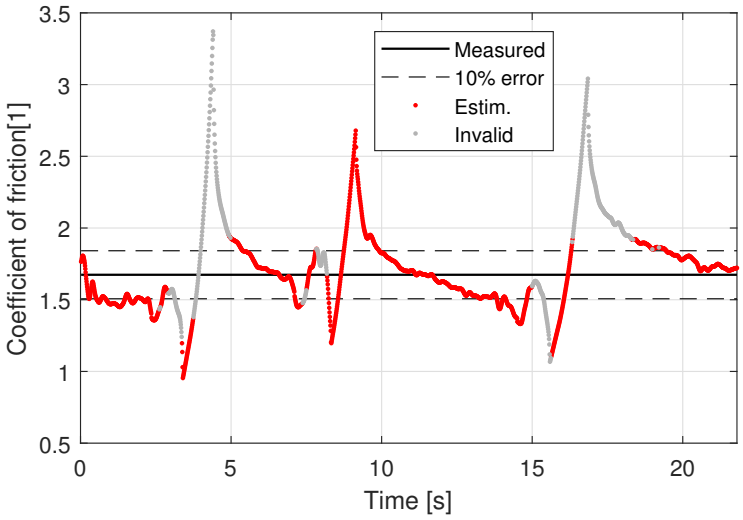


Figure 3.8: Estimation results with predicted relative error. The results shown with blue dots were predicted to have high error, thus would be not calculated.

coefficients of friction, would be more accurate than the previously proposed algorithm.

In theory, Algorithm 1 could be modified to use the improved friction characteristics of the brush tire model, but the analytic approximation of the deformation function is a much harder task. A simulation with arbitrary parameters is run to illustrate this problem. In Figure 3.9, the steering torque and the yaw angle are shown with the four time instants noted when the deformation function is illustrated in Figure 3.10. It is visible from Figure 3.10, that the analytic deformation function-based method can not be utilized, therefore another method needs to be developed.

The main idea of this new method is that a simulation, similar to the previously mentioned open-loop simulation can be run in real-time with the experiment. At a given discrete time-step k let us suppose that the deformation function is known as q_k, and the value of $\dot{q}$ is also known as $\dot{q}_k$. The measured signals at this given time-step are $\psi[k]$ yaw angle, $\dot{\psi}[k]$ yaw rate, and $\ddot{\psi}[k]$ yaw acceleration. Also, the input of the system, the steering torque $M[k]$ is known. The self-aligning torque can be measured or calculated according to (2.13) as

$$M_z[k] = M[k] - b_\mathrm{t}\dot{\psi}[k] - J_A\ddot{\psi}[k]. \tag{3.5}$$

Using Euler's formula (which is a simple and rather inaccurate solution, but quick for running the simulation real-time) to solve the next step of the differential equation:

$$q[k+1] = q[k] + \dot{q}[k] \cdot \Delta t. \tag{3.6}$$

But using (2.20), q_{k+1} can be reformulated as

$$q[k+1] = \begin{cases} q[k] + \dot{q}[k] \cdot \Delta t, & \text{if } |q[k] + \dot{q}[k] \cdot \Delta t| < q_\mathrm{cr}^{\pm}[k+1], \\ q_\mathrm{slip}^{\pm}[k+1], & \text{otherwise,} \end{cases} \tag{3.7}$$

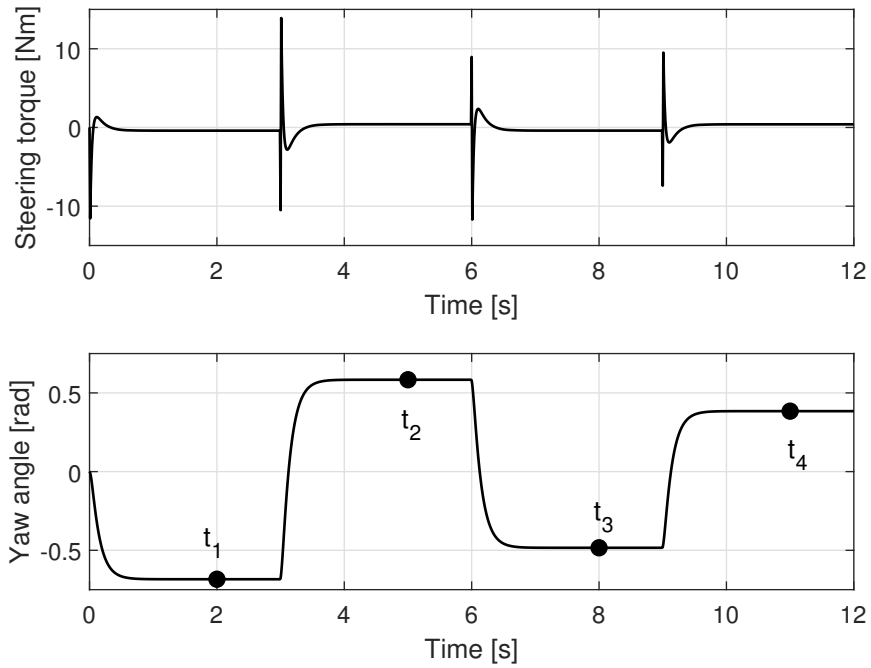


Figure 3.9: Results of the simulation regarding the steering torque and the yaw angle. The four dots represent the time instants when the deformation function will be illustrated.

where

$$q_{\mathrm{cr}}^{\pm}[k+1] = \pm \frac{3F_{\mathrm{N}}\mu_{\mathrm{s}}[k+1]}{4a^3k}\left(a^2 - x^2\right), \tag{3.8}$$

and similarly,

$$q_{\mathrm{slip}}^{\pm}[k+1] = \pm \frac{3F_{\mathrm{N}}\mu_{\mathrm{d}}[k+1]}{4a^3k}\left(a^2 - x^2\right). \tag{3.9}$$

In (3.8) and (3.9) $\mu_{\mathrm{s}}[k+1]$ and $\mu_{\mathrm{d}}[k+1]$ are unknown and we seek to estimate these values based on the self-aligning torque. Let us suppose that we choose $\mu_{\mathrm{s}}[k+1] = \hat{\mu}_{\mathrm{s}}$ and $\mu_{\mathrm{d}}[k+1] = \hat{\mu}_{\mathrm{d}}$. With these values, the deformation function $q[k+1]$ can be calculated based on (3.7), then the estimated self-aligning torque can be expressed as

$$\hat{M}_z[k+1] = \int_{-a}^{a} k \cdot q[k+1] \cdot (x - e)\ \mathrm{d}x. \tag{3.10}$$

Of course, it is more convenient to discretize the contact patch along the x axis for the real-time simulation of the system. With a N discrete points in the $[-a; a]$ interval, with a step size of Δx, the self-aligning torque can be approximated as

$$\hat{M}_z[k+1] = \sum_{i=2}^{N} k \cdot q_i[k+1] \cdot (x_i - e) \cdot \Delta x, \tag{3.11}$$

where x_i is the i^{th} element of the discretized contact patch and $q_i[k+1]$ is the deformation value at x_i at time instant $k+1$. With the calculated and estimated self-aligning torque,

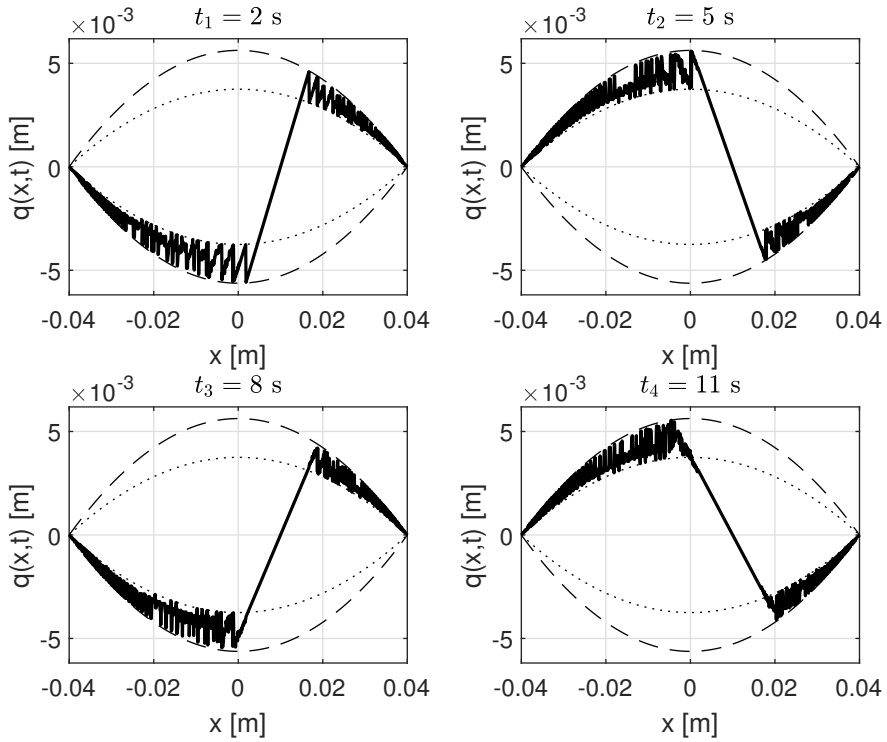


Figure 3.10: The deformation function of the center-line of the tire, illustrated at the given time instants. The dashed line represents the static friction limit and the dotted line represents the dynamic friction limit.

an optimization problem can be set as

$$\begin{aligned} &\underset{\hat{\mu}_\mathrm{s},\,\hat{\mu}_\mathrm{d}}{\text{minimize}} \quad \Big(M_z[k+1] - \hat{M}_z[k+1]\Big)^2 \\ &\text{subject to} \quad 0 \leq \hat{\mu}_\mathrm{d} \leq \hat{\mu}_\mathrm{s}. \end{aligned} \tag{3.12}$$

With this optimization task at every time step, the coefficient of friction can be estimated in real-time. The main advantage of the method is that the friction properties are not the same in the sliding and sticking region compared to the previously shown estimation methods. The high-level steps of the proposed estimation algorithm are summarized in Algorithm 3. For an arbitrarily chosen experiment, the estimation results are shown in Figure 3.11.

Although the newly proposed estimation method is seemingly more accurate than the previous ones, it also has some disadvantages that must be addressed. As the contact patch is discretized along the x axis, the accuracy of the estimation depends on the resolution of the discretization. This only causes problems for the real-time calculation, but as turns out the resolution of the contact patch does not affect the computational time as much, compared to the time needed to solve the optimization problem. A larger problem with this method is that the estimation at a given time step depends on previous results, thus a possible error will affect the accuracy of the estimation in the future.

Algorithm 3 Numerical optimisation-based estimation method

Require: J_A, b_t, k, e constants to be known

initialize $q[0]$ and $\dot{q}[0]$
$k = 0$
loop
    get the measured $\psi[k+1]$, $\dot{\psi}[k+1]$, $\ddot{\psi}[k+1]$ and $M[k+1]$
    calculate $M_z[k+1]$ from the equation of motion
    minimize $\left(M_z[k+1] - \hat{M}_z[k+1]\right)^2$ $\hat{\mu}_\mathrm{s}, \hat{\mu}_\mathrm{d}$
    $k = k + 1$
end loop

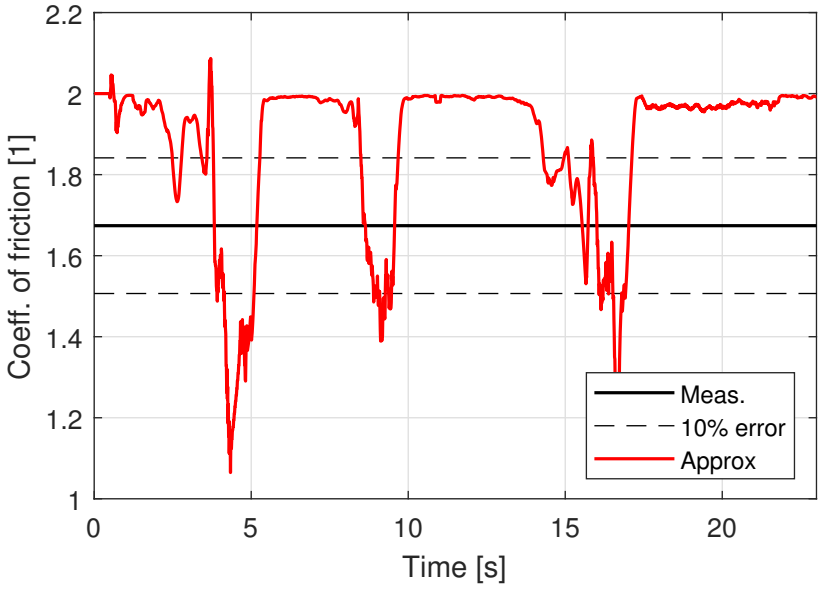


Figure 3.11: Estimation of the coefficient of friction with the proposed numerical optimisation-based method.

3.4 Comparing the different estimation methods

The accuracy of each estimation method can be evaluated on the previously discussed measurements. Two of these measured signals were used for the identification of the dynamic coefficient of friction, thus the proposed numerical method was not tested on those data sets. Similarly, the ANN-based conditions for predicting the error of the analytic estimation were not evaluated on the training data set. Therefore, only two measured data are available to illustrate the three different methods, those are visible in Figure 3.12 and Figure 3.13.

It is visible, that the proposed condition for the analytic estimation method can enhance the accuracy of the algorithm, but the numerical estimation still performed better. The main reason for this is that the originally proposed algorithm does not consider the dynamic coefficient of friction, which turned out quite an important factor for analyzing

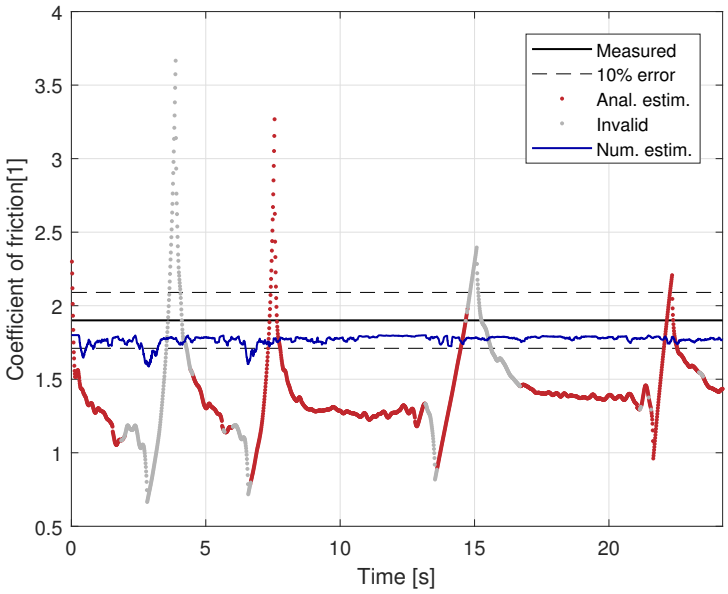


Figure 3.12: Results of different estimation methods for one of the measurements of the emery paper.

the dynamics of the contact patch. The accuracy of the three methods is summarized in Figure 3.14.

It is hard to compare the ANN-based conditional estimation with the proposed numerical method because of the small sample size and the different results for the two surfaces, but it is clear that both methods are more accurate than the original estimation. The most important aspect of the ANN-based algorithm is its simplicity and the fact, that the error at a given time step does not depend on previous time steps. From Figure 3.14, it is visible that the numerical estimation provides more robust results, based on the small deviation of estimation error. The main disadvantage of the method is the possibility of large cumulative error because the method depends on previously calculated results.

Another important aspect would be the time cost of each method. Unfortunately, these values were not monitored during this work, but as a qualitative result, we can state that the numerical method took rather longer time to compute. This aspect will be examined in a future work to provide exact values for the comparison of methods.

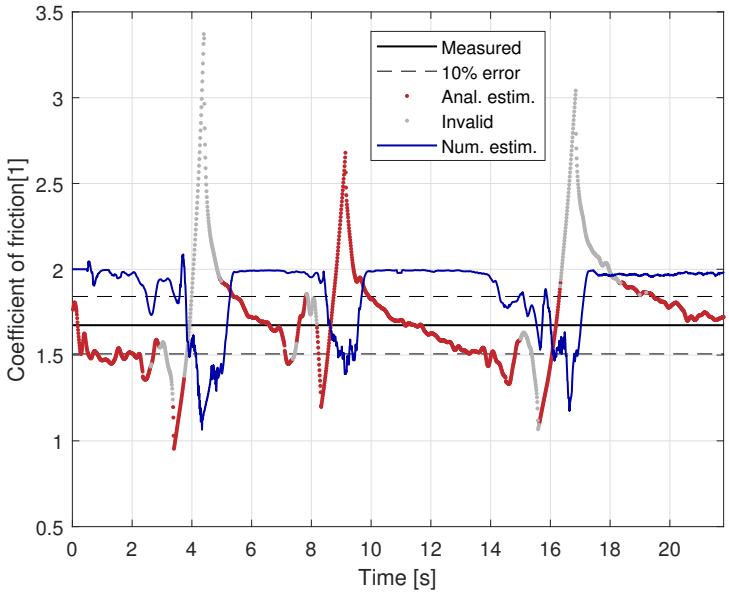


Figure 3.13: Results of different estimation methods for one of the measurements of the metal plate.

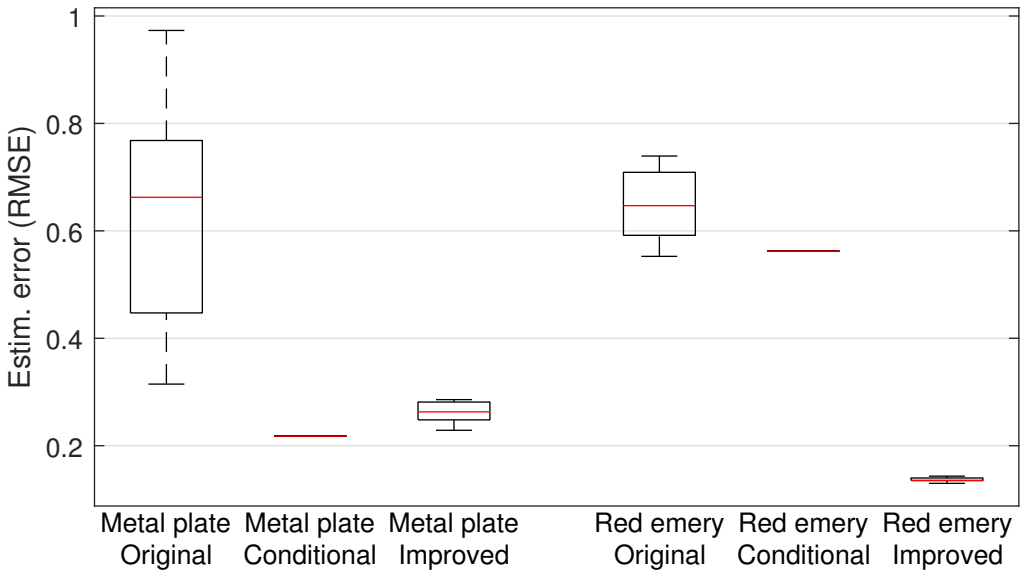


Figure 3.14: Estimation errors of the different methods. The ANN-based conditional estimation is only evaluated on the test data sets, therefore only one error value is available for each surface.

4 Development of an experimental equipment for self-balancing e-scooters

In this section, the development process of the special equipment is detailed, which can be mounted on a Xiaomi Mi Electric Scooter 3 (from now on E-Scooter). The basic idea is the following: there is a brush-less direct current (BLDC) motor fixed to the deck or the folding hinge, the important thing is that it should not rotate with the steering shaft. On the motor shaft, a timing belt pulley is placed, similarly another pulley is fixed to the steering column (stem) of the E-Scooter. With a timing belt connecting the two pulleys non-slip connection can be reached, which is necessary for accurate position control. The schematic view of the system can be seen in Figure 4.1.

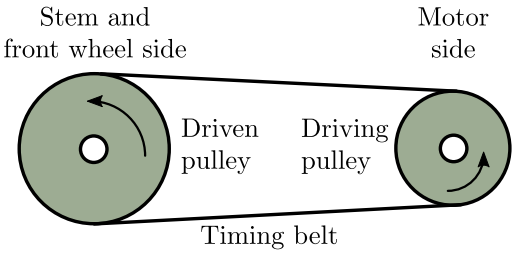


Figure 4.1: Schematic view of the timing belt steering system.

4.1 System requirements and design

The system is designed around the estimated maximal steering torque ($T_{\max}$) that is necessary for self-balancing. Based on simulations at the early stages of the development, the maximal steering torque should be around 5 Nm. But it is worth noting that the model used for the simulation, does not contain friction and damping characteristics, furthermore it has not been validated yet, therefore this $T_{\max} = 5$ Nm should be corrected with a safety factor.

The first step was to select an appropriate motor for this task, which fulfills the previously defined requirement for the max torque. The ratio of the delivered torque and the motor size is typically high for BLDC motors, hence an ODrive M8325S - 100 KV was selected for this application. The most important properties are summarized in Table 4.1. The max torque of the motor is 4.96 Nm, but it will be multiplied with the gear ratio of the belt drive to get the max steering torque.

Apart from selecting an ideal motor, a control circuit is also necessary for driving the motor. A lot of options can be found on the internet, but for simplicity the control board was ordered form the same company as the motor. Another huge advantage of the ODrive Pro control circuit (which was selected) is that it is not only driving the motor by a pre-configured control task (for example closed loop position control), but it also serves as a micro-controller, therefore many analytic data can be requested from it. The software

Parameter	Notation	Value	Unit
Speed constant	$k\phi$	100	rpm/V
Max current	I_{max}	60	A
Max voltage	U_{max}	60	V
Phase resistance	R_{ph}	44	$m\Omega$
Mass	m_{mot}	841	g
Max torque	$T_{\text{mot,max}}$	4.96	Nm
No-load speed	Ω_0	402.12	rad/s

Table 4.1: Most important parameters of the ODrive M8325S - 100 KV motor. Note that the current and torque ratings have been determined with extremely good air cooling.

support is outstanding with available Python and Arduino libraries, as well as a ROS2 CAN package. Also a web GUI is provided by the company, which simplifies the initial configuration. The selected motor and the control circuit can be seen in Figure 4.2.

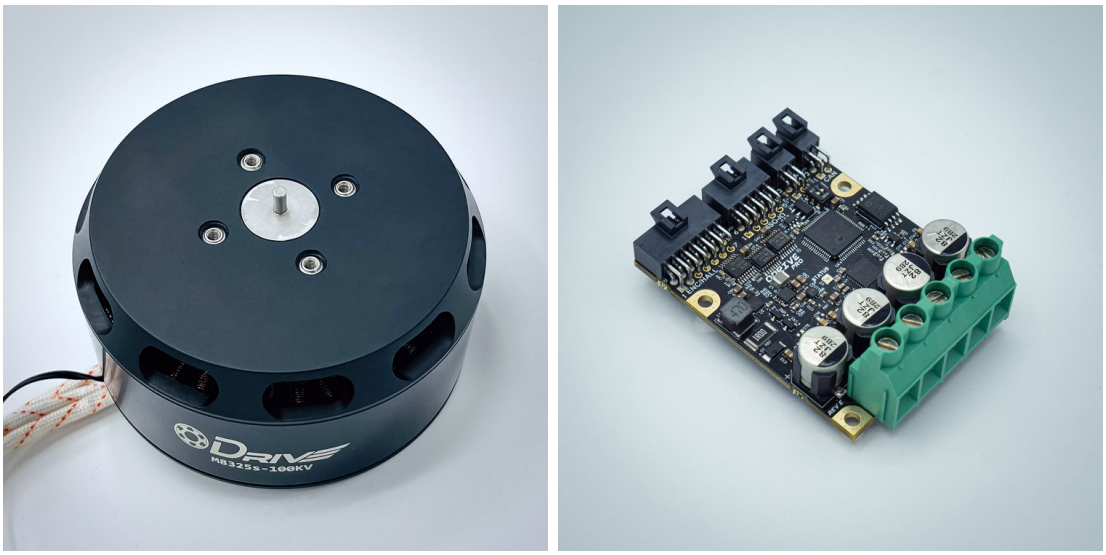


Figure 4.2: ODrive M8325S - 100 KV motor[1] (left) and ODrive Pro board[2] (right).

After selecting the motor and the control board, now the pulleys and the timing belt could be selected. The driving pulley should be fit on top of the motor, which results in a restriction for its size. Also the driven pulley needs to be mounted on the stem, hence it is an other requirement that should be considered. The two pulleys and the timing-belt was ordered from Power Belt Hajtástechnikai és kereskedelmi Kft [21]. After these considerations the smaller pulley's diameter (D_1) was selected to be 50.93 mm, the larger one's is $D_2 = 114.6$ mm. Hence, the gear ratio can be calculated as

$$i = \frac{D_2}{D_1} \approx 2.25. \tag{4.1}$$

[1]Source: `https://odriverobotics.com/shop/m8325s` (accessed: 2023-09-16)
[2]Source: `https://odriverobotics.com/shop/odrive-pro` (accessed: 2023-09-16)

The theoretical max steering torque can be calculated as

$$T_{\max} = i \cdot T_{\text{mot,max}} = 11.16 \text{ Nm}, \tag{4.2}$$

which does not take into consideration any losses caused by friction, damping or even the heating of the motor, but with a safety factor of $n = 2.23$, it can be said with great confidence, that the system will able to provide enough torque for self-balancing purposes. One can argue that the system is over-secured, but the reason for the theoretical max steering torque to be that high, is that the possibility to use this experimental equipment for other projects (for example autonomous driving e-scooters) was also considered.

4.2 Designing the motor mounting tool

As previously mentioned, the BLDC motor should be mounted onto a part, which does not rotate with the stem. A general idea would be to fix it on the deck, but in the case of the E-Scooter, another solution seemed feasible. There is a light-reflecting plastic attached to the front of the vehicle, with two screws. This is visible in Figure 4.3. By disassembling this part, a motor holding equipment can be fixed by using these screws. This is an ideal position for the motor, because it does not revolve with the steering column and the driven pulley can be fixed to the folding hinge.

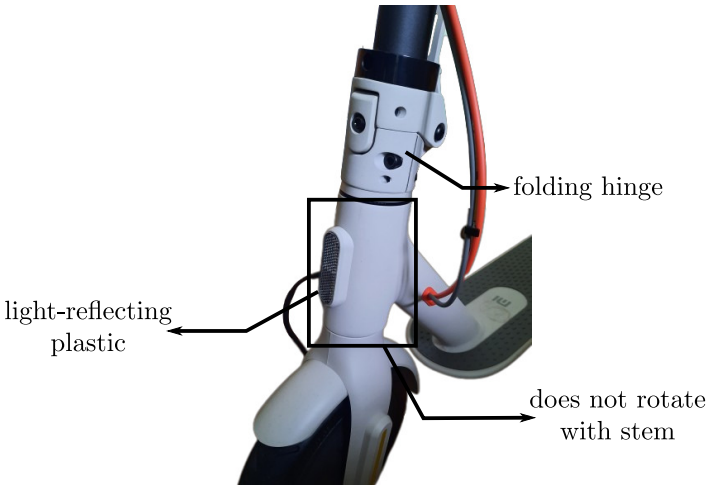


Figure 4.3: The front of the E-Scooter with the light-reflecting plastic attached with two screws. It is visible that the part which is illustrated in a rectangle, is attached to the deck and, hence does not rotate with the stem. The folding hinge is also noted.

The most important design aspect of the mounting equipment is that the motor and the control circuit need to fit inside, the driving and driven pulleys should be on the same height and it should be able to be fixed onto the E-Scooter. Another aspect was to ensure that the designed equipment can endure the mechanical stresses during operation. Based on the prototype aspect of the design, instead of traditional materials 3D printing was used. The size restrictions of the design were checked with a 3D CAD program (can be seen in Figure 4.4), therefore the final product could fit on the E-scooter perfectly.

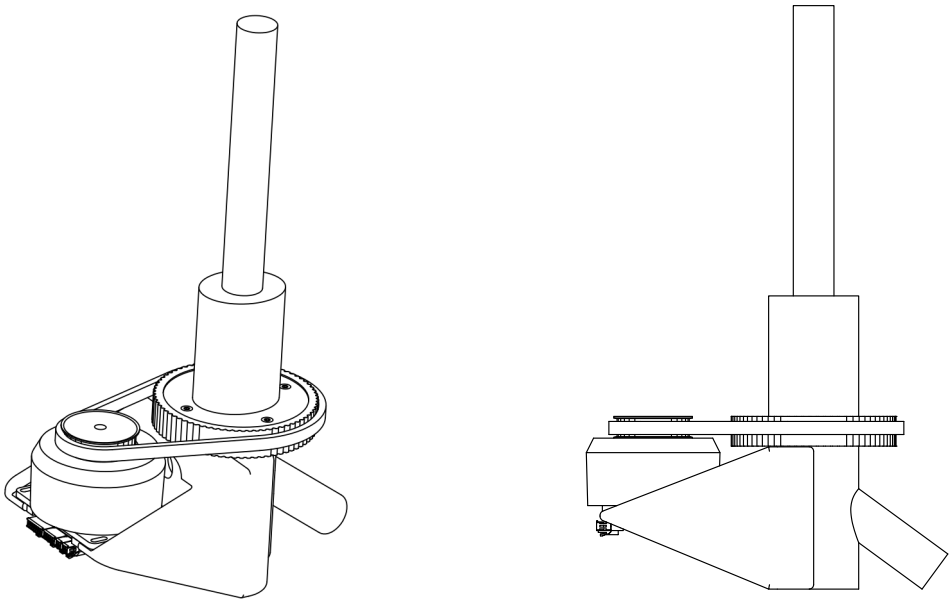


Figure 4.4: CAD drawing of the motor, two pulleys and the timing belt mounted on the E-Scooter. Axonometric view (left) and side view (right).

4.3 Power supply

The above mentioned control circuit provides the three-phase alternating current for the motor, but the power circuit itself needs a DC current to operate. The operating interval of the control ODrive Pro is between $U_{\text{min,DC}} = 15$ V and $U_{\text{max,DC}} = 58$ V, therefore appropriate power supply needs to be selected. The BLDC motor generates significant amounts of reverse current, which can be harmful for the power supply. This can be resolved with a brake-resistor connected to the ODrive Pro, or another solution is to use a battery instead of a regular power supply. Fortunately the battery of the E-Scooter (with small modifications to the plug) can be connected to the control circuit. The most important properties of the battery are summarized in Table 4.2. It is visible, that this battery is suitable for this task.

Parameter	Notation	Value	Unit
Nominal voltage	U_{n}	36	V
Charge voltage limit	$U_{\text{ch,lim}}$	36 ± 0.05	V
Standard charge current	$\boldsymbol{I_{\text{ch}}}$	1.5	A
Quick charge current	$I_{\text{ch,max}}$	2	A
Standard discharge current	$I_{\text{d,n}}$	3.9	A
Max discharge current	$I_{\text{d,max}}$	7.8	A

Table 4.2: Most important parameters of the used battery.

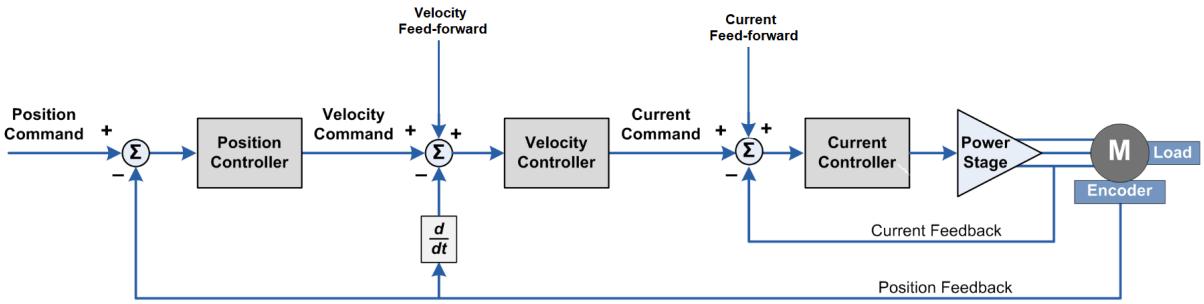


Figure 4.5: Cascade-styled controller of the ODrive Pro[1].

4.4 Motor control program

With the battery and the selected control circuit, the motor control program could be assembled and tested. With the provided ODrive Python package it was convenient to program the desired operating of the motor in Python environment. Furthermore, later on, when the self-balancing task will be tested, the developed algorithm can be easily implemented in Python and it can be used with a Raspberry Pi microcontroller [22], which is Python-based.

4.4.1 PD control

The motor position can be controlled with a simple PID controller. Unfortunately ODrive does not provide an exact documentation for how the control signal is calculated, but the structure of the system is well documented. The cascade-styled controller is illustrated in Figure 4.5. For the E-Scooter, position control will be used, therefore the whole loop is utilized. Unfortunately (with the unknown hardware specification), the control gains cannot be computed analytically, but there is a recommended tuning process provided by the manufacturer. With that process, the control gains have been determined empirically:

$$k_p = 57.122 \text{ (turn/s)/turn}, \tag{4.3}$$

$$k_d = 0.8484 \text{ Nm/(turn/s)}, \tag{4.4}$$

$$k_i = 0 \text{ Nm/((turn/s)} \cdot \text{s)}, \tag{4.5}$$

where k_p is the position gain, k_d is the velocity gain and k_i is the velocity integral term. While the use of the integral term could theoretically provide zero tracking error in steady state, but also causes many unwanted effects. For example the integrator windup phenomena [23], thus the integral gain was set to zero and only a PD controller was used.

With the ODrive software, the position measured by the onboard encoder, the estimated motor torque and the calculated velocity can be all transferred to the master computer

[1]Source: `https://docs.odriverobotics.com/` (accessed: 2023-10-25)

by simple commands. This way the experimental data can be easily analyzed after the measurement.

Another advantage of the ODrive software is that the position commands can not only set for static points, by dynamically, trajectory tracking control can be also achieved. A simple sinusoidal example is illustrated on Figure 4.6, with the baseline control gains and also with the tuned ones.

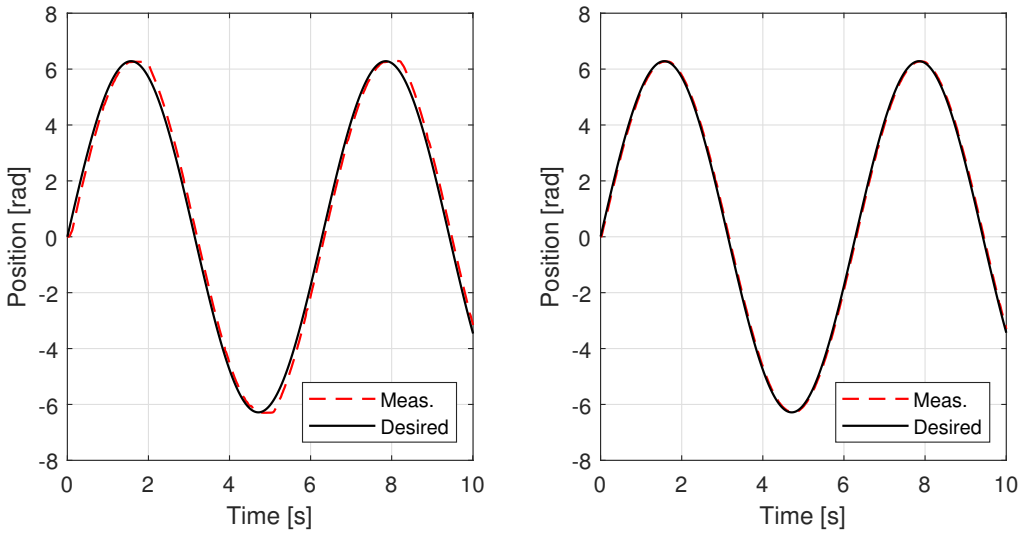


Figure 4.6: Trajectory tracking control of the motor (without any load). With the base control gains (left) and with the tuned gains (right).

4.4.2 Friction measurement of the motor

Other important parameters of the motor are the friction characteristics. While the correctly tuned control algorithm can compensate the frictional effects on the position control, the friction torque is needed for accurate calculations for the equation of motion (EoM). The motor torque is estimated by the ODrive Pro from the measured motor current, this way the torque on the E-Scooter's stem can be calculated as

$$T = i \cdot (T_{\text{mot}} - T_{\text{fric}}) \,, \tag{4.6}$$

where T_{fric} is the friction torque and the losses of the timing belt system are neglected. Based on simulations, for the self-balancing application the operating range of the motor is around $|\omega_{\text{mot}}| = 0 \ldots 10$ rad/s, which can be considered low speed for that specific motor, thus the identification of the friction characteristics is essential.

The friction characteristic is assumed to be a combined static-Coulomb-viscous model, which is typical for BLDC motors. Also, the Stribeck effect can be considered, but it is assumed to be negligible, in other words, the Coulomb friction is assumed to be equal to the static friction term. In this way the friction torque can be expressed at constant

velocities according to the Karnopp-Reynolds [24] model as

$$T_{\text{fric}}(\omega_{\text{mot}}) = \begin{cases} T_{\text{s}} \cdot \operatorname{sign} \omega_{\text{mot}}, & \text{if } |\omega_{\text{mot}}| < \Delta\omega, \\ T_{\text{s}} \cdot \operatorname{sign} \omega_{\text{mot}} + b_{\text{m}}\omega_{\text{mot}}, & \text{if } |\omega_{\text{mot}}| > \Delta\omega, \end{cases} \tag{4.7}$$

where T_{s} is the static friction force and b_{m} is the viscous friction coefficient. Also a small $\Delta\omega$ is introduced, because the friction value would be non-unique around zero velocity.

Based on [25], the static friction torque and the viscous friction coefficient of BLDC motors can be determined with a simple experiment, of continuously increasing the armature current (the motor torque could be approximated with the torque constant), while measuring the velocity. This way, the measured and identified friction characteristics are illustrated in Figure 4.7. The identified friction parameters are summarized in Table 4.3.

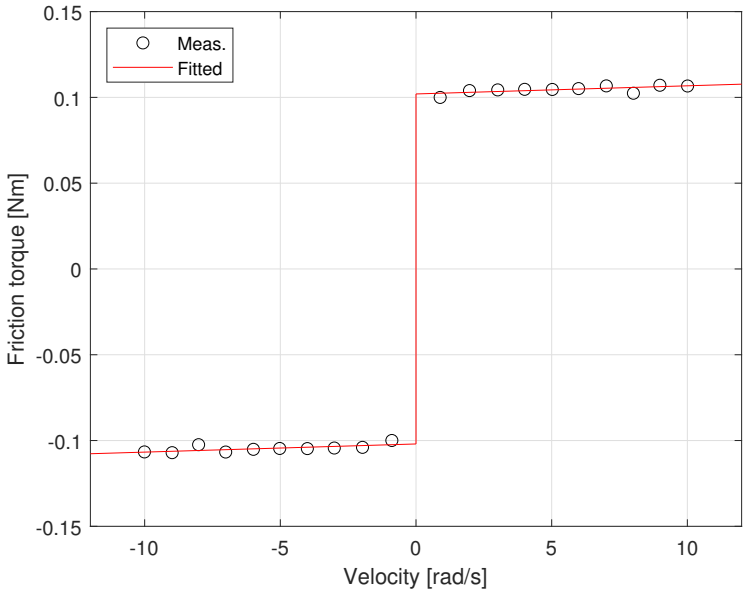


Figure 4.7: Measured and identified motor friction curve.

Parameter	Notation	Value	Unit
Static friction torque	T_{s}	0.102	Nm
Viscous friction coefficient	b_{m}	0.000475	Nm

Table 4.3: Identified motor friction parameters.

4.5 Assembling the experimental instrument

The previously discussed experimental pieces of equipment have been installed on the E-Scooter to construct real-world measurements. The CAD model and the real-world equipment are compared in Figure 4.8. The whole assembly can be seen in Figure 4.9.

Figure 4.8: Comparison of the CAD model (left) and the real-world assembly (right).

To validate the calculated gear ratio and check how synchronous transmission can be achieved with the timing belt, a simple experiment has been conducted. A sinusoidal reference trajectory has been actuated by the motor with an increasing frequency. The stem angle and the motor angle have been both measured with analogous magnetic encoders, thus the two angle signals could be compared. The results are shown in Figure 4.10. The frequency of the input signal has been logarithmically increased from 0.25 Hz to 20 Hz. The frequency dependency of the gear ratio is calculated by taking the Fourier transform of both signals (with the FFT algorithm) and dividing the frequency responses. This way, the transfer function can be determined from measurements. The results are shown in Figure 4.11. It is visible that near zero frequency, the gear ratio is 2, at higher frequencies

Figure 4.9: The whole assembly of the E-Scooter with the designed experimental set-up.

it tends toward 2.1. The differences between the characteristics at high- and low frequencies can be explained by the relative motion of the pulleys at higher frequency motions. The difference between the measured and the nominal gear ratio can be explained with the inaccurate data sheet provided by the manufacturer. And some measurement errors could also influence the results, since inaccurate calibration of the encoders could result in false results.

All-in-all, the experiments proved that the timing-belt transmission provides a synchronous connection between the motor and the stem. The experimentally determined gear ratio is $i \approx 2$.

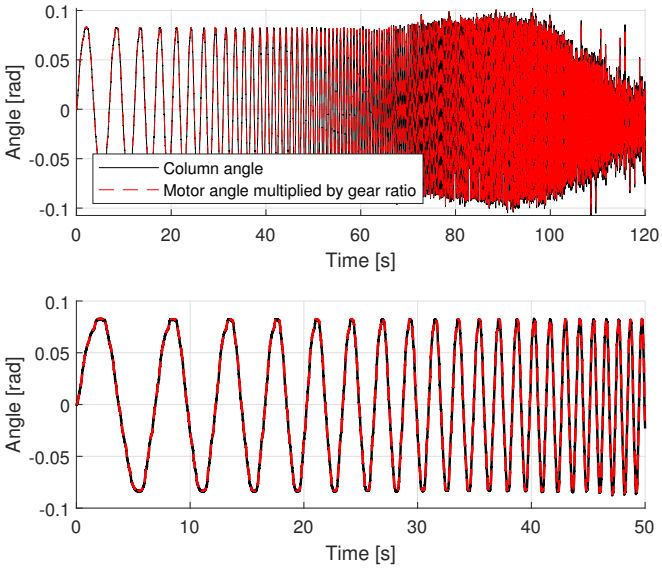


Figure 4.10: The stem angle and the motor angle multiplied by the identified gear ratio as a function of time. On the lower plot, the first 50 seconds of the signals are visible.

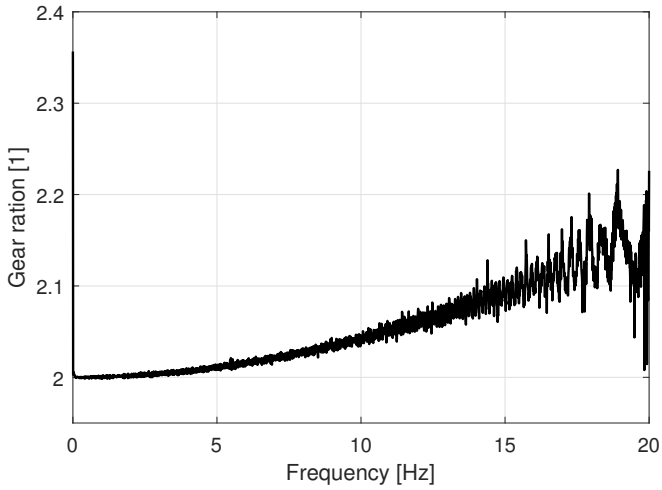


Figure 4.11: The magnitude of the transfer function.

4.6 Estimating the coefficient of friction using the developed experimental tool

With the above discussed experimental equipment, we conducted measurements to test the previously developed estimation methods with a realistic application form. Although the reduced inertia of the stem and the transmission system (the pulleys and the motor) is not known, the experiments have been executed successfully.

The mass moment of inertia of the E-Scooter can be approximated with CAD files and various data-sheets from the internet and the motor inertia can be easily identified by measurements. The inertia of the pulleys could be determined from the CAD models. Despite all that, an accurate parameter identification is necessary before testing the proposed estimation methods, therefore it will be a future task.

Although the estimation methods will only be tested with the acquired data later on, one measurement is shown in Figure 4.12 for illustration, with the measured motor signal and the motor torque (estimated from the motor current). During the measurements, the rear tire of the E-Scooter was fixed and the BLDC motor was used for steering. The previously mentioned plates (metal plate and the emery paper) were placed under the front tire. Two different signals were actuated, a simple sinusoidal and a sinusoidal curve with linearly increasing amplitude. Both signals were measured with multiple frequencies.

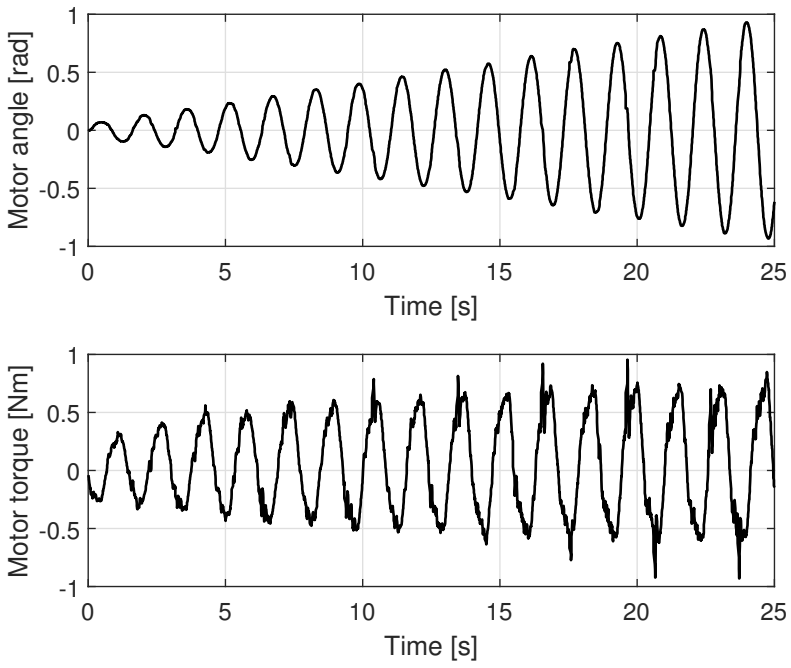


Figure 4.12: Illustration of the measurement signals for one of the experiments. The motor angle and motor torque are shown.

5 Conclusion

5.1 Summary of the achieved results

In Section 1, we defined three goals: (*i*) to set up an accurate simulation of the contact patch dynamics during similar conditions to the self-balancing task, (*ii*) to enhance the accuracy of our previously proposed friction estimation algorithm, and (*iii*) to design and build a piece of experimental equipment for the future self-balancing electric scooter experiments.

A simulation environment with the well-known brush tire model was assembled and based on measurement data, an improved model was presented, which considers different static and dynamic coefficients of friction. The accuracy of the two methods has been compared with open-loop simulation using measurement data, and as expected, the improved brush tire model performed better for all cases. The developed simulation environment will be useful for testing different self-balancing control algorithms in the future.

The previously proposed analytic friction estimation method had several flaws, which resulted in huge oscillations in the estimated signal. To solve this issue, an ANN-based error prediction method has been proposed, which serves as a condition for the estimation. This condition-based estimation does not imply problems during the self-balancing task, because the friction properties do not change quickly with zero longitudinal velocity, while false estimation results could negatively impact the stability of the system. Based on the brush model, a purely numerical estimation method has been presented, which is based on an online optimization problem. The proposed methods have been tested on measurement data and both the ANN-based conditional estimation and the numerical method performed better than the original estimation algorithm.

Lastly, a motor mounting piece has been designed for a Xiaomi Mi Electric Scooter 3, which will enable future experiments with self-balancing applications. The control gains of the BLDC motor have been set to improve the accuracy of trajectory tracking control, and the frictional parameters of the motor have been also identified. We constructed a simple experiment to validate the dynamic behavior of the timing-belt transmission and then different measurements have been made to test the proposed friction estimation methods.

5.2 Outlook, further development

The proposed estimation algorithms have been thoroughly discussed, but to present the full picture, further investigations are needed. The ANN-based error prediction has been only tested on 2 different measurement signals because the remaining data was necessary for training the network. With the developed experimental setup, future measurements can be easily carried out, hence sufficient training and testing data could be provided.

Apart from the above-discussed reason, we plan to evaluate the estimation algorithms using experimental data, which was acquired using the developed measurement setup (namely the self-balancing electric scooter). It is a necessary step to examine the estimation methods during real-life conditions.

After validating the proposed algorithms, the next step is to develop a self-balancing control for the E-Scooter, which utilizes the estimated coefficient of friction. In the literature, various friction compensation methods can be found [26], [27], [28], but to apply a similar method to the self-balancing control of a two-wheeled vehicle is and exciting and unprecedented field of research.

Acknowledgment

I would like to thank Dénes Takács for the huge amount of help and professional support during our work.

Supported by the **ÚNKP-23-2-I-BME-29** New National Excellence Program of the Ministry for Culture and Innovation from the source of the National Research, Development and Innovation Fund.

Kulturális és Innovációs Minisztérium

Nemzeti Kutatási, Fejlesztési és Innovációs Hivatal

List of Figures

List of Tables

Reference

[1] Honda, "Honda riding assist," https://global.honda/innovation/CES/2017/002.html, 2017, [Online; accessed: 2023-09-09].

[2] Xiaomi, "Mi electric scooter 3," https://xiaomiofficial.hu/okoseszkoz/elektromos-rollerek/mi-electric-scooter-3/, [Online; accessed: 2023-09-09].

[3] B. M. Györök, "Estimating the coefficient of friction based on the memory effect of an elastic tire," *Budapest University of Technology and Economics, Students' Scientific Conference*, 2022.

[4] T. Kuno and H. Sugiura, "Detection of road conditions with ccd cameras mounted on a vehicle," *Systems and Computers in Japan*, vol. 30, no. 14, pp. 88–99, 1999.

[5] F. Holzmann, M. Bellino, R. Siegwart, and H. Bubb, "Predictive estimation of the road-tire friction coefficient," in *2006 IEEE Conference on Computer Aided Control System Design, 2006 IEEE International Conference on Control Applications, 2006 IEEE International Symposium on Intelligent Control*, 2006, pp. 885–890.

[6] C. Ahn, H. Peng, and H. E. Tseng, "Robust estimation of road frictional coefficient," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 1, pp. 1–13, 2013.

[7] T. Umeno, E. Ono, K. Asano, S. Ito, A. Tanaka, Y. Yasui, and M. Sawada, "Estimation of tire-road friction using tire vibration model," *SAE Transactions*, vol. 111, pp. 1553–1558, 2002.

[8] L. R. RAY, "Nonlinear tire force estimation and road friction identification: Simulation and experiments," *Automatica*, vol. 33, no. 10, pp. 1819–1833, 1997.

[9] K. Nakajima, M. Kurishige, M. Endo, and T. Kifuku, "A vehicle state detection method based on estimated aligning torque using eps," in *SAE 2005 World Congress and Exhibition*. SAE International, apr 2005.

[10] Y. Yasui, W. Tanaka, Y. Muragishi, E. Ono, M. Momiyama, H. Katoh, H. Aizawa, and Y. Imoto, "Estimation of lateral grip margin based on self-aligning torque for vehicle dynamics enhancement," 03 2004.

[11] J. Svendenius, "Tire modeling and friction estimation," Ph.D. dissertation, Department of Automatic Control, 2007.

[12] H. B. Pacejka, "Chapter 3 - theory of steady-state slip force and moment generation," in *Tyre and Vehicle Dynamics (Second Edition)*. Oxford: Butterworth-Heinemann, 2006, pp. 90–155.

[13] A. O'Neill, J. Prins, J. F. Watts, and P. Gruber, "Enhancing brush tyre model accuracy through friction measurements," *Vehicle System Dynamics*, vol. 60, no. 6, pp. 2075–2097, 2022.

[14] S. Beregi, D. Takacs, G. Gyebroszki, and G. Stepan, "Theoretical and experimental study on the nonlinear dynamics of wheel-shimmy," *Nonlinear Dynamics*, vol. 98, no. 4, p. 2581–2593, Sep 2019.

[15] MathWorks, "Solve system of nonlinear equations - MATLAB," https://www.mathworks.com/help/optim/ug/fsolve.html?s_tid=srchtitle_fsolve_1, [Online; accessed: 2022-10-29].

[16] M. Altrichter, G. Horváth, B. Pataki, G. Strausz, G. Takács, and V. József, *Neurális hálózatok*. Panem Kft., 2006.

[17] Á. Antal and B. M. Györök, "Központi idegrendszer működésének modellezése neurális hálóval egy helyben állás közben," *Budapesti Műszaki és Gazdaságtudományi Egyetem, Tudományos Diákköri Konferencia*, 2021.

[18] S. Beregi, D. A. Barton, D. Rezgui, and S. Neild, "Using scientific machine learning for experimental bifurcation analysis of dynamic systems," *Mechanical Systems and Signal Processing*, vol. 184, p. 109649, 2023.

[19] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D. G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, and X. Zheng, "Tensorflow: A system for large-scale machine learning," in *Proceedings of the 12th USENIX Conference on Operating Systems Design and Implementation*, ser. OSDI'16. USA: USENIX Association, 2016, p. 265–283.

[20] A. Gulli and S. Pal, *Deep learning with Keras*. Packt Publishing Ltd, 2017.

[21] Power Belt, "Main page," https://www.powerbelt.hu/, 2018, [Online; accessed: 2023-09-17].

[22] Raspberry Pi, "Buy a raspberry pi 4 model b," https://www.raspberrypi.com/products/raspberry-pi-4-model-b/, 2021, [Online; accessed: 2023-09-23].

[23] K. J. Astrom and L. Rundqwist, "Integrator windup and how to avoid it," in *1989 American Control Conference*, 1989, pp. 1693–1698.

[24] D. Karnopp, "Computer Simulation of Stick-Slip Friction in Mechanical Dynamic Systems," *Journal of Dynamic Systems, Measurement, and Control*, vol. 107, no. 1, pp. 100–103, 03 1985.

[25] A. Milovanovic, M. Bjekić, and S. Antic, "Permanent magnet dc motor friction measurement and analysis of friction's impact," *International Review of Electrical Engineering*, vol. 6, pp. 2261–2269, 10 2011.

[26] V. van Geffen, "A study of friction models and friction compensation," 2009. [Online]. Available: https://api.semanticscholar.org/CorpusID:9975924

[27] H. Olsson, K. Åström, C. Canudas de Wit, M. Gäfvert, and P. Lischinsky, "Friction models and friction compensation," *European Journal of Control*, vol. 4, no. 3, pp. 176–195, 1998.

[28] C. Canudas, K. Astrom, and K. Braun, "Adaptive friction compensation in dc-motor drives," *IEEE Journal on Robotics and Automation*, vol. 3, no. 6, pp. 681–685, 1987.

Appendix

The used scripts

The used scripts can be found at this Github repository: `https://github.com/gyorokbendeguz/GyB_TDK_2023`. The structure of the directory and user guides can be found on the front page of the repository.

Estimation results

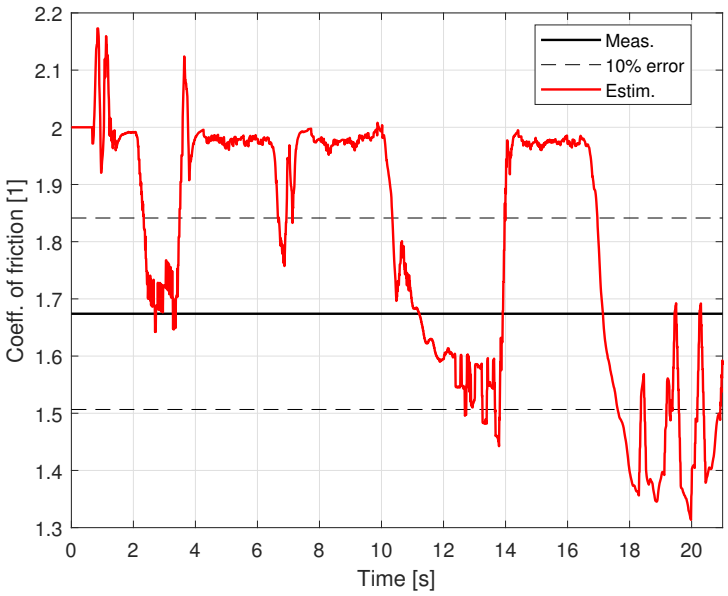


Figure 5.2: Numeric estimation method for the metal plate, measurement #1.

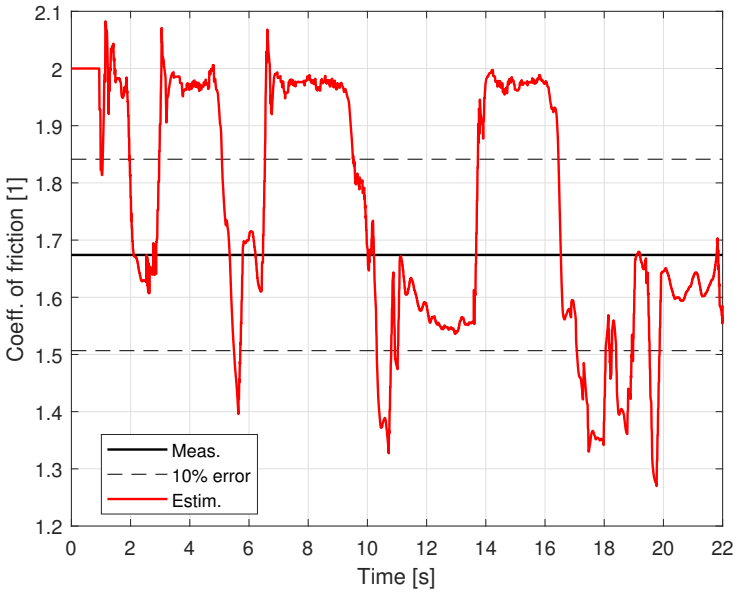


Figure 5.3: Numeric estimation method for the metal plate, measurement #2.

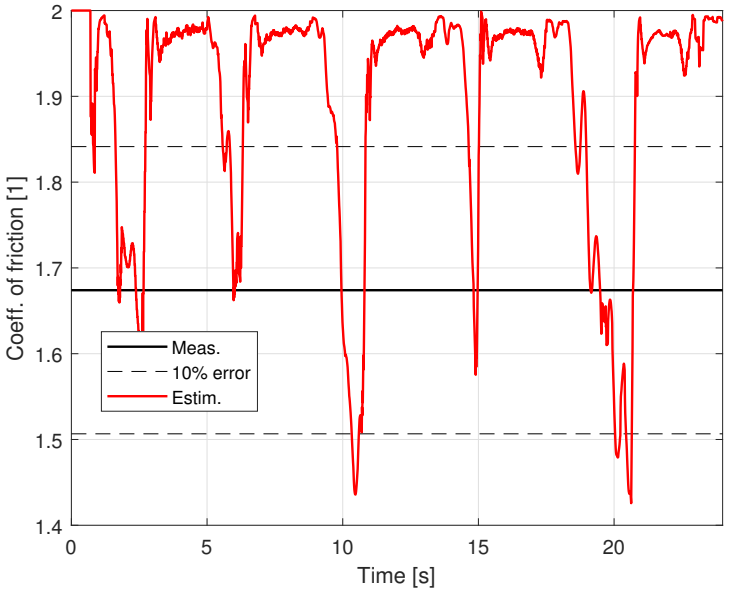


Figure 5.4: Numeric estimation method for the metal plate, measurement #3.

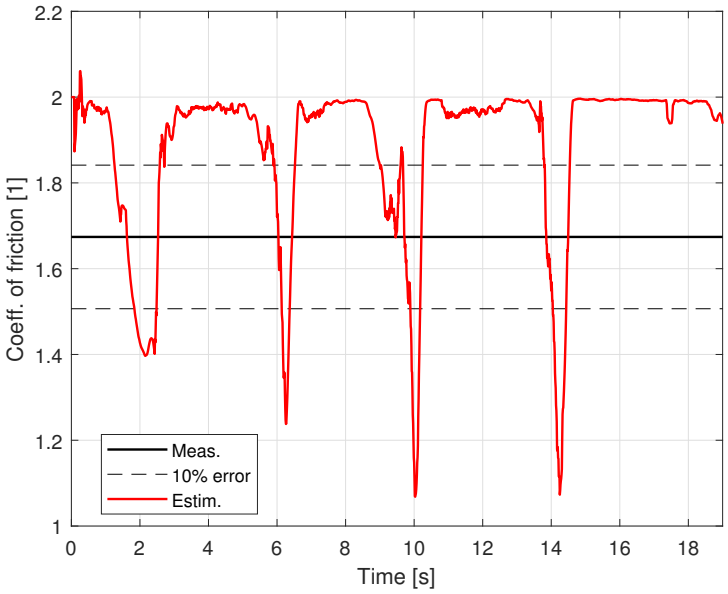


Figure 5.5: Numeric estimation method for the metal plate, measurement #4.

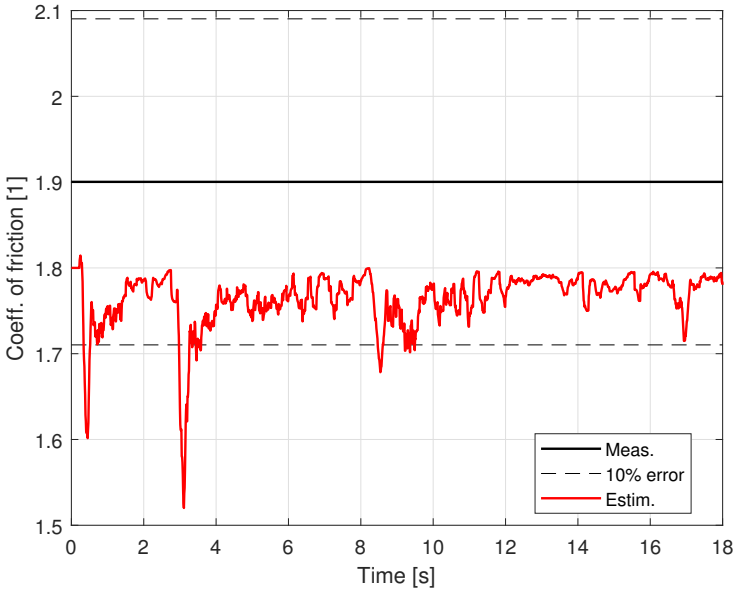


Figure 5.6: Numeric estimation method for the red emery paper, measurement #1.

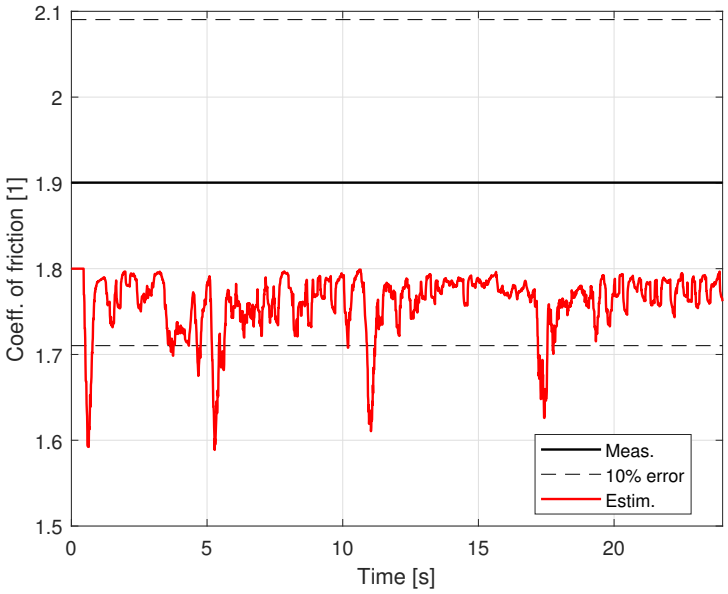


Figure 5.7: Numeric estimation method for the red emery paper, measurement #2.

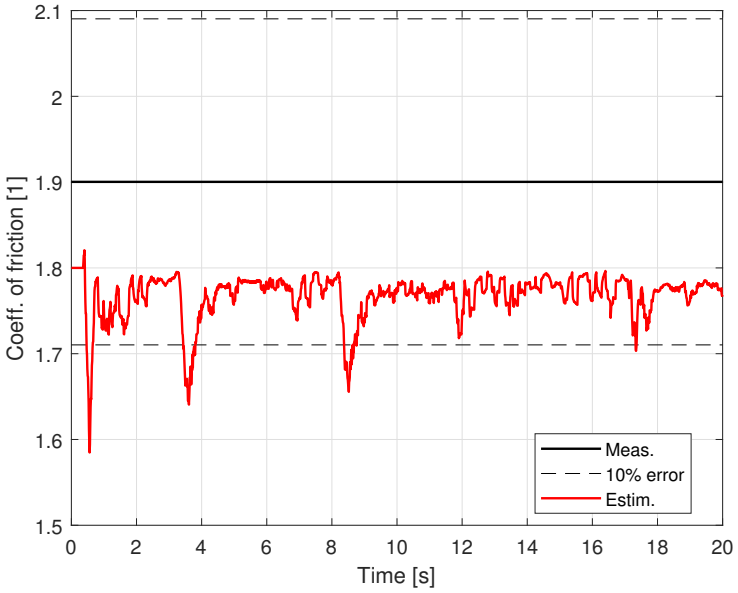


Figure 5.8: Numeric estimation method for the red emery paper, measurement #3.

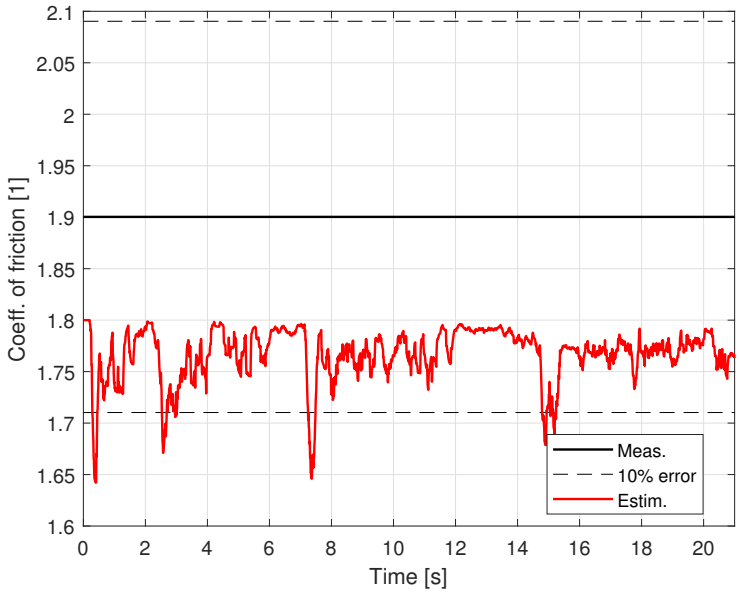


Figure 5.9: Numeric estimation method for the red emery paper, measurement #4.

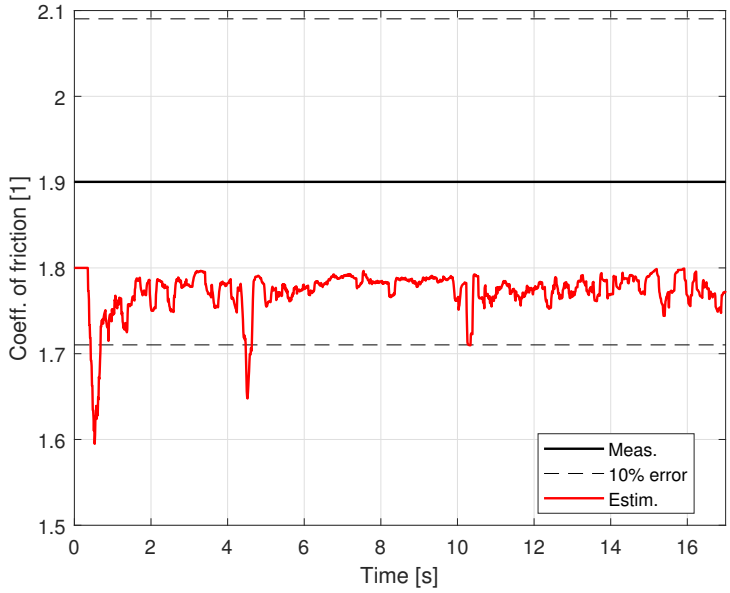


Figure 5.10: Numeric estimation method for the red emery paper, measurement #5.