Riding a Bicycle without Hands

How to do it and the Bicycle Dynamics behind it

Simon Sorgedrager

Delft University of Technology

Riding a Bicycle without Hands

How to do it and the Bicycle Dynamics behind it

by

Simon Sorgedrager

Student number:	4728734
Supervisors:	dr. J.K. Moore, BMechE
	J. Farias
Committee:	Dr. J.K. Moore, BMEchE
	J. Farias
	Dr. G. Papaioannou
Faculty:	Faculty of Mechanical Engineering, Delft

Preface

So here we are, finally. After over seven and a half years of studying at the TU Delft, it is time to say goodbye. When I started back in 2017, I had imagined that I would take a maximum of 6 years to complete. And while it did not end up going the way it was planned, I am very happy that I chose to also enjoy my time here in Delft, and spend a year at the Forze student team. This taught me so much, and I had the most amazing time there, and it might end up being one of the best years of my life.

Firstly, I want to thank my parents. My response to questions on how the exams or courses went was always something like ”yeah alright, I think”, to then end up with a disappointing grade. While we all knew that I was gonna end up fine in the end, I am very grateful for your support throughout this long journey. I would also like to thank my friends in Delft, but also in De Rijp, Alkmaar and other places. Being able to go away from the university and not having to think about studies really helped in me staying relaxed throughout these years, which might have not been so beneficial for my grades, but it was for my mental health. Special thanks goes out to Noah. Living with you for over six and half years has been great. All the beers, whiskys and stupid conversations we’ve had about football, politics and other useless things greatly helped me, and I don’t know how I would have lived in Delft all these years without that.

Similarly, I would like to thank the Bike Lab in particular. I had not expected that I would enjoy being at the university this much, and you all pulled me through my thesis. I would like to thank all the PhD’ers for stopping by and giving us coffee almost every day, this really improved the quality of life at the faculty. Anna, Bart, Eloy and Neville deserve a special notion for being the core crew at the Lab throughout my time, and the laughs, jokes and thoughts we have shared have been immensely valuable to me. Also, many thanks to my other supervisor José. You really helped in giving other perspectives on my problems, and coming up with useful tips to solve those problems, or new ideas to research. I would also like to thank Timo for the incredibly useful meetings we’ve had about BRiM, and that you helped me solve some issues that I could not fix myself.

However, all of this would not have been possible without the Big Boss of the Bike Lab, Jason. I would like to express my sincere gratitude for your guidance throughout my project. Your passion for bicycles, dynamics and control optimization really inspired me to work hard for this project, and I cannot emphasize enough that I think you have been a great supervisor for me. I love how the Bike Lab is managed, and I’m looking forward to seeing what more research will get presented on the website or in TV shows for kids. I am very happy with the thesis I am submitting, and I really hope that you feel the same.

Simon Sorgedrager
Delft, April 2025

Usage of AI Tools

Recently, the possibilities and usage of AI tools have increased significantly. In the development of this thesis, ChatGPT-4.0 by OpenAI has been used in coding tasks in Python and LaTeX. They have contributed in bug-fixing and replacing google searches for coding solutions. Also, the cover image has been created with the help of AI. Every contribution has been reviewed and adjusted where necessary by the author. AI tools have not contributed in the writing of this thesis in any way, as I think the AI tools are not advanced enough that they can replicate my preferred style of writing.

Abstract

Riding a bicycle without hands, while being pleasant and sometimes challenging, exposes the self-stabilizing properties of the bicycle. While human balancing techniques are widely understood, balancing on a bicycle is predominantly done by turning the handle bars and not many details about body motion control strategies are understood yet. Therefore, creating a rider model that controls the bicycle solely by lateral body motion enables these strategies to be amplified and better appreciated. Only controlling the bicycle by lateral body motion also gives more insight into the relation between roll and steering dynamics as the steering control is non-existent. Further inspiration for the model is obtained by analyzing motion capture data of a rider on a bicycle. This thesis describes the process of creating such a hands-free rider models and the development of simulation tasks to examine this control behavior. These tasks involve performing a double lane switch maneuver, a $90°$ turn and riding through perturbations. The model is created using the *BRiM* software package, and simulations are performed by optimal control program *Opty*, which allows for the symbolic equations of the Whipple bicycle model to be used for multi-body-dynamics calculations. After analyzing the control strategies employed by the rider, new insights are obtained on the lateral mechanics of bicycle control, paired with improved joint mechanics that have not been used before in bicycle-rider models. Such new joint mechanism modelling the seat connection between bicycle and rider, that couples lateral translations to body lean proves to be more effective than only rider body lean. Also, in terms of optimal control, using a multi-link pendulum model for the rider does not increase the control performance of the rider compared to a single pendulum rider lean model. Furthermore, novel evidence is presented that disproves the assumed need for counter-steering when attempting to make a turn. The rider can make a steady turning motion without having to initially steer in the opposite direction of the turn. Moreover, the effects of placing a spring between the front- and rear frame of the bicycle, both in stabilizing the bicycle as making the bicycle more responsive are investigated. From this it is found that adding a spring will increase the self-aligning properties of the front wheel but that it hampers maneuverability. Finally, the results of this thesis are put in perspective, and recommendations and improvements for future work are shared.

Contents

List of Figures

List of Tables

Nomenclature

Abbreviations

DoF	Degree(s) of Freedom	vi, 9, 12
CoM	Center of Mass	ii, 1, 4, 5, 7, 10, 31
CoP	Center of Pressure	vi, 7, 9, 10
FBD	Free body diagram	12
BRiM	Bicycle-Rider Models	i-iii, 11, 12, 14, 34, 39
EoM	Equations of Motion	14, 39
RMS	Root mean square error 15, 19, 20, 26-29	

Symbols

q	Coordinate vector of the bicycle-rider system $\in \mathbb{R}^n$.	2
$\dot{\mathbf{q}}$	First time derivative of **q** $\in \mathbb{R}^n$.	2
$\ddot{\mathbf{q}}$	Second time derivative of **q** $\in \mathbb{R}^n$.	2
u	(Angular) velocity vector of the bicycle-rider system $\in \mathbb{R}^n$.	14
$\dot{\mathbf{u}}$	First time derivative of **u** $\in \mathbb{R}^n$.	14
M	Mass matrix.	2
$\mathbf{K_0}, \mathbf{K_2}$	Stiffness matrices.	2
$\mathbf{C_1}$	Damping matrix.	2
λ	eigenvalues of linearized bicycle-rider system.	2
g	Gravity.	2, 12
v	Longitudinal bicycle speed.	2
c	Bicycle trail.	2, 5
ϕ	Rider lean angle.	5, 6
ψ	Bicycle yaw angle.	5, 6
C	Translation factor for the CombiJoint	
$f_{hc}(\mathbf{q})$	holonomic constraint equation	12, 13
$\mathbf{f}(\dot{\mathbf{u}}, \mathbf{u}, \mathbf{q}, \mathbf{r}, \mathbf{p}, t)$	Function for creating the equations of motion.	14
t	Time variable	14
$\mathbf{r}(t)$	$\in \mathbb{R}^m$ vector of input variables at time t.	14
p	$\in \mathbb{R}^p$ the vector of constant parameters.	14
q_0, u_0	Initial state constraints.	14
q_f, u_f	Final state constraints.	14
q_L, u_L	Lower state bounds.	14
q_U, u_U	Upper state bounds.	14
r_L, r_U	Lower and upper input bounds.	14
$\mathbf{x}$	$\in \mathbb{R}^n$ State vector consisting of **q** and **u** at time t.	15
W	Weighting variable in the cost function.	15
w_t	Tracking weight.	15
w_r	Input weight.	15
$q_{i,path}$	coordinates describing the path trajectory of the paths.	15
ε	Error between path and bicycle trajectory at the rear wheel contact point.	15

1
Introduction

For many people growing up in The Netherlands, riding a bicycle is as natural as walking. Many people learn to ride bicycles at a young age starting with a bicycle with small side wheels to avoid falling. The balancing system of children is not yet mature and riding a two-wheeled vehicle, which is fundamentally unstable at low speeds requires practice. Those who ride a bicycle will often notice that, while cycling in a straight line at certain speeds, the bicycle does not really need steering input to maintain its trajectory. While experimenting with this, it is possible to teach oneself to ride a bicycle without having to keep the hands on the handle bars, and you can even control the bicycle to make a turn. When the handlebars cannot be used to steer, the bicycle can be controlled by exerting a lateral force on the bicycle to cause the bicycle to roll. Due to the geometry of the bicycle, roll will induce a rotation of the front frame and thus indirectly steer the bicycle.

The Bicycle

The geometric configuration of the bicycle has come about through many years of development, originating from the pedal-less 'velocipede' to the funny-looking Penny-Farthing bicycle with the huge front wheel, gradually evolving into roughly the geometry of the bicycle as we know it today, with equally sized wheels and tilted and curved front fork, near the end of the 19th century. Around this time, the first mathematical models that describe the dynamics of the bicycle and exhibit self-stability had been independently published by Francis Whipple [1] and Emmanuel Carvallo [2].

Bicycle self-stability

The governing equations of motion of the bicycle describe the forces and moments acting on the bicycle as it moves. As mentioned, these equations have first been described separately by Whipple and Carvallo at the end of the 19th century. Their respective models have been tried and tested numerously in the 100 years that followed. These derivations have been done symbolically or numerically. In 2005, Schwab et al.[3], [4] aimed to properly test and verify them, and created what they called the Benchmark Bicycle Model. This benchmark bicycle confirmed the Carvallo-Whipple model to be accurate for small perturbations when riding straight. The bicycle model of Carvallo-Whipple and the benchmark consist of four rigid bodies as in Figure 1.1; the front frame H and rear frame B connected via a revolute joint, with a front and rear wheel F and R also connected to their respective frames via revolute joints. The mass of the rider is rigidly added to the rear frame, with the CoM of the combined rider and rear frame located around the saddle of the bicycle. The figure shows the CoM's of the four bodies, the exact location of these will naturally depend on the geometric parameters. The wheels have single contact points with the ground P and Q, with no slip occurring between the wheels and ground, which is referred to as having knife-edge wheels in the model. The position and orientation of the bicycle can be described by 8 coordinates. These are the x- and y-coordinates of the rear wheel contact point P, q_1 and q_2. The yaw, roll and pitch coordinates q_3, q_4 and q_5 determine the orientation of the bicycle frame, and q_1 is the steer angle. q_6 and q_8 are the angles of the rear wheel and front wheel respectively. Because in the convention by Moore [5] the wheels are constrained to the ground, holonomic constraints can be formed to simplify numerical integration, and so that the system can be linearized. Because the wheels

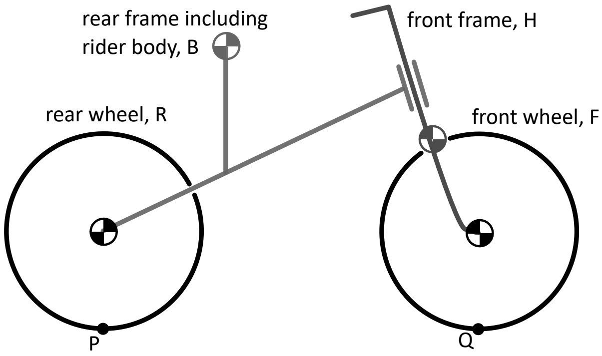


Figure 1.1: Configuration of the benchmark bicycle with its four bodies and their respective CoM's.

are constrained to have contact with the ground, q_5 is not an independent coordinate as it depends on the roll and steer angle of the bicycle. The configuration of the bicycle according to Moore's convention is shown in Figure 1.2.

The benchmark model describes the linearized equations of motion of the bicycle, which can be written in canonical form as shown by Equation 1.1, where the lateral stability equation is a function of the coordinates of the bicycle $\mathbf{q}$ and its first and second derivatives, and the speed of the bicycle v, with the constants being the physical parameters of the bicycle-rider system captured in the Mass matrix $\mathbf{M}$, Stiffness matrices $\mathbf{K_0}$ and $\mathbf{K_2}$ and Damping matrix $\mathbf{C_1}$. The other variable for gravity g can also be considered a constant.

$$\mathbf{M}\ddot{\mathbf{q}} + v\mathbf{C_1}\dot{\mathbf{q}} + [g\mathbf{K_0} + v^2\mathbf{K_2}]\mathbf{q} = f \tag{1.1}$$

With the matrices known, we can determine the eigenvalues of the system by solving the characteristic polynomial Equation 1.2. When solved, the eigenvalues can be plotted against the speed of the bicycle to create a stability plot, as shown in Figure 1.3. Here, the different colored lines represent a specific mode of the bicycle, with the solid lines being the real values, and the dashed line being the imaginary part of the eigenvalue. The blue line represents the weave mode, which is a motion where the bicycle steers in an oscillatory way around the direction of heading. The red line is capsize mode which is connected to the leaning of the bicycle. When in unstable capsize mode, the bicycle cannot correct itself into the fall and will fall over. The green line represents caster mode and this describes the front wheel wanting to follow the direction of travel. When a mode has a positive value it means that it is unstable, and an uncontrolled bicycle cannot stabilize itself properly at this speed. The speed at which a mode eigenvalue crosses zero is called the respective mode speed, i.e. weave speed and capsize speed. Caster is always real and negative for a conventional bicycle, thus there is no such thing as caster speed. Between the weave- and capsize speed there is a region where both real parts of the eigenvalues are negative and consequently the bicycle is self-stabilizing within this speed range, marked in gray.

$$det(\mathbf{M}\lambda^2 + v\mathbf{C_1}\lambda + g\mathbf{K_0} + v^2\mathbf{K_2}) = 0 \tag{1.2}$$

The self-stabilizing properties of the bicycle have long been a topic of both interest and discussion in bicycle research. Whipple described that at certain speeds, the bicycle could correct itself when it was in a fall, and accredited this behavior to the gyroscopic effect of the wheels which would resist a fall of the bicycle by generating a stabilizing torque when it is rotating at speed. This idea that the gyroscopic effect was the reason for self-stability was generally accepted in the 19th century, and stood for quite some time [7].
However the geometry of the bicycle, specifically the front frame is of significant importance. The angle at which the the front fork is tilted with respect to the vertical, or the castor angle, causes the front wheel contact point with the ground to be behind the point where the steering axis intersects with the ground. The distance between these two points is referred to as *trail*, usually denoted by the letter *c*, depicted in Figure 1.4a, which is positive on a conventional bicycle where the steering axis intersection point is

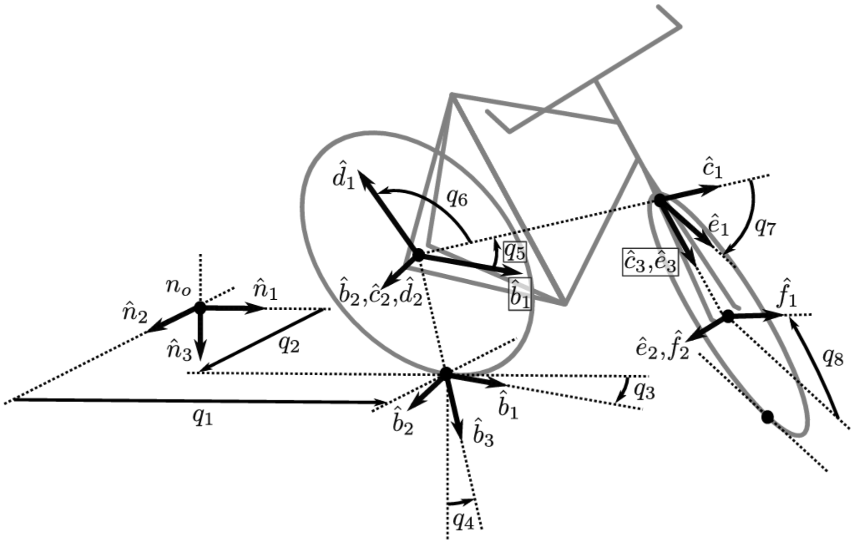


Figure 1.2: Configuration of the Whipple Bicycle Model with its 8 coordinates **q** by the convention of Moore [5]. Reproduced under CC BY 3.0 license [6]

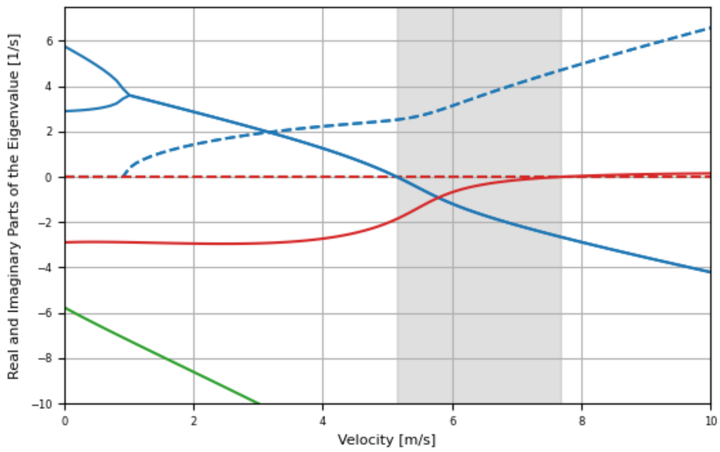


Figure 1.3: Stability plot of a bicycle with rigid rider.
—: Weave , —: Capsize, —: Caster
—: Real parts, - -: Imaginary parts

in front.

Trail has two different effects on the bicycle. The first is the coupling of the steering to the roll angle of the bicycle, which causes the front wheel to steer into the fall. This mechanism can easily be demonstrated by leaning a stationary bicycle to the side; one will notice that the steer and the front wheel will then fall to that same side. While riding, this steering into the fall causes the centrifugal force to pull the bicycle back up to a vertical position again. The other effect that trail creates is a self-aligning torque on the front wheel. When the bicycle is making a turn, there is a sideways force on the tire at the contact point with the ground, and because this contact point is behind the steering axis, it creates a torque that pulls the wheel back to its straight orientation, see Figure 1.4b.

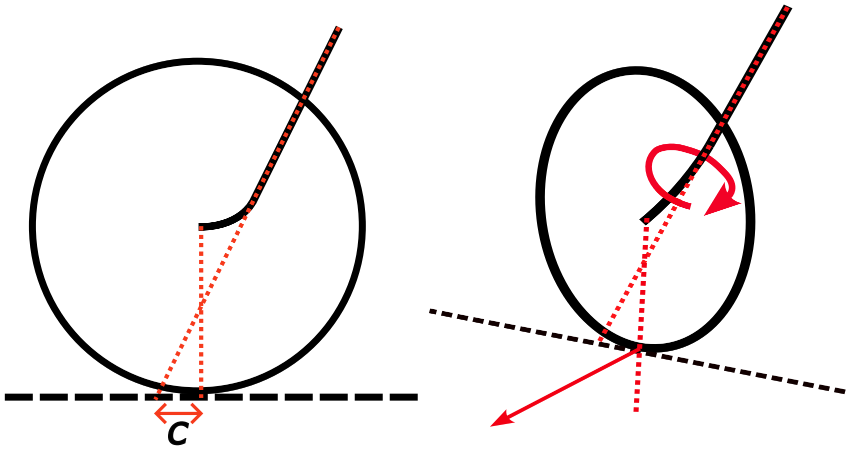
Another geometric trait of the bicycle front assembly that plays a role in self-stability, is the curve or bend that is found in the front fork. This geometry has two effects: Firstly it decreases the trail, which allows

the castor angle of the front frame to increase without getting into excessive amounts of trail, which makes the handling abilities of the bicycle more sluggish [8], and allows the handle bars to be closer to the rider for increased comfort. This offset of the wheel center with respect to the steering axis shifts the center of mass (CoM) in front of the steering axis, and this is also the reason why the handlebars always initially curve forwards on a conventional bicycle, to keep the CoM of the front frame in front of the steering axis. The consequence of this weight distribution is that when the bicycle leans, the gravity on the front frame pulls the wheel in the direction of the lean. In fact, Kooijman et al. demonstrated that a two-wheeled single-track vehicle, in other words a bicycle, that had no gyroscopic effect or trail could still be self-stabilizing by having its CoM in front of the steering axis [9].
Jones aimed to experimentally verify the influence on the ride-ability of the three different mechanisms; gyroscopic, trail and the CoM offset [10]. He found that canceling gyroscopic effects on the front wheel by adding a counter rotating wheel alongside it did not have a noticeable effect on the stability of the bicycle. When testing a bicycle with the front fork reversed thus having the CoM steer the front wheel further out of the fall while leaning, while stating that it was awkward to ride, it did seem to be stable when riderless according to Jones. When moving the front wheel hub to the front, creating negative trail was deemed to be unridable by Jones, and he concluded that trail is the leading factor in bicycle self stability. It is clear from these different experiments that the self-stability of the bicycle does not depend on a single parameter, but is an intricate interplay between the front frame geometries and the traveling speed of the bicycle.

Addition of a Rider

The bicycle being able to stabilize itself while riding at certain speeds, though it is designed to be ridden by a rider. Adding a rider on the bicycle, with its own CoM sitting relatively high, has a significant effect on the dynamics of the system, as the rider makes up the majority of the weight (80-90%) [5]. There have been multiple mathematical and multi-body dynamic models of bicycle-rider combinations created, with varying degrees of rider complexion and interaction, of which some will be described here. As stated, Whipple added the rider as a pointmass to the rear frame, and this is seen as the baseline for rider inclusion in bicycle models, or described as a passive rider model, that does not actively interact with the bicycle. Whipple however did do some analysis on rider interaction with the bicycle, and describes in his paper that there is a speed window between 16.6 km/h and 19.6 km/h, these speeds varying for different bicycles or riders, where the bicycle-rider combination is stable without rider input and that with decreasing speeds, increasingly more control is needed, either by body motion, steering the handle bars or eventually both.
Focusing on modeled rider additions, Bulsink added multiple arm model extensions as adding the mass of the lower arms to the front assembly, arm damping and -stiffness, as well as spring-dampers between body segments [11]. The individual influence of these features, as well as the combined influence on the eigenvalues of the weave- and capsize mode were investigated. The results showed that the passive flexible rider, arm features and the tire model marginally improved the stability of the weave mode, with the exception of arm stiffness, which made the weave mode unstable for any speed. The tire model strongly improved the weave mode stability. Regarding the capsize mode, it was the arm damping features that caused slightly worse performance, while the other rider features slightly improved capsize stability. The tire model however destabilized the capsize mode. Bulsink showed that adding flexible rider properties and more realistic arm properties does have an effect on the stability of the bicycle rider system. The obvious way to control a bicycle is via the handlebars, as that is what they are intended for. While it might seem obvious that to simulate steering input by the rider, it would suffice to add a torque to the handlebars to simulate a steering input. Whipple argues that the change in position of the riders' arms when using the handlebars is negligible and the rider and frame can still be considered a single rigid body, and only a couple has to be added between the front and rear frame. Though the dynamics of an integrated bicycle-rider model are much more complex than that. This is inaccurate in multiple ways as the torque on the handlebars can be applied by the arms of the rider, via the shoulders and the elbows, and the reactionary forces are led back onto the rear frame via the body of the rider. Steering can also be achieved by rotating the upper body around its longitudinal axis and keeping the arms straightened, which has a different but also significant inertial effect on the bicycle. Schwab and Kooijman [12], [13] added these features to the Whipple model, and found that especially the steering from the arms and elbows undo the self-stabilizing features of the bicycle. Whipple also discussed the possibility of riding the bicycle with no hands in his paper. He states that the rider steers the bicycle by

laterally moving his body, and mentions that this movement is likely dependent on the lateral movement of the back frame, which appears to be a notion of the bicycle-rider acting like a double pendulum.



(a) Trail of the Bicycle **(b)** Self-aligning torque while turning left

Figure 1.4: Sketches of Bicycle steering geometry characteristics

Many articles have been published that add a leaning rider feature to a bicycle model by either controlling the lean angle of the rider's upper body, or by applying a leaning torque between the rider and the rear frame. The latter is more realistic according to Roland and Lynch [14] as the reaction torque on the frame is critical for riding dynamics. Figure 1.5 shows how a bicycle with leaning rider is modeled as a pendulum in the frontal plane, with an additional rider lean angle θ relative to the bicycle roll angle ϕ. Garziad and Saka [15] made and analyzed a model with a leaning rider and a steering function, and compared control strategies of a PI, PD and PID controller. Results show that a PI controlled gave the best results, and Garziad and Saka suggest as an explanation that the rider can function as the derivative control action, and state that the rider body has a secondary impact on the stability, whereby the rider is able to stabilize the capsize mode. Wang et al. [16] added a human leaning torque to a bicycle model, with a human control model that includes short-, medium- and long-latency sensory inputs and passive joint stiffness. They compared results of their simulations with measurements from 5 subjects performing cycling tasks on a robotized bicycle, which was used to induce perturbations so that the human balance control model could be estimated and identified. This yielded fairly similar results, implying that their model with just a leaning rider and steering torque can accurately simulate an actual bicycle-rider system. As multiple researchers point out, using a steering torque is much more effective in controlling the bicycle than applying a rider roll torque [5]. However Wang notes that the sensorimotor response of body movement is faster than that of steering actuation which indicates the importance of combined use of both control mechanisms. Cain et al. [17] measured steering angle, rate and torque, and bicycle roll angle and roll rate of 14 different subjects, of whom half of them were skilled bicycle riders and the other half were not. While at slower velocities all riders showed similar balance control performance, it was clear that the skilled riders used more body lean and less steering input than the non-skilled riders at higher velocities. It was observed that these body movement angles were opposite to the bicycle roll angle, and were used to keep the CoM laterally central, just like a double inverted pendulum does. As a result, the experienced cyclists had to use very little steering power to maintain a stable ride. This shows that with good lateral body control, steering via the handlebars is not really needed and hands-free bicycle riding can be quite physically effortless.

Motion Capture

Moore et al. [18] analyzed the motion of three young adult riders on a treadmill using a motion capture system, with 11 markers placed on the bicycle, and 20 markers placed on the subject. They analyzed rider motion during normal pedaling, stabilizing without pedaling, line tracking and also stabilizing with no hands, all within a speed range between 2 km/h and 30 km/h. The data of the no-hands riding was not analyzed in detail, though the two tasks which were analyzed, normal pedaling and stabilizing without pedaling, give great insights into the complex rider motions at different speeds. From the data,

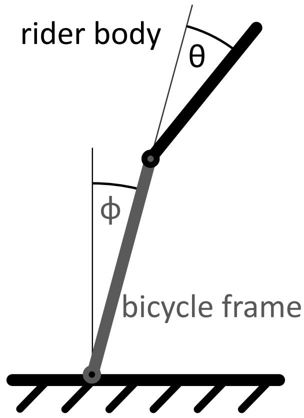


Figure 1.5: Laterally leaning rider on a bicycle, modeled as a double inverted pendulum.

six different rider motions were identified, which are displayed in figure 1.6. The rider data points are plotted with respect to the bicycle frame, and the motions are amplified in the figure for clarity.

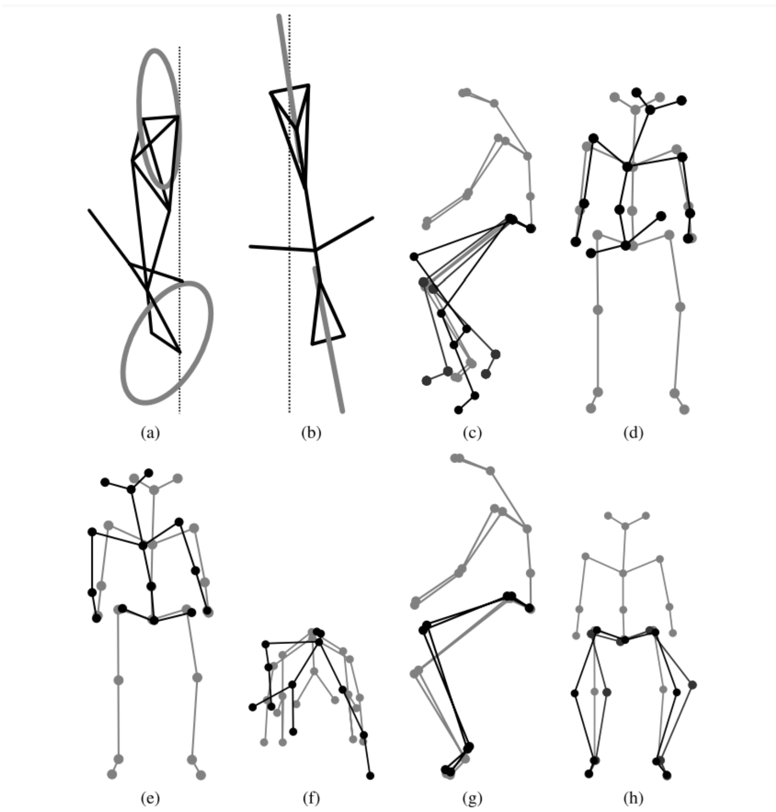
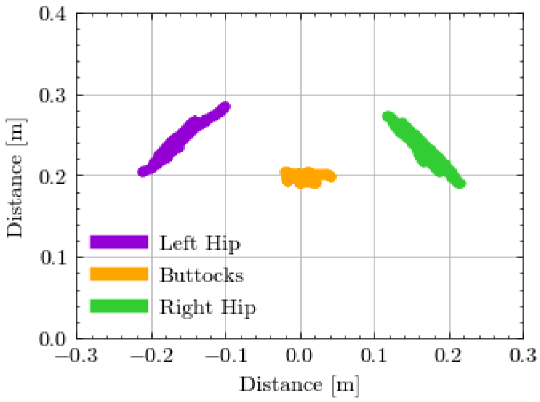
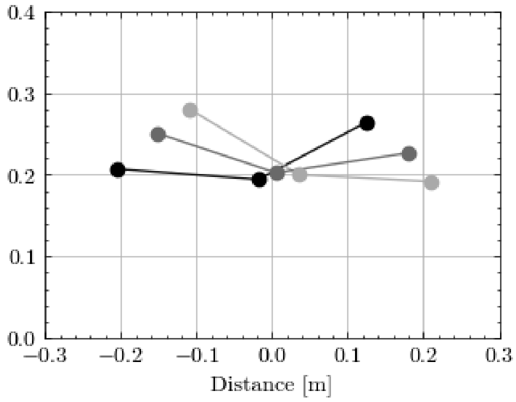


Figure 1.6: Six identified rider motions by Moore et al. [18]: (c) horizontal and vertical components of pedaling, (d) spine bend, (e) rider lean, (f) top view of rider twist, (g) knee bounce and (h) two lateral knee motions. Licensed under CC BY 3.0 [6]

Motion (e) is most similar to the standard lateral rider lean as used in most models like a single inverted pendulum. The rotation "point" of this motion appears to be at the height of the umbilicus, and not at the seat like some models [14], [15], [16]. Motion (d) seems to represent behavior best described as a double inverted pendulum, and the rotation point matches with that of the rider lean model. It must be noted that these motions were identified to be linked to the pedaling motion of the legs, however this data was gathered from cycling tasks that did not require cornering maneuvers or significant body control as the hands were on the handle bars to control steering, so it is unclear what the exact motions would be when making a sharp turn. The twisting motion of the upper body in (f) is only used to turn the handlebars. The lateral motion of the knees (h) was only observed when cycling at speeds below 10 km/h during pedaling. And while stabilizing the bicycle without pedaling, the knee motion was more present, and not only at low speeds. At 25 km/h, no lateral knee motion was observed, and while this is within the stable speed range of the bicycle with a rigidly connected rider, it is possible that with a non-rigidly connected rider the stable speed range is different. It is also possible that because the capsize mode gets unstable very gradually as speed increases, that it requires little body motion to mitigate which is difficult to independently measure. When closely examining the markers around the hips in motion (d), besides a rotation around the bicycle frame, a lateral translation is also visible. To get a better insight in this motion, data from Moore's motion capture during a no-hands riding run at 12 km/h is plotted in Figure 1.7. Here, Figure 1.7a shows the location of the left hip, right hip and buttocks markers, with respect to the rear frame of the bicycle. The view of this plot is oriented in the forward direction, as if the rider is viewed from behind. Figure 1.7b shows three instances of this data, where the three markers of the same instance are connected with lines. This clearly shows that the rotation of the hips is coupled to a sideways translation, and this is in line with observation from multiple researchers, who state that the bicycle can be controlled by such motion; " it is possible to imagine rolling the bicycle frame underneath the body using leg and buttock muscles" [5]. (more references are coming)



(a) Tracking data over a period of time



(b) Three individual frames of the tracking data.

Figure 1.7: Hip and Buttocks movement with respect to the bicycle rear frame, taken from Motion Capture data by Moore et al. [18]

Counter-steering

The phenomenon of counter-steering is well known and described in bicycle and motorcycle research. Fajans [19] describes steering of bicycles as a complicated interaction between centrifugal and gravitational forces. When attempting to make a turn without counter-steering the centrifugal force would make the bicycle lean in the opposite direction of the turn, and continue to fall unless countermeasures are taken by steering back. However, initially turning in the opposite direction of the intended turn will cause the bicycle-rider to lean into the intended turn and shift the CoM laterally outside the bicycle track, which will cause gravity to create a torque that can balance the centrifugal force caused by steering back into the intended turning direction. While this interaction between centrifugal and gravitational forces does indeed occur and is a factor, this description of bicycle steering is far from complete. This is because, as described in chapter 1, the ground contact point of the front wheel moves laterally relative to the bicycle rear frame due to trail. This movement shifts the reaction force, or Center of Pressure (CoP) of the bicycle rider with the bicycle can be balanced laterally. CoP control is discussed in more detail further in this section about human balance control.

Limebeer and Sharp [20] showed that the response of the steer torque and bicycle roll angle are very dependent on travel speed with Whipple bicycle model without leaning rider, only controlled via steer torque. They describe that below the weave speed, to make a turn to the left, there is not actually a steer torque to the left necessary. A steer torque to the right will let the bicycle fall to the left, after which the bicycle steer will follow to the left due to trail. Careful steer torque inputs to the right can prevent the unstable bicycle from falling too far to the left, while keep making a controlled turn to the left. When in its stable speed range, similar behavior by the bicycle is demonstrated, with the difference being that this motion can be achieved while maintaining a constant steer torque to the right. When the bicycle travels above the capsize speed, after an initial steering torque input to the right to initiate the counter-steer movement, a steer torque to the left, or into the corner is required to maintain a turn to the left. This influence of speed on the counter-steering characteristics of the the bicycle highlight the complexity of bicycle steering dynamics.

Aström et al. [21] describe that leaning without counter-steering is possible and sufficient for slow changes of direction as using only leaning to steer has a slow response, and counter-steering is necessary for faster maneuvers. This seems like a statement that debunks the need to counter-steer in order to make a turn. Aström seem to solely count actively turning the handle bars away from the turn initially as counter-steering. Fajans describes that while cycling no-hands, one can make a turn by leaning into the turn. When leaning, the conservation of angular momentum causes the bicycle to roll out of the turn, initiating the counter-steer motion because of the front-frame geometry before the bicycle rolls back into the corner as the center of gravity is shifted into the corner by the body lean. While not actively steering the bicycle, in this way the bicycle counter-steers itself. Prince and Al-Jumaily [22] tested steering behavior of a bicycle with different steering geometries like the head tube angle and front fork rake, and found evidence of counter-steering in all tests, implying that the need for counter-steering is not solely the result of bicycle front frame geometries, though not all configurations have been examined. Fundamentally, balancing a bicycle can be compared with controlling an inverted pendulum on a cart, where the cart must first move shortly in the opposite direction of the intended motion to prevent the pendulum from falling while making the turn. Such a response is called non-minimum phase behavior, and is characterized by having one or more zeros in the right-half plane of its stability plot. Moore [5] shows that the step response to both a steer torque and a roll torque will incur this non-minimum phase behavior, but stays stable when said torques are exerted at a speed of 7 m/s which is in the stable range of the used bicycle-rider model for this test. When applying a steer torque above the capsize speed or below the weave speed, the bicycle obviously needs stabilizing inputs to stay upright, but also here the non-minimum phase behavior is evident. Aström et al. explain that controlling the bicycle by steering torque and rider lean simultaneously, can solve the problem of the right-hand plane zero by having two inputs, and imply that we intuitively use a combination of both control methods when cycling. This is theoretical proof that counter-steering is not essential to make a turn with a bicycle, though to my knowledge this has not been demonstrated further.

Crosswind effect on the bicycle-rider

It is clear that the bicycle can be controlled in many ways by the rider, and while cycling at the right speeds there is very little control input needed to keep the bicycle stable. Though are situations where it is important have robust control over the bicycle, like when riding on very uneven surfaces, or when it is windy outside. It would be interesting to see what kind of control strategy a rider employs when the hands cannot be used to steer the bicycle. At the TU Delft, there is a notorious section of a bicycle path that lies next to the Faculty of Electrical Engineering, Mathematics and Computer Science, which is the tallest building in Delft. This building stands perpendicular in its width to the South-western direction, which ironically is the direction where the strongest winds often come from along the coastal regions of The Netherlands. This building creates a funneling effect for the wind which results in very strong winds perpendicular to the bicycle path, and with it many entertaining videos of students being surprised by it. This leads to the question on how to handle a sudden strong sideways wind when riding with no hands.

Removing the steering

As several researchers have pointed out, using the handle bars is the dominant control method on a bicycle [23], [24]. When both steering control and body control is permitted, a rider will use inputs from the body though these are minimal and it is difficult to measure whether these motions are active control motions or a consequence of pedaling, steering and inertial effects, as Moore [18] claims that his identified rider upper body motions where coupled to the pedaling motions. When a model of a rider is forced to control the bicycle purely by body motion, this could amplify the control methods and give new insights in bicycle control strategies.

Sprung Steer

An interesting case to investigate when no steering control can be used on a bicycle, is the influence of a ”stabilizer spring”. Some bicycles have a spring attached between the front and rear frame of the

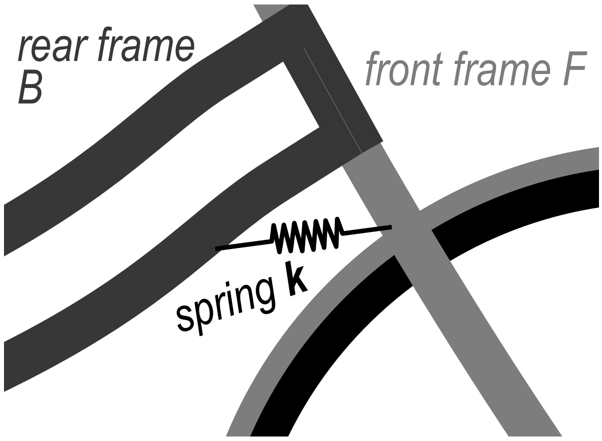


Figure 1.8: Representation of a spring between the front and rear frame.

bicycle. This can often be found on bicycles that have loading basket connected to the front frame. The weight of the basket while it is loaded can cause the front frame to rotate sideways when the bicycle is parked, and even cause the bicycle to fall over. Figure 1.8 presents how such a spring is usually connected on the bicycle. While this spring is often not very stiff, and only requires little more steering torque to achieve the same steering angle as without the spring, the influence on the steering dynamics of such spring can be exposed when the only control input is rider lean.

Human Balance Mechanisms

When the aim is to understand how a person controls a bicycle without using hands, it is important to know how the human body balances in general. This section gives a quick overview of some important human balance mechanisms, and discusses how this can be used in answering the research questions.

Human balance is a complex process involving multiple sensory systems, including the vestibular system, proprioception, and visual input. These mechanisms all have their function in detecting when the human body get in an unstable position, and can also work together in reflexes. The vestibulo-ocular reflex accounts for head rotations by constantly adjusting eye movements to maintain a steady visual field. The cervico-ocular reflex also corrects eye position, but for movement of the neck and trunk. These two reflexes are supported by the cervicocollic reflex which corrects the neck and head in response to trunk movement according to Morningstar [25]. These reflexes aim to maintain head stability, which is in line with the findings of Buchanan and Horak [26] that the main function of vision in balancing is maintaining head and trunk stability in space. This implies that when leaning on a bicycle, the head is maintained stable by leaning the neck in the opposite direction of where the torso leans to.

Any form of balance control by the human body is actuated by muscles. Therefore it is important to understand how the muscles are used, either conscious or in a reflexive manner, during balance control and how this could be implemented in a model. Muscle co-contraction is the activation of two antagonistic muscles to increase muscle stiffness. Van Dieën et al. [27] states that the spine is made stiffer by antagonistic muscle activation during anticipative postural movement or to combat external perturbations. Granata and Orishimo [28] concluded that when there is a higher need for stability, the muscle co-contraction will also increase. Previous studies have predicted muscle co-activation in models with multi-joint muscle models or multiple DoF joints [29], however most biomechanical multi-body optimization simulations are focused on torque or energy minimization, and muscle co-contraction has an adverse effect on that optimization according to Dreischarf et al. [30]. Relating this to human balance control while riding a bicycle, it can be assumed that during hands-free cycling a substantial amount of muscle co-activation around the hip and spine takes place. However implementing this in a model would be counteracting the mechanics that aim to find the most efficient control strategies.

In essence, balancing can be described as the control of the center of mass (CoM) of the body in space, which is often done by altering the CoP of the reaction force of the base on which balance is to be maintained. Multiple researches have been conducted that investigate this CoP control during different tasks like standing [31],[32],[33],[34], walking [35] and sitting [36].

While riding a bicycle, the CoP cannot be moved laterally from the line between the two contact points of the wheels. However as described earlier in chapter 1, the contact point of the front wheel does move laterally while steering, and thus the CoP can be moved laterally with respect to the rear frame of the bicycle. The rest of the balance control on the bicycle by the human can be achieved by applying a roll torque on the rear frame, or by moving its CoM around to lean the bicycle.

Research Questions

To summarize, the initial interest to investigate no-hands bicycle riding originated in a personal affinity with cycling without hands. Upon investigating the dynamics of the bicycle with its complex dynamic properties, and learning about human balancing mechanisms, this interest grew into the objective of creating a bicycle-rider model to that controls a bicycle without hands that can be used to answer several research questions. The first, and underlying question of this thesis is: How can a human ride a bicycle without hands in the best way? When this question is answered, learnings can be used to find out more about counter-steering of a bicycle: Can a bicycle make a turn without counter-steering? From personal experiences, the interest in negating windy conditions while cycling leads to the following question: How do you handle wind perturbations while cycling without hands? Lastly, the existence of a spring between the front and rear frame on some bicycles initiates the following question: What is the effect on steering characteristics and stability by adding a spring between the front frame and rear frame of a bicycle?

2

Methods

2.1. Model Development and Decisions

The model used in this study is designed to answer the research questions specifically, and therefore several decisions have been made. The underlying goal of this study is to find the best control strategies when riding a bicycle without hands, but also to compare different rider models, and determine if different strategies are utilized at different travel speeds. The bicycle-rider models in this research are created using BRiM [37], which is a software package in python developed for this purpose; creating **B**icycle-**Ri**der-**M**odels. A short description on how BRiM is built can be found in **??**. The complete source code developed and used throughout this thesis can be accessed via the following GitHub repository: https://github.com/mechmotum/no-hands-riding [38].

As described in the previous chapter, when a person controls a bicycle hands-free, the torso and head of the person acts like multi-link pendulum. To find out what the added benefit of such a multi-link control approach is, if at all, tasks will be performed by a single-, double- and triple-pendulum rider model. These models consist of the same model parts, however by using either *Weldjoint* or *Pinjoint* connections these models can be configured differently. Two single-pendulum models will be analyzed, where the first "standard" model has the pinjoint located at the seat to represent a single body leaning rider. The second model has the pinjoint located at the *torso joint* and the *seat joint* is a weldjoint. This comparison will highlight what the effect of the location where the torque is exerted is on the control effectiveness.

The hands and arms of the rider will be completely ignored in this model. However, it would be unrealistic to not include the mass of the arms in the model. Therefore, the mass of the arms will be added onto the upper torso body, as if the rider has their arms crossed while performing the maneuvers. Similarly, the weight of the legs can not be neglected in this model as it is relatively large. The legs are a more complicated case than the arms, as they are obviously connected to the rider via the hips, but make contact with the pedals and also transfer loads to the bicycle used to laterally control the bicycle [5]. Because this research focuses on upper body rider control and do not integrate leg motion of the rider, the mass of the legs are added to the rear frame of the bicycle.

Inverted Pendulum

As described in chapter 1, the rider on the bicycle can be modeled as and behaves like an inverted pendulum in real life, and together with the bicycle, which will be referred to as the bicycle-rider, like a multi-link inverted pendulum in the lateral direction. From the rider motion identification data by Moore et al. [18], it can be concluded that the rider itself can behave both like a single- and a double-inverted pendulum. Because a human stabilizes its head in space during balancing tasks, it is decided to also create a triple-inverted pendulum rider model. To accommodate this, the upper body of the rider consists of three bodies: the lower torso, upper torso and the head. The three joints connecting these bodies to the bicycle and each other are the seat, torso-joint and the neck. While adduction of the torso happens through the vertebrae and is thus not a single hinge point, from the motion capture data and by trying the adduction movement yourself, the center of rotation appears to be in the area between the umbilicus and the lowest ribs. To achieve a more balanced mass distribution between the lower and upper torso bodies, the torso-joint is located at the height of the lowest ribs. Configuring the model to be different degrees of freedom is done via assigning different joint types to the connections of the bodies. Figure 2.1 shows a free body diagram (FBD) of the bicycle-rider with the bicycle roll coordinate q_4, and the rider lean coordinates q_{seat}, q_{torso} and q_{neck}. For reference, the coordinate system x-y-z corresponds to the $\hat{n_1}$-$\hat{n_2}$-$\hat{n_3}$ coordinate system as portrayed in Figure 1.2 in the previous section. Figure 2.2 shows the bicycle-rider model plotted, with the colored dots representing the locations of the joints, with the colors matching those of Figure 2.1.

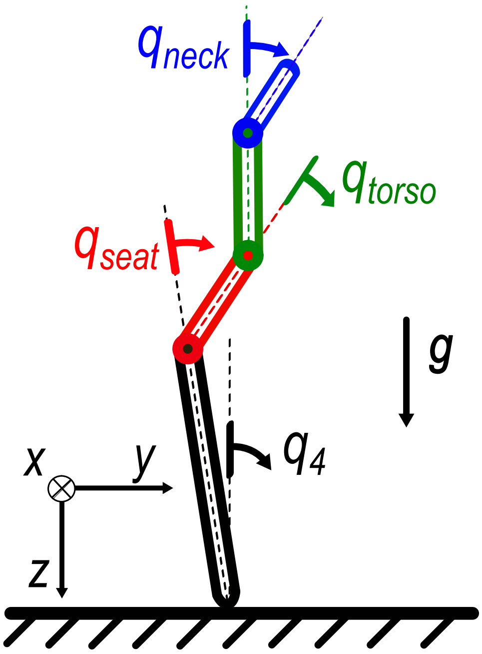


Figure 2.1: Free Body Diagram of the multi-link inverted pendulum in the lateral plane.

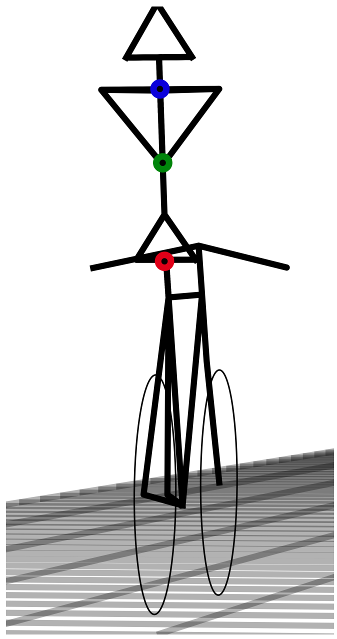


Figure 2.2: Figure of the bicycle-rider in upright position, with the three rider joints highlighted

Joint types

The *SymPy mechanics* toolbox on which BRiM is built, has different joint types to use. To fix a connection a *WeldJoint* is used, and this rigidly connects two bodies together. To allow for a single DoF rotation between two bodies a *PinJoint* is used. Because the bicycle must be controlled in the lateral plane, and while a pitch and yaw moments on the bicycle affect the overall dynamics, a *SphericalJoint* is not necessary, as torsional twist was only identified as being used to turn the handle bars, and torso flexion is not utilized when riding a bicycle as per Moore et al. [18].

From the motion capture data described in Figure 1 it was observed that a rider does in fact lean from the torso, and uses combination of rotation and translation of the buttocks to control the bicycle via the seat. To simulate this motion of combined rotation and translation of the hips around the saddle, a combination of a *PinJoint* and *PrismaticJoint*, a 1 DoF translational joint, is used for the seat. SymPy cannot assign two joints to the same points on a body, so an intermediate body is created. This is a massless body consisting of two points on virtually the same position, so that it physically acts as a singular point. This point body can laterally slide over the bicycle saddle with coordinate $q_{sliding\ seat}$, where the hips and lower torso can laterally rotate around this body with q_{seat}. From Figure 1.7 we can conclude that this translation is coupled to the rotation, and that the motions can not independently happen as all three markers follow a single trajectory. To couple the two motions, a holonomic constraint is placed on the two coordinates. The motion capture data shows that for 40° or rotation, the buttocks marker translates approximately 10 cm. Equation 2.1 shows the calculation of the scaling factor C between the two coordinates and the holonomic constraint equation $f_{hc}(\mathbf{q})$. By differentiating this holonomic constraint equation, SymPy will create a velocity constraint to also bind the velocities of the coordinates. Table 2.1 depicts which joint types the different model connections have for each upper body model used in the simulations, the combined 2 DoF seat joint is referred to as *CombiJoint*. A schematic representation of the *CombiJoint* is shown in Figure 2.3, where the ground represents the saddle of the bicycle

$$\begin{aligned}
40^\circ &= 0.698\ rad \\
0.10m \cdot C &= 0.698\ rad \\
C &= \frac{0.698}{0.10m} = 6.98\ m^{-1} \\
f_{hc}(\mathbf{q}) &= q_{sliding-seat} \cdot C - q_{seat} = 0
\end{aligned} \tag{2.1}$$

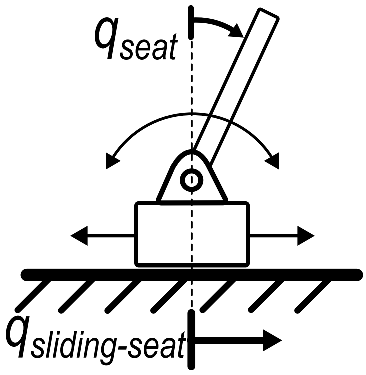


Figure 2.3: Schematic representation of the $CombiJoint$.

Upper Body Model	Seat Joint		Torso Joint	Neck Joint
Single Pendulum-Seat	PinJoint	CombiJoint	WeldJoint	WeldJoint
Single Pendulum-Torso	WeldJoint		PinJoint	WeldJoint
Double Pendulum	PinJoint	CombiJoint	PinJoint	WeldJoint
Triple Pendulum	PinJoint	CombiJoint	PinJoint	PinJoint

Table 2.1: Different models and their associated joint types

Parameters

For the physical parametrization of the bicycle and the rider, data is used from the *Bicycle Parameters* library [39]. This is a database where the physical parameters weight, length and inertia's of the bicycle according to Whipple's benchmark bicycle are stored for different bicycles. Also human rider parameters, measured according to the *Yeadon* method [40]. Yeadon is a software package for Python that employs a method for estimating body segment parameters developed by Fred Yeadon. For all simulations performed in this paper, the parameters used will be of a *Batavus Browser* bicycle and the human parameters will be of dr. Jason K. Moore. This is done as this measured data is already stored in the *Bicycle Parameters* library, and this bicycle-rider combination is also what performed some of the tasks to record the motion capture data in [18]. An overview of the parameters used in this research can be found in B in B.1. The listed rider lengths are used to set the joint locations and plot the rider model. The Bicycle Parameters package has functions to determine and group inertia of different body segments, and thus do not have to be manually added. A better insight at the measuring the bicycle- and human parameters can be found in the Bicycle Parameters git repository [41] and documentation, as well as the Yeadon documentation.

2.2. Simulation and Optimization

Once the model is defined and the system verified, the task optimization are set up. The research questions of this thesis will be answered by having the different bicycle-rider models perform various optimal control problems. Optimal control can be used to find the best solutions to various problem. In this case, what the most efficient control strategy is or, given a certain model with several degrees of freedom. Moreover, these control problems allow for restrictions to be placed on the model, which enables the model to search for possible solution that adhere to these restrictions. In this case that can be used to find out if a bicycle can make a turn without counter-steering. This thesis will use the optimal control software package $Opty$ [42]. This is chosen because it works very well with the symbolic mathematics of $Sympy$, and BRiM is also developed to work with this software. $Opty$ requires certain information to set up and solve an optimal control problem. The verified system expresses the bicycle-rider model in symbolic descriptions of the EoM's created by BRiM in implicit form:

$$\mathbf{f}(\dot{\mathbf{u}}(t), \mathbf{u}(t), \mathbf{q}(t), \mathbf{r}(t), \mathbf{p}, t) = \mathbf{0} \tag{2.2}$$

where:

- t is time
- $\dot{\mathbf{u}}(t) \in \mathbb{R}^n$ vector of model accelerations at time t
- $\mathbf{u}(t) \in \mathbb{R}^n$ vector of model speeds at time t
- $\mathbf{q}(t) \in \mathbb{R}^n$ vector of model coordinates at time t
- $\mathbf{r}(t) \in \mathbb{R}^m$ vector of input variables at time t
- $\mathbf{p} \in \mathbb{R}^p$ is the vector of constant parameters

The optimizer needs a starting position for the model, and for that initial state constraints q_0 and u_0 need to be provided as in Equation 2.3. Not all states and speeds need a defined initial value, if some are left undefined the optimizer assign a value to them.

$$\begin{aligned} \mathbf{q} &= \mathbf{q_0} \\ \mathbf{u} &= \mathbf{u_0} \end{aligned} \tag{2.3}$$

For the optimization simulation to be completed, the final state constraints q_f and u_f will have to be satisfied by the system. Similarly to the initial state constraints, not all speeds and coordinates require a final state value as in Equation 2.4.

$$\begin{aligned} \mathbf{q} &= \mathbf{q_f} \\ \mathbf{u} &= \mathbf{u_f} \end{aligned} \tag{2.4}$$

If the start- and end-coordinates and speeds of the system are defined, the optimizer will also require the physical bounds in which the q's and u's can be controlled. Adding these bounds will not only prevent the model from taking on unrealistic positions, it will also help the optimization speed as $opty$ has a smaller window of possibilities to evaluate to find the optimal solution. The unknown inputs r also have to be bounded as the optimizer needs a range to find the optimal solution within that range. Setting the range as narrow as necessary will speed up the simulation, but may result in finding a sub-optimal solution. In the optimizations conducted for this research, the inputs have a very wide window because the objective function includes minimizing the torque exerted by the model. While this does increase the computing time, it gives the model more freedom in finding the optimal solution.

$$\begin{aligned} \mathbf{q_L} \leq \mathbf{q} \leq \mathbf{q_U} \\ \mathbf{u_L} \leq \mathbf{u} \leq \mathbf{u_U} \\ \mathbf{r_L} \leq \mathbf{r} \leq \mathbf{r_U} \end{aligned} \tag{2.5}$$

Performance metrics

When the operating bounds for the optimization are defined, the optimizer requires a target to optimize between the start- and end-positions. This is in the form called the *cost-function* $J(\mathbf{x}, \mathbf{r})$, also referred to as the *objective function*. Here, **x** is the system's state vector, which consists of $\mathbf{q}$ and $\mathbf{u}$ as stated in Equation 2.6. The cost-function, stated in Equation 2.7a, consists of a function that sums the squared torques r_i exerted by the model over the simulation, and a function $f(\mathbf{x})$ in Equation 2.7c that includes the distance from the path trajectory to the coordinates in the state vector $\mathbf{x}_{qi(t)}$ that determine the location of the bicycle. Both of these parts have a weight assigned to them that is between 0 and 1, calculated according to Equation 2.7b, where w_t represents the tracking weight and w_r is the input weight, which will be elaborated on shortly.

$$\mathbf{x} = \begin{bmatrix} \mathbf{q} \\ \mathbf{u} \end{bmatrix}^\top \tag{2.6}$$

The optimizer aims to minimize the costfunction, and it determines that an optimal solution is found when the the numerical value of the constfunction has converged to minimum. Because there can be multiple local minima in the costfunction, and ideally the global minimum is reached by the optimizer, it will do iterations around a found minimum to see if that opens up a path to a better solution. Because the optimizer cannot know for sure if a global minimum is found, the optimizer can be pushed into finding different solutions by providing the optimization with an *initial guess*, this will be further discussed at the end of this section.

$$J(\mathbf{x}, \mathbf{r}) = W \cdot f(\mathbf{x})^2 + (1 - W) \cdot \sum \mathbf{r}_i^2 \tag{2.7a}$$

$$W = \frac{w_t^2}{w_t^2 + w_r^2} \tag{2.7b}$$

$$f(\mathbf{x}) = \sum_{i=1}^{2} (q_{i,path} - \mathbf{x}_{q_i(t)}) \tag{2.7c}$$

Changing the weights of the system enables the user to skew the performance of the simulation to either a very torque-efficient or a very accurate line tracking execution of the task. Because the numerical value $f(\mathbf{x})^2$ depends on the duration (and thus the amount of time steps of the simulation), and $\sum \mathbf{r}_i^2$ depends strongly on the particular model that is used, it is difficult to have comparable results across different simulations. Therefore it is chosen to play with the weights for each simulation, and to target a certain path tracking performance for some tasks. The tracking performance is measured by calculating the RMS (Root-Mean-Square) error of the location of the rear wheel with respect to the determined path. This is done by dividing the paths total length over the amount of nodes N into the x- and y-coordination called $q_{1,path}$ and $q_{2,path}$, so that the distance between these points and the actual path the bicycle follows in the executed simulation $q_1(t)$, $q_2(t)$. The calculation of this RMS error RMS is shown in Equation 2.8.

$$\begin{aligned} q_{1,\varepsilon} &= q_{1,path} - q_1(t) \\ q_{2,\varepsilon} &= q_{2,path} - q_2(t) \\ \varepsilon &= \sqrt{q_{1,\varepsilon}^2 + q_{2,\varepsilon}^2} \\ RMS &= \sqrt{\frac{1}{N} \sum_{i=1}^{N} \varepsilon_i^2} \end{aligned} \tag{2.8}$$

For the simulations conducted in this paper, the numerical value of the sum of the model torques $\sum \mathbf{r}_i^2$ is consistently much higher than that of the path tracking value. Therefore, the weighting for the calculation of W in Equation 2.7a is heavily skewed to the tracking weight w_t, and changing the weights is done by adjusting w_t, while the torque weight w_r will be kept as a value of 1. To quantify the effort that the model has to do to perform a task within a certain RMS path following error, the used energy from the joint torques is calculated. The energy, or work, done by the system can be calculated by multiplying the exerted joint torque r_i with their respective angular velocity ω_i to get the exerted power, and then integrating this over the length of the simulation, as done in Equation 2.9a to get the energy, where P is the indicator for which pendulum model is used and thus how many torque actuators are present in the model. Because the simulations are done in discrete time, Equation 2.9b shows how this function is used in the model where k stands for the timestep and N is the total amount of timesteps of the optimization. Because the human muscular actuation is not a conservative system, and a negative value from multiplying torque by angular velocity does not mean that energy is regenerated by the muscles, the absolute value of this number is used when adding the power

$$J_c = \sum_{i=1}^{P} \int_{t_0}^{t_f} r_i(t) \cdot \omega_i(t)\, dt \tag{2.9a}$$

$$J_d = \sum_{i=1}^{P} \sum_{k=1}^{N} |r_i[k] \cdot \omega_i[k]| \cdot \Delta t \tag{2.9b}$$

Initial Guess

As mentioned, the optimizer has no way of knowing if it has found the *global minimum*, or is stuck in a *local minimum*. Therefore there is an option built in to the code to quickly provide different initial guesses. These different options with description are presented in Table 2.2. Initially, all optimizations will be ran with simulated initial guesses. However, when an optimization displays vastly different behavior than similar tasks one of the other options will be used until a satisfactory result is achieved. The use of the outcome of a previous simulation is used to push a simulation in a certain direction if no other option gives a good result. It can also happen that the Simulated initial guess can not simulate correctly and crashes the whole simulation, in which case other initial guess options are used.

Initial Guess	**Description**
Simulated	Runs a simulation with the initial and final constraints, system bounds, and system inputs.
Zeros	Fills the state vector **x** with zeros.
Ones	Fills the state vector **x** with ones.
Random	Fills the state vector **x** with random values between $-\frac{\pi}{2}$ and $\frac{\pi}{2}$.
Path	Fills the state vector **x** with zeros, except for q_1 and q_2, which will follow the task path trajectory divided over the optimization length.
Previous guess	Uses the state vector **x** of the outcome of a previous optimization as the new initial guess.

Table 2.2: Available strategies to initialize the initial guess in the optimization process.

2.3. Optimization Tasks

To answer the vastly differing research questions proposed in this report, letting the different models perform a single task to compare the performance of these models, a single task would not suffice.

2.3.1. Double Lane Switch

The first tasks performed by the model are focused on what is the most effective way to control the bicycle. For this, a simple lane-switching task will be performed, where the bicycle will do a lateral translation of X_{lat} is 2m, after which the bicycle will return to its 'starting lane' and continue in the longitudinal direction. The trajectory follows the function of equation 2.3.1, where s is 6m, making the total longitudinal length of the trajectory X_{long} is 30m. Additionally, the path is depicted in Figure C.3b.

$$q_2(q_1) = \begin{cases} 0, & q_1 < s \\ \frac{X_{lat}}{2} \cdot (1 - cos(\pi \frac{q_1 - s}{s})), & s \leq q_1 < 2s \\ X_{lat}, & 2s \leq q_1 < 3s \\ \frac{X_{lat}}{2} \cdot (1 + cos(\pi \frac{q_1 - s}{s})), & 3s \leq q_1 < 4s \\ 0, & 4s \leq q_1 \leq X_{long} \end{cases} \tag{2.10}$$

This path tracking task provides a good reflection for the amount of control effort needed to perform the task, as it involves making a turn from straight, upright riding during the initial lane switch, a direction change to make the second lane switch back, and it requires the bicycle-rider to straighten itself before reaching the end of the path. The initial state constraints $\mathbf{x_0}$ and final state constraints $\mathbf{x_f}$ for the double lane switch task are given below in 2.3.1. Here, all the independent bicycle coordinates are set to be zero, ensuring that the bicycle starts in an upright position. The rider model, of which the coordinates are grouped as q_{rider} to encompass the different configurations that can consist of 1 to 4 coordinates, are also set to be zero to ensure an upright start to the task. A similar reasoning is present for the listed initial speeds of the bicycle and the rider, which ensures that the model starts the task stable, and not already moving in a certain direction. This is to make sure that all starts of the different comparisons are equal, and thus the results can be compared better. The unlisted initial bicycle speeds are not constrained as they represent the longitudinal velocity of the bicycle. This value is left open for the optimizer to choose, as it will depend on the distance traveled to complete the task within the set duration. This traveled distance can change based on the weighting of the torque and tracking in the objective function. The bicycle and rider speeds do not have a final state constraints, as it is not deemed important for the model to be in a static state. This gives the model some freedom to determine the most effective way to get to the final position. The absence of initial or final state constraints of q_5 and u_5, which are the bicycle pitch angle and rate, is due to this coordinate being dependent on q_4 and q_7, so constraining this value would have no added value and only make computation more difficult.

$$\begin{aligned} &\mathbf{x_0}: \\ &q_{1,0} = 0.0m & q_{rider,0} &= 0.0^\circ \\ &q_{2,0} = 0.0m & u_{2,0} &= 0.0^\circ/s \\ &q_{3,0} = 0.0^\circ & u_{3,0} &= 0.0^\circ/s \\ &q_{4,0} = 0.0^\circ & u_{4,0} &= 0.0^\circ/s \\ &q_{6,0} = 0.0^\circ & u_{7,0} &= 0.0^\circ/s \\ &q_{7,0} = 0.0^\circ & u_{rider,0} &= 0.0^\circ/s \\ &q_{8,0} = 0.0^\circ \end{aligned}$$

$$\begin{aligned} &\mathbf{x_f}: \\ q_{1,f} &= X_{long} = 30.0m \\ q_{2,f} &= 0.0m \\ q_{3,f} &= 0.0^\circ \\ q_{4,f} &= 0.0^\circ \\ q_{7,f} &= 0.0^\circ \\ q_{rider,f} &= 0.0^\circ \end{aligned}$$

The bounds for the system are presented in the lists below. Their function is to make sure that the bicycle-rider model behaves in a "natural" way, meaning that there can't be motions that a rider would never make while riding a bicycle. The bounds on q_1 and q_2 are set in such a way that the bicycle stays within a box around the path that is to be tracked, with some extra margin so that the bicycle is allowed to overshoot the turns or counter-steer if it finds that will benefit the overall objective that is given. The bounds on the rider model joints are estimates of how far the joints can rotate while riding a bicycle.

$$\begin{aligned} -0.2m \leq q_1 &\leq 5s + 0.2m & -45^\circ &\leq q_{seat} \leq 45^\circ \\ -0.5m \leq q_2 &\leq X_{lat} + 0.5m & -0.1m &\leq q_{seat\ shift} \leq 0.1m \\ -120^\circ \leq q_3 &\leq 120^\circ & -45^\circ &\leq q_{torso} \leq 45^\circ \\ -45^\circ \leq q_4 &\leq 45^\circ & -60^\circ &\leq q_{neck} \leq 60^\circ \\ -75^\circ \leq q_7 &\leq 75^\circ \end{aligned} \tag{2.11}$$

2.3.2. 90 Degree Turn

To investigate the actual steering dynamics of the bicycle-rider model while turning, specifically to look counter-steering, a trajectory tracking task of a single 90 degree turn is performed by the model. This trajectory follows a straight of s is 6m, then a turn with radius r is 6m, to then follow a straight of equal length to the first, which is depicted in Equation 2.12. A visualization of this path is shown in Figure C.3a. Because the bicycle only has to turn in one direction, the optimizer can be instructed to prohibit the bicycle from steering and heading in the other direction. If the optimizer is still able to find a solution to make the turn, this could be considered as evidence against the need for counter-steering.

$$q_2(q_1) = \begin{cases} 0, & q_1 < s \\ r - \sqrt{r^2 - (q_1 - s)^2}, & s \leq q_1 < s + r \\ r \leq q_2 \leq s + r, & q_1 = s + r \end{cases} \tag{2.12}$$

The initial- and final state constraints of the 90 degree turn task look similar to those of the Double Laneswitch task, with the model expected to start the task in a steady, straight-up position, and also ending up in a straight-up position. Obviously, because the path taken is different, the final positions and heading of the bicycle are different and the final state constraints $\mathbf{x_{f,90^\circ turn}}$ specifically for this task are listed below in Equation 2.13. Coordinates and speeds that are not listed, are equal to those of the Double Laneswitch task.

$$\begin{aligned} &\mathbf{x_{f,90^\circ turn}}: \\ &q_{1,f} = s + r = 12.0m \\ &q_{2,f} = r + s = 12.0m \\ &q_{3,f} = 90^\circ \end{aligned} \tag{2.13}$$

For the task, the bounds are adjusted to accommodate the different path of the model in $\mathbf{x_{90^\circ turn}}$. To find out if the model can make the turn without counter-steering, two bounds on the bicycle have to be tightened to prohibit the bicycle from counter-steering. These are the yaw angle q_3 and steering angle q_7 and its bounds are defined in $\mathbf{x_{90^\circ turn,no\ CS}}$, shown together with the translational bicycle bounds in 2.14.

$$\begin{aligned} &\mathbf{x_{90^\circ turn}}: && \mathbf{x_{90^\circ turn,no\ CS}}: \\ &0 \leq q_1 \leq \ s + r + \ 0.5m && 0.0^\circ \leq q_{3,no\ CS} \leq 135^\circ \\ &-0.5m \leq q_2 \leq s + r \ + \ 0.5m && 0.0^\circ \leq q_{7,no\ CS} \leq 75^\circ \end{aligned} \tag{2.14}$$

2.3.3. Perturbed Riding

When riding hands-free in a straight line, at velocities where the bicycle itself is stabilizing, controlling the bicycle requires little control effort even with a rider on the bicycle. Though when riding in the real world, there are external forces acting on the bicycle-rider in the form of road imperfections and or wind. To simulate the rider cycling in windy conditions, a time-varying force is placed on the bicycle saddle in lateral direction. The amplitude of this force is defined by a function of time. Though the objective of the task is not necessarily about tracking a line, a line tracking objective is still included to encourage the model to stay in a reasonably straight line, following Equation 2.15. As with the previous paths, this path is depicted in Figure C.3c.

$$q_2(q_1) = \begin{cases} 0, & 0 \leq q_1 \leq 5s \end{cases} \tag{2.15}$$

The amplitude of the wind force is modeled as as a curve that starts with an amplitude of zero, and quickly rises to a certain amplitude and oscillates around that value. Fintelman et al. [43] measured the effects of crosswinds on cyclists for different yaw angles. For a yaw angle of 90°, a C_dA of approximately 0.75 m^2 was found. That leads to a wind force of F_w with v_w being the sideways wind velocity. A strong side wind that is challenging to cycle through, but would not blow you of the bicycle is around estimated to be 55 km/h, or 15.3 m/s. Following the formula for F_w, with ρ = 1.23 kg/m^3 and $v_w = 15.3$ m/, this results in a wind force of approximately 108 N as depicted in Equation 2.17. Therefore the wind force is determined to oscillate around 100 N, with the peak in wind force to be 108 N. The time-varying formula describing F_w is stated in Equation 2.17. While this equation seems unnecessarily complicated, it is finetuned to make sure the initial value is equal to zero. Figure C.4 shows the wind function over a period of 9 seconds, which is the longest duration for this task.

$$
\begin{aligned}
F_w &= \frac{1}{2}\rho C_d A v_w^2 \\
&= \frac{1}{2} \cdot 1.23 \cdot 0.75 \cdot 15.3^2 \\
&\approx 108\ N
\end{aligned} \tag{2.16}
$$

$$
F_w(t) = 100 + \frac{-20}{x + 0.1335} + cos(\frac{x - 0.05}{5}) + 4sin(3(x - 0.05)) + 8cos(4(x - 0.05)) \tag{2.17}
$$

Fintelman et al. showed that there is a negative moment around the vertical axis between the front and rear wheel exerted by the side wind, which indicates that the resultant force of the wind on the bicycle-rider system is rearwards of the center point between the two wheels of $C_y A$ = -0.025. Equation 2.18 shows that this resultant force would be a distance d_w approximately 3.3 cm behind the center of the wheelbase. This moment was measured with a rider that is leaning forwards on the handlebars, and thus it can be assumed that when the rider sits upright, or even leaning back slightly as it does during hands-free riding, the resultant force will be located further backwards. Therefore, the location where the wind force is applied is set to be at the saddle point of the model.

$$
\begin{aligned}
M_w &= \frac{1}{2} \cdot 1.23 \cdot -0.025 \cdot 15.3^2 \\
&= -3.6\ Nm \\
d_w &= \frac{M_w}{F_w} = \frac{-3.6}{108} = -0.033\ m \\
&= 3.3\ cm
\end{aligned} \tag{2.18}
$$

2.3.4. Sprung Steering

To get a good understanding of the effect of a spring between the front and rear frame on cycling, optimizations with such a spring are conducted for two different tasks. Firstly, the Double Lane Switch task as described in subsection 2.3.1 will be performed using different spring rates, while riding at 15 km/h. The threshold for "sufficient" path tracking here is determined to be an RMS error of 0.20 m, to make sure this tracking is reachable with the more extreme spring rates. Because the required torque rises exponentially when the tracking difficulty increases for the model, as this was observed during the Double Lane Switch task at velocities of 19.5 km/h, and this should keep the required energies by the model in a reasonable window. Spring rates used for this task range between k = 30 Nm/rad and k = -15 Nm/rad and are displayed in Table 2.3 below. The steps between the spring rates is 5 N/rad, with smaller steps being used around zero as the expectation is that the results here could show some interesting behavior. The aim of this task is to see what effect such a spring has on the handling performance of the bicycle. While a negative spring rate is not something that is used in the real world, this model enables us to test it anyway which might lead to new insights.

-15	-10	-5	-2.5	-1	0	1	2.5	5	10	15	20	25	30

Table 2.3: Used spring rates k [Nm/rad] during the Double Lane Switch task with sprung steer.

To test the effect of a spring between the front and rear frames on the stability of the bicycle, the Perturbed Riding task described in the previous subsection 2.3.3 is performed with a spring. Because the stability of the bicycle is very dependent on the speed of the bicycle as discussed in chapter 1, this task is performed at five different speeds; 12, 15, 18, 21.5 and 25 km/h

Task	$RMS\ [m]$	w_t
Double Lane Switch	0.10	-
90 deg Turn	-	1000
Wind Cycling	-	100
Sprung Steer / Double Lane Switch	0.20	-
Sprung Steer/ Wind Cycling	-	100

Table 2.4: Overview of the different tasks, and what their optimization tracking objective or tracking weight value is

3

Results

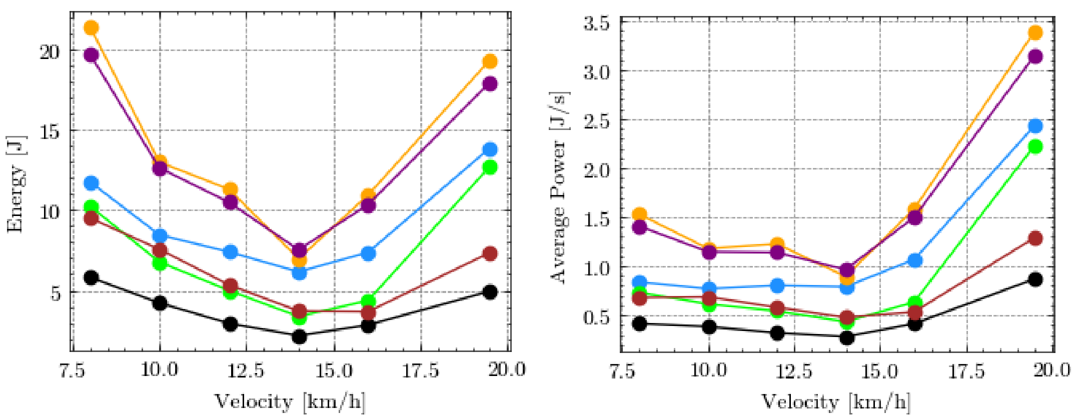
The results of the optimizations are presented in this chapter. Firstly the control performance of the different rider models will be compared on the double lane switch task. Then, findings on the need for counter-steering when making a turn will be discussed. This will be proceeded by an analysis on how a rider can handle being hit by a strong sideways winds. Lastly results will be presented on how having a spring attached between the front- and rear frame can influence the dynamics and controllability of the bicycle.

3.1. Bicycle Control performance

Here, the results from the double lane switch task described in subsection 2.3.1 are presented, by first comparing the simulation at varying cycling speeds, after which the different model features are discussed. While not being a separate research question, it is decided that it is beneficial for the rest of the results to discuss the effect of the travel speed first.

3.1.1. Influence of Cycling Speed

Figure 3.1a shows the required energy by the different models to perform the double lane switch task. Because the different speeds result in different durations of the simulation, the data is also plotted divided by time in Figure 3.1b as the average power used per task. Both the total energy required and the average power values show that the tested velocity of 14 km/h is the most efficient velocity in performing this specific path following task. While the difference in required power is not large between the range of 8 to 16 km/h, it is clear that the runs at 19.5 km/h is significantly more demanding for the model. The specific energy consumption of the used models will be discussed in the following sections.

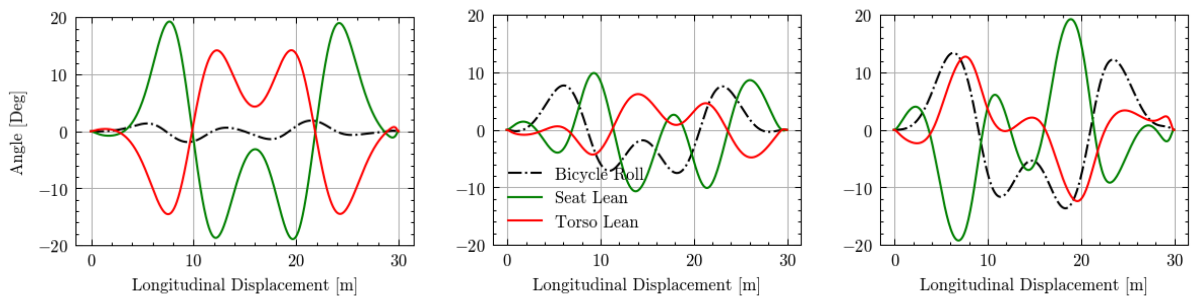


(a) Energy required

(b) Average power

Figure 3.1: Effort of different rider models plotted for different speeds

Figure 3.2 shows the different control strategies the double pendulum model with PinJoint seat joint employs while making the double lane switch maneuver at three different speeds. At 8 km/h, in 3.2a it can be noted that the bicycle roll angle is quite low, and the amplitude of the rider angles are relatively large, while the steering angles are very comparable, as the same path is followed with near equal error. Another noticeable difference, is that during the 8 km/h run the bicycle roll angle seems to follow the seat lean angle, and at 14 km/h bicycle roll angle is going in the opposite direction of the seat lean. When riding at 19.5 km/h this is even more exaggerated, as the initial turning maneuvers of the lane switches, not the part where the bicycle returns on a straight trajectory, require the biggest leaning angles of the whole maneuver. For reference, the yellow line in Figure 3.1 represents the double pendulum model with PinJoint seat.



(a) 8 km/h **(b)** 14 km/h **(c)** 19.5 km/h

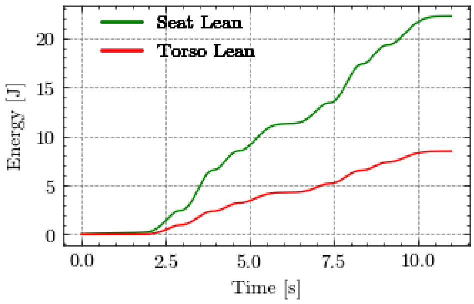
Figure 3.2: Double pendulum lean angle strategies while performing double lane switch at different speeds.

3.1.2. Seat or Torso actuation

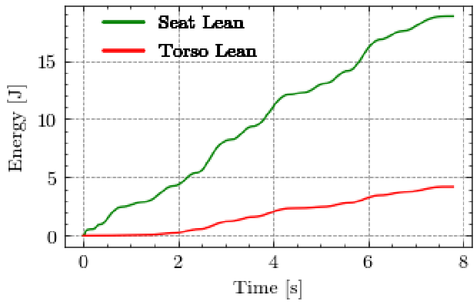
Figure 3.3 shows comparisons of the used energy, lean angles and the exerted torques between the single pendulum models with its PinJoint located at the seat and at the torso. It is immediately noticeable from Figures 3.3a and 3.3b that actuation from the torso requires more energy than when the rider lean is from the seat, and that this difference is larger at 14 km/h compared with 10 km/h. Figures 3.3c and 3.3d illustrate that the lean angle strategy is very similar, however the torso-actuated rider has to reach much higher lean angles to perform the lane switch maneuver. Because the path trajectories are the same for these tasks, and the roll angle of the bicycle is roughly negatively proportional to the rider lean angle, this same leaning strategy is not surprising. Interestingly, Figures 3.3e and 3.3f show that to achieve the same lean angle strategy, an almost opposite torque actuation strategy is used between the Seat and Torso leaning riders. The only virtual difference between the two models is that the mass distribution between the ”bicycle” and the ”rider” is different. The reason for the use of the quotation marks is because the lower part of the torso of the model that is actuated at the TorsoJoint is dynamically part of the bicycle as they are connected via a WeldJoint. Because of this the rider has much less control authority over the bicycle, therefore has to do more control effort and has to employ a different strategy. This extra required control effort due to the different mass distribution and control actuation point, is in some way comparable with doing the task at a higher speed as discussed in subsection 3.1.1, in the sense that a more challenging control task requires a different strategy.

3.1.3. Seat Comparison

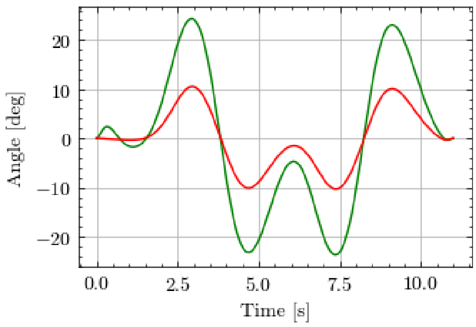
It is clear that from a performance perspective, it is much more effective to control the bicycle at the seat than from leaning from the torso. From the motion capture data described in Figure 1 it was observed that a rider does in fact lean from the torso, and uses combination of rotation and translation of the buttocks to control the bicycle via the seat. The same tasks as described earlier have been performed by the three pendulum models but with the CombiJoint seat connection. The work done to perform the double lane switch at 14 km/h is plotted in Figure 3.4. Here, the models with a CombiJoint seat connections have dotted lines, and it is clear that the models with this seat connection are significantly more efficient in controlling the bicycle than their counterparts with just a PinJoint seat. Figure 3.4 also shows that adding the CombiJoint seat to the Single Pendulum model is more effective compared with the Double- and Triple Pendulum models. Though this difference is not significant, and this relative difference can be due to the Single Pendulum - PinJoint model adopting a slightly different strategy



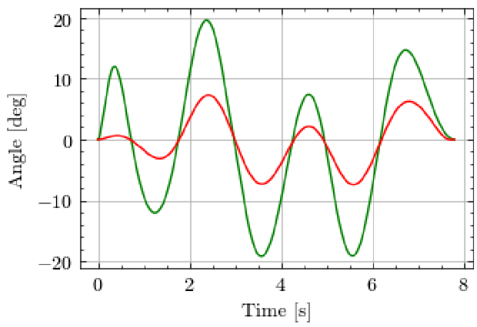
(a) Used energy during Double Lane Switch Task at 10 km/h.



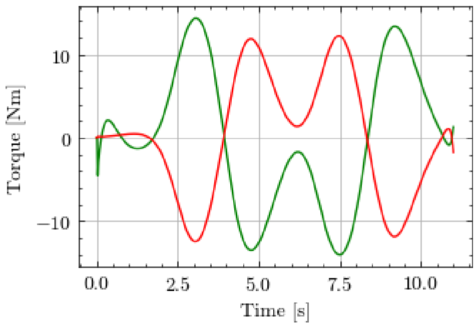
(b) Used energy during Double Lane Switch Task at 14 km/h.



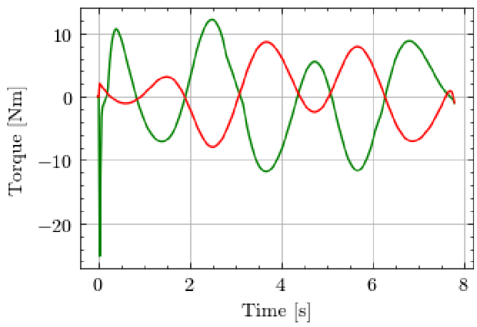
(c) Pendulum lean angle during Double Lane Switch Task at 10 km/h.



(d) Pendulum lean angle during Double Lane Switch Task at 14 km/h.



(e) Exerted torque during Double Lane Switch Task at 10 km/h.



(f) Exerted torque during Double Lane Switch Task at 14 km/h.

Figure 3.3: Energy, Torque and Lean Angle comparisons between Seat Lean and Torso Lean actuation of the Single Pendulum model, on the left: 10 km/h, and on the right: 14 km/h.

where it immediately uses energy to prepare the bicycle for the first turning motion.

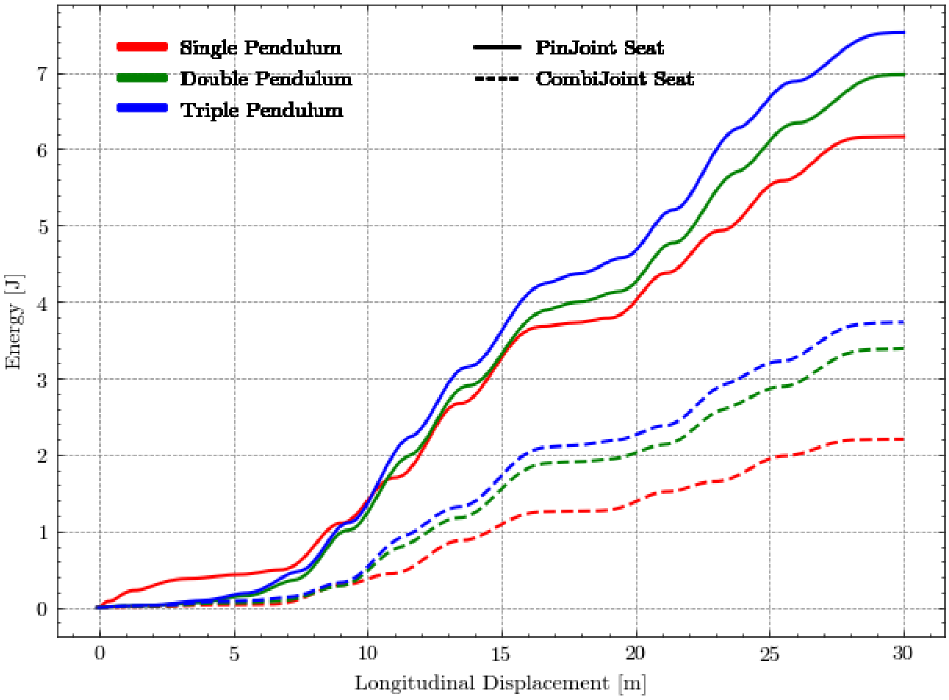
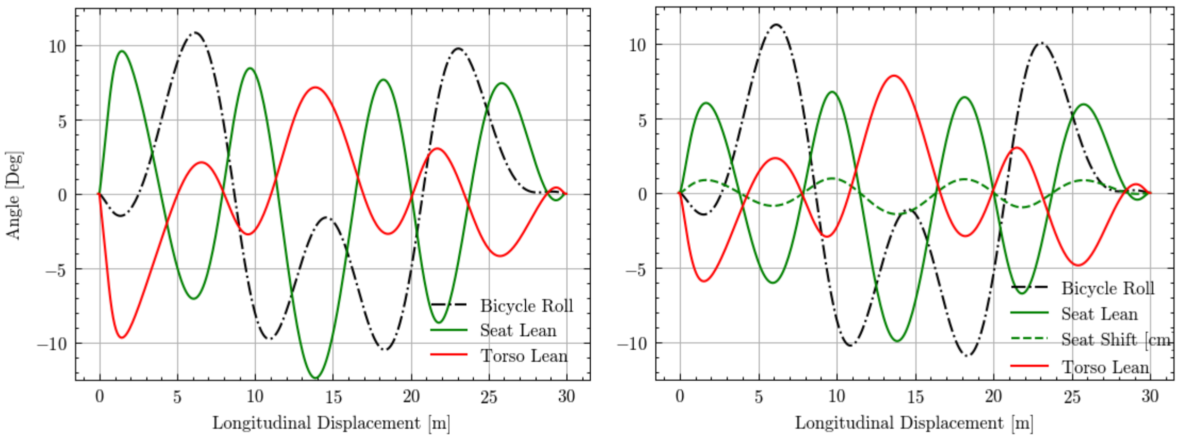


Figure 3.4: Work done by the different models performing a double lane switch at 14 km/h.

Taking a closer look at the control strategies of the Double Pendulum models with the different seat connections in Figure 3.5, it can be noted that there is no real difference in strategy when comparing roll- and lean angles of a certain velocity. The only noticeable difference is that the lean angles of the rider model are larger when using the PinJoint, which implies that this lateral motion of the bicycle between the hips is more effective in transferring a resultant force on the bicycle rear frame than just leaning.



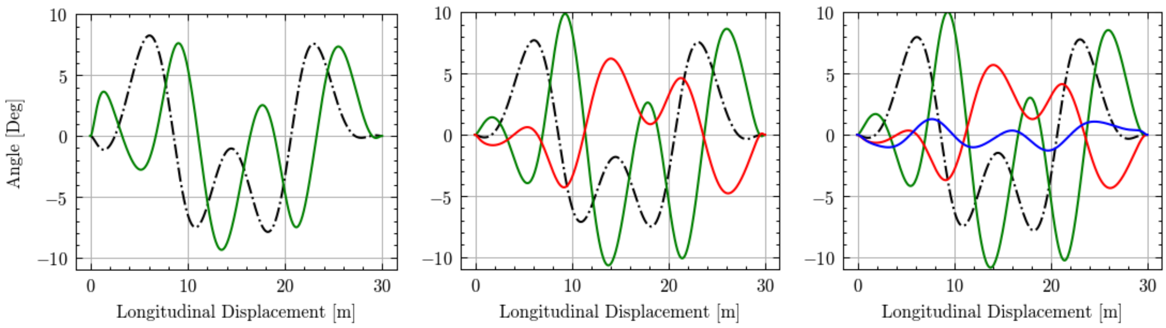
(a) PinJoint seat

(b) CombiJoint seat

Figure 3.5: Model Roll Angles of a: **(a)** PinJoint- and **(b)** CombiJoint seat connection at 16 km/h.

3.1.4. How many Pendulums?

In this section, the difference in control strategies between different models is investigated further. Figure 3.6 presents the roll angles of the Single-, Double- and Triple Pendulum models, with PinJoint seat, next to each other. In the Single Pendulum model, the bicycle roll angle follows the rider roll angle after initiating a counter-steer movement from the start. The roll amplitudes are generally equal between the roll and the rider. The seat lean of the Double Pendulum shows similar behavior, and the bicycle also follows. However here the amplitude of the seat lean can be larger and can thus exert a bigger reaction torque on the bicycle frame. This is because the next link in the pendulum can balance the lower torso pendulum section by rotating in the opposite direction. This is due to the fact that the first link in the multi-stage pendulum is of a significantly lower mass, and will thus achieve a higher rotational speed when the required control torque is exerted on the bicycle. Regarding the Triple Pendulum model, looking at the roll angles of the bicycle, seat and torso joint, there is little difference compared to the Double Pendulum model. The neck moves minimally, and logically has little effect on the rest of the model due to the relatively low mass of the head. The extra complexity of the Double and Triple pendulum is expressed in the higher energy requirement for these models in Figure 3.4 because in such a pendulum model, only the reaction torque on the bicycle frame is what really matters. Figure 3.4 also shows that adding more links to the pendulum decreases the efficiency of controlling the bicycle, and that this is the case for both types of seat joint. However this is not the cases for all speeds at which the tasks is performed. In fact, the Double- and Triple pendulum models can both be more efficient than the other, depending on the speed and the seat joint. An overview of the work done by all models during the Double Lane Switch task is presented in Table 3.1.

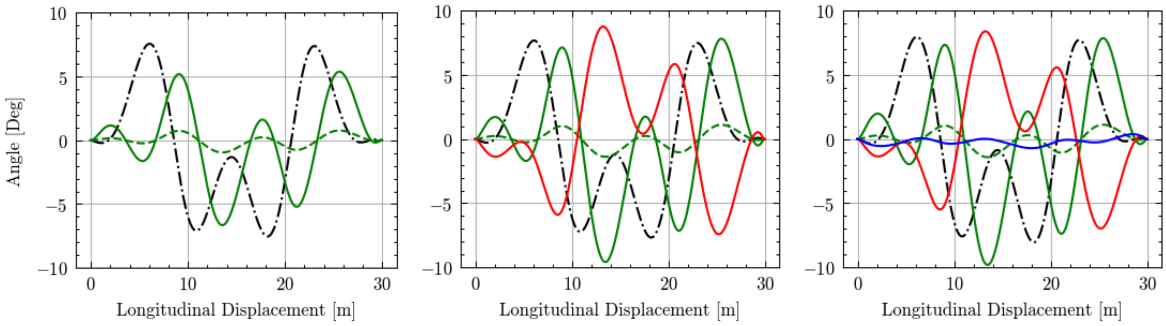


(a) Single Pendulum **(b)** Double Pendulum **(c)** Triple Pendulum

Figure 3.6: Bicycle roll and lean angles of different rider models with PinJoint seat at 14 km/h.
-·-: Bicycle Roll, —: Seat Lean, —: Torso Lean, —: Neck Lean.

The same simulations, though this time with the CombiJoint models, are shown in Figure 3.7. The difference between the three models follow a similar reasoning to that of the previous comparison with the PinJoint seat connection; because of the balancing counter-rotation of the 2^{nd} and 3^{rd} pendulum bodies, the seat connection has to achieve larger angles to get the necessary reaction torque on to the bicycle frame. A difference here is that the upper torso body achieves relatively larger angles than compared with the PinJoint models. This is partly explained by there being a larger shift of mass from the same seat lean angle as the translation of the bodies also has an inertial effect on the system, which means that overall the roll angles are smaller with a CombiJoint seat connection.

Figure 3.4 also shows that adding more links to the pendulum decreases the efficiency of controlling the bicycle, and that this is the case for both types of seat joint. However this is not the cases for all speeds at which the tasks is performed. In fact, the Double- and Triple pendulum models can both be more efficient than the other, depending on the speed. An overview of the work done by all models during the Double Lane Switch task are presented in Table 3.1. Additionally, the same table but with the average torque output instead of work done can be found in Appendix C in Table C.1.



(a) Single Pendulum **(b)** Double Pendulum **(c)** Triple Pendulum

Figure 3.7: Bicycle roll and lean angles of different rider models with CombiJoint seat at 14 km/h.
-·-: Bicycle Roll, —: Seat Lean, - -: Seat Shift [cm], —: Torso Lean, —: Neck Lean.

Models	8 km/h	10 km/h	12 km/h	14 km/h	16 km/h	19.5 km/h
Single Pendulum - Pin Seat	11.75	8.46	7.40	6.16	7.37	13.85
Double Pendulum - Pin Seat	21.39	12.97	11.29	6.97	10.93	19.31
Triple Pendulum - Pin Seat	19.69	12.61	10.48	7.53	10.34	17.92
Single Pendulum - Combi Seat	5.82	4.23	2.96	2.20	2.86	4.96
Double Pendulum - Combi Seat	10.25	6.77	5.00	3.39	4.38	12.70
Triple Pendulum - Combi Seat	9.51	7.56	5.35	3.73	3.69	7.35
Single Pendulum - Torso Joint	28.95	22.21	-	18.79	-	-

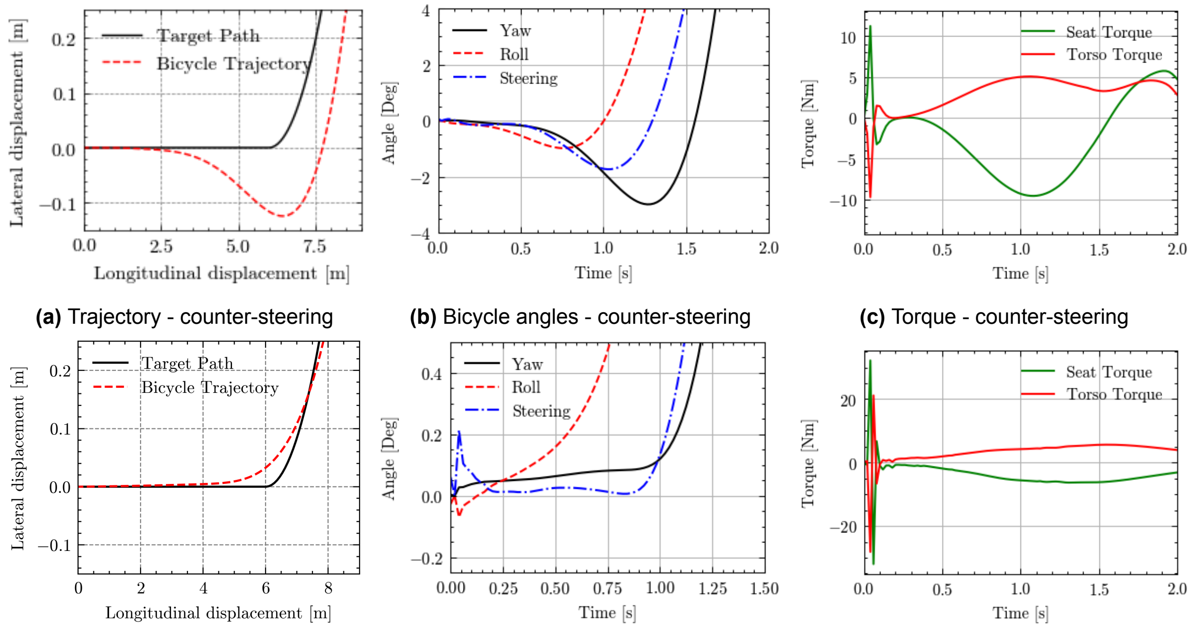
Table 3.1: Overview of the energy used [J] to complete the double lane switch for the different rider models at different speeds.

3.2. The Need for Counter-steering

Figure 3.8 presents the path tracking, bicycle angles and the exerted rider torques during the $90°$ turn task, which is a turn to the left. The results shown are of the Double Pendulum model with PinJoint seat, where the plots on the first row are of the simulation where counter-steering is allowed, while on that of the second row the bicycle is not allowed to steer or yaw to the right. It can be noticed quickly that the bicycle indeed does not steer or yaw to a negative angle. Looking at the data of the simulations, no constraint violations of the bicycle steer angle $q_7(t)$ or yaw angle $q_3(t)$ are present in the turning motion for all three models, with the only exception being a single violation of $q_7 = -5.913e^{-9}$ rad while the bicycle straightens itself at the end. Figure C.1 shows the exerted torque(s), bicycle orientation angles and the path trajectories of all three used models in this task, both with and without counter-steering.

When looking at the trajectory of the bicycle, it seems like that of the no-counter-steer simulation is much smoother. In fact the simulations with no counter-steering have significantly better optimization performance than their counter-steering counterparts. This can be seen in Table 3.2 where the used energy and the RMS tracking error is displayed per simulation. This shows that, for these optimizations with the specific cost-function, not counter-steering is a better control strategy than making turns with counter-steering. Of course, using counter-steering is a much more responsive way of controlling the bicycle, and gives overall better control, though it is interesting that the optimizer could not find this solution without having to be constrained into it.

Let's take a closer look at how the rider model can control the bicycle without counter-steering in Figure 3.9. The Single Pendulum model seems to very carefully lean into the corner, which causes a very



(a) Trajectory - counter-steering **(b)** Bicycle angles - counter-steering **(c)** Torque - counter-steering

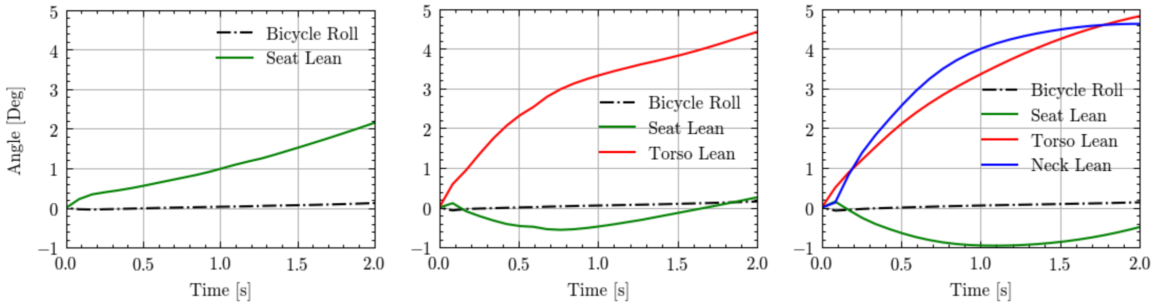
(d) Trajectory - no counter-steering **(e)** Bicycle angles - no counter-steering **(f)** Torque - no counter-steering

Figure 3.8: A comparison between the path following, torque and bicycle angles during a $90°$ turn with a Double Pendulum rider model, while counter-steering and not.

Pendulum Model	Counter-Steering	Energy [J]	RMS [m]
Single	Yes	4.50	0.149
Double	Yes	8.37	0.147
Triple	Yes	7.48	0.146
Single	No	2.51	0.123
Double	No	3.73	0.127
Triple	No	4.63	0.131

Table 3.2: Performance metrics of different pendulum models with PinJoint seat during 90° turn task.

small roll of the bicycle out of the corner. However this negative roll lasts a very short time, less than 0.2 seconds in fact, with a maximum roll angle of 0.04°. Due to the careful and smooth lean of the rider the bicycle frame slowly follows and initiates a small roll into the corner, until the point that the majority of the turning is set in motion. The model lean angles in Figures 3.9b and 3.9c demonstrate a different strategy. By leaning the hips and lower part of the torso out of the corner, the bicycle rolls into the corner. Because the torso joint -and neck joint for the Triple Pendulum model- lean into the corner, at a greater angle than the seat lean, the CoG of the rider is also on the inside of the corner relatively to the bicycle frame. This slowly causes the bicycle rear frame to also lean into the corner until the front frame follows and the corner is made.

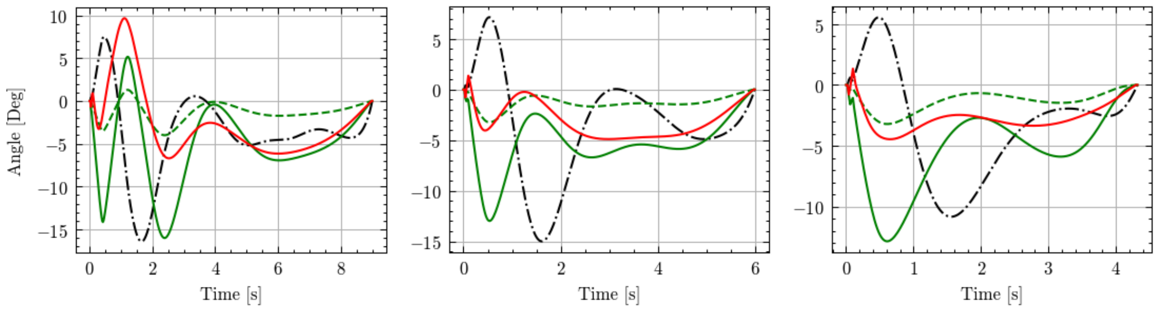


(a) Single Pendulum **(b)** Double Pendulum **(c)** Triple Pendulum

Figure 3.9: Bicycle roll angle and rider lean angles during the initial phase of a $90°$ without counter steering.

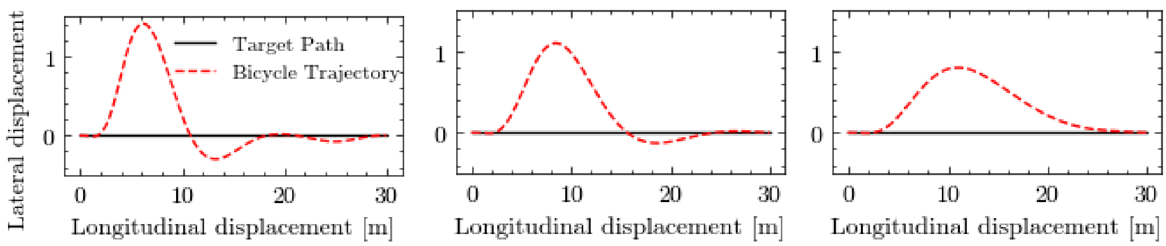
3.3. Riding Through the Wind

The Figures in 3.10 show the lean angles of a Double Pendulum model with CombiJoint in response to the wind perturbation force at 12 km/h(3.10a), 18 km/h(3.10b) and 25 km/h (3.10c). While cycling at higher speeds, the wind perturbation leads to less high roll angles of the bicycle. As can be seen for all three speeds, the rider quickly leans into the wind to counter the strong perturbation incurred by the wind, and when the bicycle is back on the path, the rider has to keep leaning into the wind to counter the wind force. The maximum deviation of rear wheel contact point P from the path, which are 1.41, 1.10 and 0.80 meter for the 12, 18 and 25 km/h respectively, is noticeably lower at higher speeds. The trajectories of the three runs are displayed in Figures 3.11a, 3.11b and 3.11c. From the trajectory plots, it can be noted that it takes the bicycle-rider model approximately 4 seconds to go back to the path. For a better visualization of the leaning strategy by the rider, a timelapse of the optimization can be viewed in Figure C.5.



(a) 12 km/h **(b)** 18 km/h **(c)** 25 km/h

Figure 3.10: Bicycle roll angle and rider lean angles when cycling with a side wind at three different speeds. -·-: Bicycle Roll, —: Seat Lean, - -: Seat Shift [cm], —: Torso Lean



(a) 12 km/h **(b)** 18 km/h **(c)** 25 km/h

Figure 3.11: Bicycle-rider trajectories in response to wind perturbation at three different speeds.

3.4. Sprung Steering Stabilizer

[h!] This section discusses the influence of attaching a spring between the front- and rear frame. Firstly, results are presented for the double lane switch task by the Triple Pendulum CombiJoint model. Figure 3.12 plots the required energy to perform the task with a tracking error of RMS = 0.20 m, for different spring rates. The data clearly follows a parabolic curve, but the minimum of this parabola is not located at k=0 N/rad. In fact, the lowest effort was simulated to be with a stiffness of k = -1 N/rad. An explanation for this initially counterintuitive result can be that the steering of the bicycle becomes a bit more responsive which helps in performing a lane switching maneuver. Too much negative spring rate would make it more difficult to straighten the bicycle, and k = -1 N/rad appears to be the right trade-off for this simulation. The model configuration used for this task is the Triple Pendulum rider with CombiSeat.

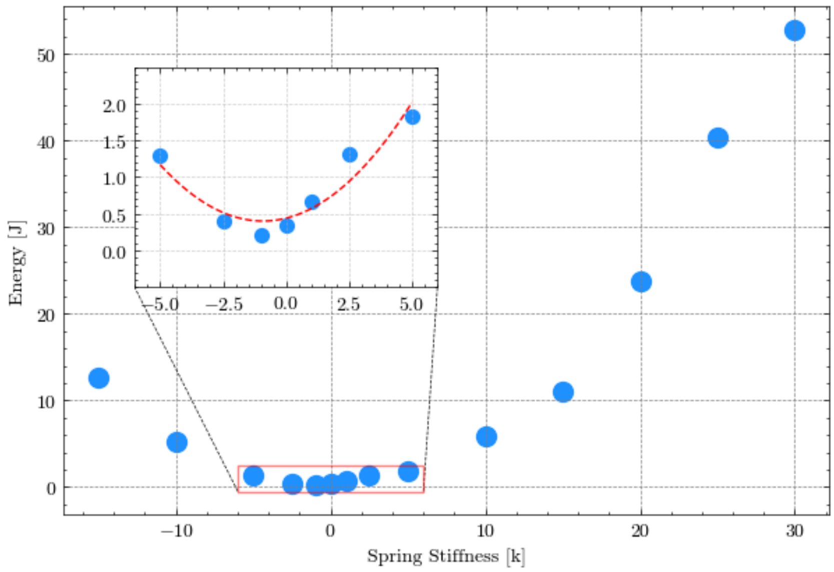


Figure 3.12: Required energy to perform Double lane switch task for different front frame spring rates

While it appears to be useful to have a negatively sprung spring between the front and rear frame when performing a turning maneuver, there could be situations where the opposite is true, like when the rider aims to keep going straight while riding in heavy wind. Figure 3.13 plots the used energy against the RMS tracking error for riding through the wind like the previous section. The graphs are colored for riding without a spring, with a spring of k = 10 N/rad and a spring of k =20 N/rad, for 5 different speeds which have their own marker. Here, a result being in the bottom-left would mean that it is easier to stay in a straight line than when a result is in the upper-right of the graph. A general trend in these results is that adding a spring seems to make it easier to counter the perturbations. This trend is not really consistent however, as the 12 km/h run with a spring stiffness of k = 10 N/rad has better performance than the other runs at the same speed.

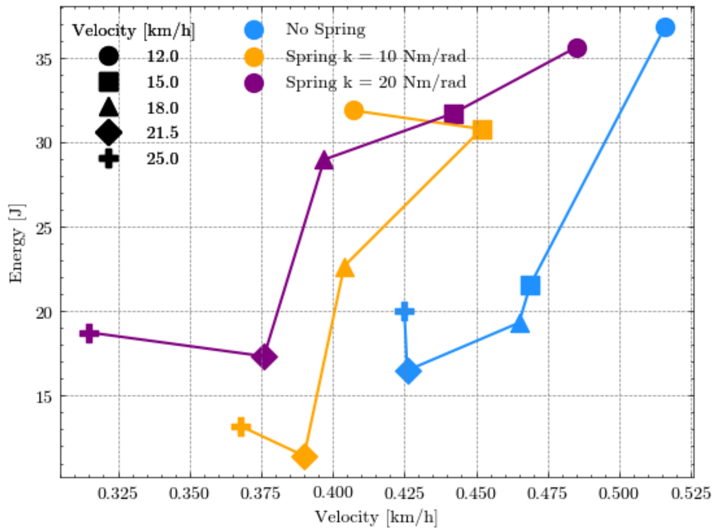


Figure 3.13: Tracking performance and required energy for different velocities and different front frame spring rates.

4 Discussion

The results in the previous chapter present comparisons between different upper body rider models that are inspired by literature, motion capture data and personal experience. These models are used to investigate steering mechanics, and it present evidence that a bicycle-rider can make a turn without having to counter-steer. Furthermore the rider control strategy to handle strong wind perturbations are analyzed along with the effect of a sprung front frame in two different situations

Results Discussion

The performance of the models presented in section 3.1 has to be viewed in the context of already existing models to judge the scientific accuracy of this research. Previously, lateral body control on the bicycle has been simulated by a single pendulum leaning rider. The results of this report imply that increasing the amount of pendulum joints does not result in more efficient, or better control over the bicycle when riding with no hands. This is because the only two inputs on the bicycle that really matter are the reaction torque of rider on the bicycle frame, and the location of the CoG of the system. The multi-pendulum models with the leaning seat are neither much more physically accurate models, or more representative of real-world behavior. The only motion identified from the motion capture data in Figure 1.6 that can be replicated by the solely leaning models is the rider lean (e) which would be mimicked by the Single Pendulum Torso Lean model. However this model performed significantly worse than the others, which is not surprising due to the lower relative rider mass. Though, the magnitude of difference between the Single Pendulum actuation from the seat joint or the torso joint was quite surprising, and this illustrates the impact the mass of the rider has on the dynamics of the bicycle.

The addition of the CombiJoint seat however does make the models, in particular the double- and triple pendulum models, more representative of human behavior and this is reflected in the in the effectiveness of the control. Moving the saddle of the bicycle between the buttocks and thighs is a much more responsive control method and can have a bigger reaction force on the bicycle than just leaning. As Moore notes, these movements are mostly actuated by the legs via the pedals of the bicycle [5]. It would be really interesting to see what would happen if legs are added to the active rider model. Combining the legs with this CombiJoint, and letting the optimizer find an optimal solution could potentially expose other ways to control the bicycle, and give more insight into how the legs and the upper body cooperate while riding a bicycle.

From Figures 3.6 and 3.7 can be concluded that control input of the head on the lateral dynamics is negligible, as it hardly moves and the angle traces of the seat- and torso joint are virtually equal between the double- and triple pendulum. This is not surprising as the weight of the head is small with respect to the rest of the system, and the torques of the neck will have less effect on the bicycle frame than the other joint torques. So in terms of control optimization, there is little point in adding this joint in the mode. However in terms of creating a rider model that can replicate the way a human controls a bicycle it would add value, because stabilization of the head in space, as described in the Introduction, happens constantly during balancing tasks.

When cycling at higher speeds the effort required to make such a maneuver increases significantly, likely because the turns it has to make are too sharp at those speeds. An explanation for why the lower speeds also require slightly more effort to control than at 14 km/h, while performing a more gentle turning motion can be because these speeds are below the weave speed of the bicycle-rider. The stable region of a rigid-rider model with the parameters used in this thesis is between 5.15 m/s and 7.70 m/s, or between 18.54 km/h and 27.72 km/h. Figure 1.3 is actually the stability plot of the used bicycle-rider model, though with a rigid rider, and not connected via pin joints. While 14 km/h and 16 km/h are also below the weave speed, the rider in the optimizations not being rigid changes the stability of the system, and the lower the travel speed, the more unstable the weave mode gets and thus this is expected to require more effort to control. The use of the same bicycle and rider parameters of the bicycle and rider that were subject in the motion capture identification research [5] also brings its own questions. This continuity of using the same parameters enables the results from the simulations to be compared to the motion capture data, however it does have a risk of drawing conclusions on bicycle-rider behavior from a subject who might have a very unique style of riding a bicycle, which would make the results of this research less relevant and applicable in real life. With more time, I would have loved to have my own parameters used, and also do my own motion capture data acquisition.

When judging the results of adding a spring between the front- and rear frame of the bicycle, the results on the practicality can be disputed. As such a spring is essentially just a steering actuator which, depending on the sign of stiffness k, can help making a turn improve straight-line stability of the bicycle. In essence such a spring is a steering actuator, that can be amplifying or nullifying the self-stabilizing effects of the self-aligning torque while turning or the steering torque caused by the CoM position of the front-frame. So for specific use cases such a spring can be beneficial for the handling characteristics of the bicycle, though it would be easier to just use the handlebars to steer the bicycle with the hands when riding in windy conditions, or when making quick turning maneuvers.

As described in 2.2, sometimes the optimizer does not find the optimal solution to the problem, or the global minimum of the cost function. While this issue has been tried to be mitigated by providing multiple initial guesses for simulations, it cannot be said for sure that the optimal solution is always found. Results where outcomes of simulations do not follow the trend of other solutions, like the Torso Torque actuation of the 14 km/h double lane switch optimization in **??**, or the 12 km/h wind riding optimization with a $10\ Nm/rad$ spring result. I think this could be solved by letting the optimizer run a very large amount of simulations to eradicate this risk, though that requires significantly more computing time and power.
It was surprising to see that the no-counter-steering optimizations had better performance than their counter-steering counterparts. If not counter-steering is more optimal control than the optimizer should find such a method also without the constraints on the model that prevents counter-steering, thus the conclusion has to be that the optimizer did not find the global minimum of the cost function. The explanation that I have for this is that without the constraints, the optimizer finds that counter-steering is the "easiest" way to even make the turn, as the initial and final state constraints are the most strict conditions to comply with. Just like when humans learn to ride a bicycle, we often unconsciously find that counter-steering is the best way to make a turn. But once the optimizer has gone down the path of counter-steering, it is likely difficult to find another strategy that complies with the constraints and thus the optimizer will just focus more on optimizing the counter-steering turning movement. Once the optimizer is then close to the local minimum of this solution, iterations to find other minima are not able to find a potentially better strategy. This can likely be solved by changing the parameters of the optimizer and forcing the optimizer to look for more minima. Also control techniques like tunneling algorithms that aim to find solutions that are on the same "height" of the cost function as a found local minimum might help.

Because the simulations in this paper are solving an optimal control problem, the model can find solution that a human could not, as it "knows" in advance what will happen and can adjust its control strategies on that. Moreover, because there are no biomechanical features like muscle and tendon damping and stiffness, the force-length or force-velocity relationship of the muscles are included in the model, it is difficult to draw conclusions from the model regarding used joint torques and energy. This is especially evident for the no-counter-steering simulations, as the turning motion seems to be initiated by some very explosive and sudden torque inputs to achieve the required weight distribution without counter-steering. As this shows that it is possible to turn without counter-steering, it is still a question if this is possible in the real world by a human, on a real bicycle. Also, features like variable joint stiffness, and/or nonlinear joint stiffness that increases exponentially when it reaches extreme stretch- and flex angles could sketch a more complete overview of how muscle co-contraction and tendon stiffness can be used to achieve stability during cycling. However the optimal control strategies presented in this theses do give insights on what is objectively the best way to control the bicycle, which is the general research question of this thesis, for this specific bicycle-rider model which naturally has some limitations.

In my opinion there is a balance to be struck between finding the optimal control solutions of a model, and making the model more biomechanically resemblant to human behavior. For this thesis it has been decided to focus more on finding the optimal control for the given model, as this can be more objectively assessed. Therefore, the upcoming summary of limitations of the model and the optimization will not be discussing all the details that could make the model behave more like an actual human, but will be discussing improvements on the optimal control problem possibilities and model improvements.

Implications

To put the results of this thesis into perspective, it can be seen as containing valuable additions to the knowledge of bicycle control by humans and bicycle handling dynamics. It provides more insight into the multi-pendulum-like behavior and seat mechanics that are, mostly subconsciously, employed while riding a bicycle both with and without hands on the handlebars. The novel proof that it is possible to make a turn without turning the handlebars in the opposite direction of the turn, invites for a more detailed analysis on this process, and if there are other, different solutions to this problem. The addition of a spring on the front frame gives more insights into which geometric features of the front frame and its impact on riding dynamics and their functionality in different situations.

Limitations of the Model and Optimization

As with any simulation model, there are shortcomings and limitations to this research that, with the right additions and improvements, can result in better and more conclusive results. As discussed, the legs play an important roll in the way loads are transmitted from the rider to the bicycle, and are also used during low-speed balance control on the bicycle. Adding legs would give the model more freedom in finding optimal control solutions and a more complete comparison between the different control strategies available to the rider. The same can be said for adding a spherical torso joint. This would allow for rider twist, which I expect allows the bicycle to be controlled in a different way than by just body lean, as it would exert a torque round the yaw axis of the bicycle. I would like to note that legs and a spherical torso joint have been integrated in the code, however have not been used to get results due to some persistent bugs and a significantly more complex model to solve for the optimizer and thus very long simulation times.

Regarding the need for counter-steering, I think the research can be greatly improved. Firstly, it would be valuable to see if a rider with the CombiJoint seat is also able to perform the maneuver, as I think this model is more resembling of how a human rider a bicycle, and this joint mechanism can control the roll of the bicycle frame better. Regarding the optimization task of the 90° turn, it would be better to leave the path tracking part out the cost-function. If there are only initial and final state constraints given, combined with the adequate bounds to prohibit counter-steering, the optimizer will have more freedom to find a solution that satisfies the conditions, and perhaps this will result in better and more realistic results.
The simulations of the perturbations on the bicycle due to wind can also be improved, as in this thesis the wind is simulated by a force on a single point on the bicycle frame. While the location on the bicycle is a good point for the resulting force on the bicycle as a whole, it would be better if there was also a force pointed at the center of the front wheel. This way the wind exerts a moment around the steering axis and influences the handling of the bicycle more, which would likely require a different control strategy of the rider. The bicycle model could further be improved by adding a more complex tire model, or adding frame flexibility, which influences the self-stabilizing behavior of the bicycle.

While the optimizer has minimization of used torques as it's objective, the performance of the model is partly judged on the used energy in it's simulation. While exerted torques and used energy are linked to each other, it would be better to let the optimizer solve for the work done by the model to make the performance assessment more resembling of the objective of the task.

5

Conclusion

This thesis intended to delve into the dynamics of controlling a bicycle without using the handle bars. Research into the vast existing knowledge on bicycle dynamics and human balance control mechanisms, resulted in a basis from which interesting questions in the world of bicycles could be answered, fueled by an underlying interest in cycling without hands due to personal affinity with this activity. The general research question for this thesis is therefore ”How to cycle without Hands” which is investigated by letting multiple hand-less rider models perform tasks and asses their performance and behavior. This model is built in the software package BRiM, which is a multi-body dynamics python package utilizing symbolic computation program Sympy, to create Bicycle-Rider models. With the Bicycle-Rider models built, several optimal control problem tasks are executing using the program opty, a python package to solve such control programs. To answer the general research question, seven different upper-body models of a rider are created and tasked to perform a lane switching maneuver at different velocities. Inspiration for these upper-body models comes from motion-capture data on rider motions, both with hands on the handle bars and without. This data is analyzed and together with literature it has inspired several ideas on how to create the models. Besides optimal hands-free bicycle control strategies, the models of this research are used to investigate questions that are not fully understood yet. This paper presents novel proof that a bicycle *can* make a turn without having to initially counter-steer. This has, to my knowledge, not been demonstrated before with the use of a tested an accepted adaptation of the Carvallo-Whipple bicycle model. While the model in this thesis is not a fully realistic rider model and the optimal control problem can find solutions that a human might not be able to recreate, the results of this research invite for more thorough research in this phenomenon with a more extended rider model and more strict constraints. Furthermore, the effect of having a spring between the front and rear frames of the bicycle on both handling and stability properties of the bicycle is analyzed and concluded is that a spring does increase the stability and resistance to perturbations. Though this comes at the cast of handling performance, where a negative spring rate would have the opposite effect. So depending on the specific use case of the bicycle, a spring may or may not improve the control characteristics. Lastly, the model sheds some light on how a bicycle is best controlled when riding in windy conditions when no hands can be used.
While new insights have been gained, this thesis provides a basis for more extensive research to be done in bicycle control behavior by the rider and control techniques.

Future Work

For future research on this topic, there are a number of improvements that can be made regarding this thesis. The rider model can be extended by adding legs, which can give insights in what effect pedaling has on the stability of the bicycle. Also, this can explore the effectiveness of lateral stability control by the legs, both in terms of inertial control and exerting loads from the hips to the bicycle frame. The rider model can be improved by including a pin joint for the torso joint which would enable body twist to also assist in controlling the bicycle without hands. Moreover, adding realistic nonlinear joint stiffness and damping could give a better insight in the energy consumption of controlling a bicycle with body-motions. The cost function can be altered to include the work done by the model instead of just torques, and for the counter-steer task the path following part can also be left out so that the optimizer solely focuses on optimizing the steering behavior within the system bounds.

References

[1] Francis JW Whipple. “The stability of the motion of a bicycle”. In: *Quarterly Journal of Pure and Applied Mathematics* 30.120 (1899), pp. 312–348.

[2] Emmanuel Carvallo. *Théorie du mouvement du monocycle et de la bicyclette...* Gauthier-Villars, 1901.

[3] Arend L Schwab, Jaap P Meijaard, and Jim M Papadopoulos. “Benchmark results on the linearized equations of motion of an uncontrolled bicycle”. In: *Journal of mechanical science and technology* 19 (2005), pp. 292–304.

[4] Jaap P Meijaard et al. “Linearized dynamics equations for the balance and steer of a bicycle: a benchmark and review”. In: *Proceedings of the Royal society A: mathematical, physical and engineering sciences* 463.2084 (2007), pp. 1955–1982.

[5] Jason Keith Moore. *Human control of a bicycle*. University of California, Davis, 2012.

[6] Creative Commons BY 3.0.

[7] JP Meijaard et al. “History of thoughts about bicycle self-stability”. In: *Science Magazine* (2011).

[8] Milan Paudel and Fook Fah Yap. “Development of an improved design methodology and front steering design guideline for small-wheel bicycles for better stability and performance”. In: *Proceedings of the Institution of Mechanical Engineers, Part P: Journal of Sports Engineering and Technology* 234.3 (2020), pp. 227–244.

[9] Jodi DG Kooijman et al. “A bicycle can be self-stable without gyroscopic or caster effects”. In: *science* 332.6027 (2011), pp. 339–342.

[10] David EH Jones. “The stability of the bicycle”. In: *Physics today* 23.4 (1970), pp. 34–40.

[11] Vera E Bulsink et al. “The effect of tyre and rider properties on the stability of a bicycle”. In: *Advances in mechanical engineering* 7.12 (2015), p. 1687814015622596.

[12] Arend L Schwab, Jaap P Meijaard, and Jodi DG Kooijman. “Lateral dynamics of a bicycle with a passive rider model: stability and controllability”. In: *Vehicle system dynamics* 50.8 (2012), pp. 1209–1224.

[13] AL Schwab and JDG Kooijman. “Lateral dynamics of a bicycle with passive rider model”. In: *The 1st Joint International Conference on Multibody System Dynamics*. 2010.

[14] R Douglas Roland and James P Lynch. *Bicycle dynamics: Tire characteristics and rider modeling*. Cornell Aeronautical Laboratory, 1972.

[15] Mouad Garziad and Abdelmjid Saka. “Influence of rider on the stability and control of two wheeled vehicles”. In: *Journal Européen des Systèmes Automatisés* 52.5 (2019), pp. 515–520.

[16] Pengcheng Wang, Jingang Yi, and Tao Liu. “Stability and control of a rider–bicycle system: Analysis and experiments”. In: *IEEE Transactions on Automation Science and Engineering* 17.1 (2019), pp. 348–360.

[17] Stephen M Cain, James A Ashton-Miller, and Noel C Perkins. “On the skill of balancing while riding a bicycle”. In: *PLoS one* 11.2 (2016), e0149340.

[18] Jason K Moore et al. “Rider motion identification during normal bicycling by means of principal component analysis”. In: *Multibody System Dynamics* 25 (2011), pp. 225–244.

[19] Joel Fajans. “Steering in bicycles and motorcycles”. In: *American Journal of Physics* 68.7 (2000), pp. 654–659.

[20] D.J.N. Limebeer and R.S. Sharp. “Bicycles, motorcycles, and models”. In: *IEEE Control Systems Magazine* 26.5 (2006), pp. 34–61. DOI: `10.1109/MCS.2006.1700044`.

[21] Karl J Astrom, Richard E Klein, and Anders Lennartsson. “Bicycle dynamics and control: adapted bicycles for education and research”. In: *IEEE Control Systems Magazine* 25.4 (2005), pp. 26–47.

[22] PJ Prince and Ahmed Al-Jumaily. “Bicycle steering and roll responses”. In: *Proceedings of the Institution of Mechanical Engineers, Part K: Journal of Multi-body Dynamics* 226.2 (2012), pp. 95–107.

[23] Robin S Sharp. “On the stability and control of the bicycle”. In: *Applied Mechanics Reviews* (2008).

[24] David H Weir. “Motorcycle Dynamics and Rider Control”. In: *Vehicle System Dynamics* 6.2-3 (1977), pp. 187–190.

[25] Mark Morningstar et al. “Reflex control of the spine and posture: A review of the literature from a chiropractic perspective”. In: *Chiropractic osteopathy* 13 (Sept. 2005), p. 16. DOI: `10.1186/1746-1340-13-16`.

[26] John J Buchanan and Fay B Horak. “Emergence of postural patterns as a function of vision and translation frequency”. In: *Journal of Neurophysiology* 81.5 (1999), pp. 2325–2339.

[27] Jaap H van Dieën, Idsart Kingma, and JCE Van Der Bug. “Evidence for a role of antagonistic cocontraction in controlling trunk stiffness during lifting”. In: *Journal of biomechanics* 36.12 (2003), pp. 1829–1836.

[28] Kevin P Granata and Karl F Orishimo. “Response of trunk muscle coactivation to changes in spinal stability”. In: *Journal of biomechanics* 34.9 (2001), pp. 1117–1123.

[29] Alice Caimi, Stephen J Ferguson, and Dominika Ignasiak. “Evaluation of trunk muscle coactivation predictions in multi-body models”. In: *Journal of Biomechanics* 168 (2024), p. 112039.

[30] Marcel Dreischarf et al. “Estimation of loads on human lumbar spine: A review of in vivo and computational model studies”. In: *Journal of biomechanics* 49.6 (2016), pp. 833–845.

[31] Glen M Blenkinsop, Matthew TG Pain, and Michael J Hiley. “Balance control strategies during perturbed and unperturbed balance in standing and handstand”. In: *Royal Society open science* 4.7 (2017), p. 161018.

[32] David A Winter. “Human balance and posture control during standing and walking”. In: *Gait & posture* 3.4 (1995), pp. 193–214.

[33] Shirley Rietdyk et al. “Balance recovery from medio-lateral perturbations of the upper body during standing”. In: *Journal of biomechanics* 32.11 (1999), pp. 1149–1158.

[34] David A Winter et al. “Unified theory regarding A/P and M/L balance in quiet stance”. In: *Journal of neurophysiology* 75.6 (1996), pp. 2334–2343.

[35] AL Hof and Jaak Duysens. “Responses of human hip abductor muscles to lateral balance perturbations during walking”. In: *Experimental brain research* 230 (2013), pp. 301–310.

[36] Pilar Serra-AÑó et al. “Postural control mechanisms in healthy adults in sitting and standing positions”. In: *Perceptual and motor skills* 121.1 (2015), pp. 119–134.

[37] Timótheüs J Stienstra, Samuel G Brockie, and Jason K Moore. “BRiM: A Modular Bicycle-Rider Modeling Framework”. In: *The Evolving Scholar-BMD* (2023).

[38] Jason K. Moore Simon J. Sorgedrager. *No-Hands Riding*. Accessed: 2025-04-17. 2024. URL: `https://github.com/mechmotum/no-hands-riding`.

[39] Jason K Moore et al. “Accurate measurement of bicycle parameters”. In: *Proceedings, Bicycle and Motorcycle Dynamics 2010 Symposium on the Dynamics and Control of Single Track Vehicles, 20Ű22 October 2010, Delft, The Netherlands*. 2010.

[40] Christopher Dembia, Jason K Moore, and Mont Hubbard. “An object oriented implementation of the Yeadon human inertia model”. In: *F1000Research* 3 (2015), p. 223.

[41] Jason K. Moore. *BicycleParameters*. `https://github.com/moorepants/BicycleParameters`. Version 1.2.0. 2024.

[42] Jason K Moore and Antonie J van den Bogert. “opty: Software for trajectory optimization and parameter identification using direct collocation”. In: *Journal of Open Source Software* (2018).

[43] DM Fintelman et al. “The effect of crosswinds on cyclists: an experimental study”. In: *Procedia Engineering* 72 (2014), pp. 720–725.

[44] Aaron Meurer et al. “SymPy: symbolic computing in Python”. In: *PeerJ Computer Science* 3 (Jan. 2017), e103. ISSN: 2376-5992. DOI: `10.7717/peerj-cs.103`. URL: `https://doi.org/10.7717/peerj-cs.103`.

A
BRiM description

The simulations in this study are conducted using BRiM [37], a specialized program designed to model and analyze bicycle dynamics. BRiM is built with flexibility and accuracy in mind, incorporating advanced computational techniques to simulate the intricate interactions between the bicycle and its rider. This section provides an overview of the program's structure and functionality, detailing its key components and how they contribute to accurate modeling.

BRiM is a Python software package that enables the creation of bicycle and bicycle-rider models using symbolic computation of multibody dynamic systems using SymPy [44]. In BRiM there are three types of components with which systems can be designed. The first type is a ”model”, which defines a system within defined system boundaries. A model can consist of different submodels as long as these are well defined within the boundaries. These submodels are children of their parent model. Models can be connected to each other by a ”connection”. These connections are joint types from the *Sympy.mechanics* library, and in BRiM these joint types would be a pin joint, spherical ball joint, or a weld joint, though a contact constraint is also possible. These connections require a point defined on the coupled models so that interactions between the models can take place. The third type of component within BRiM is a *load group*. Such load groups are either torques or forces, and be placed on a connection between models like a joint torque, or as a force on models like gravity. Loadgroups can be defined as a constant force or a time-varying function. This construction of BRiM allows multi-body systems to be created with different models and submodels interacting with each other. Submodels can be optional or mandatory for a parent model to function. This allows for a modular design where different configurations of models can be tested without having to introduce new parent models.

BRiM models the bicycle, like the Carvallo-Whipple model, as 4 rigid bodies consisting of the front- and rear frame, with two wheels free to rotate within their respective frame. To make computation of the EoM's possible, the wheels are constrained to touch the ground so that the loop-requirement is satisfied. This contact between the tire and the ground can be modeled as a knife-edge contact, but it is also possible to implement a more complex tire model like the TU Delft Magic tire to have a more realistic interaction between bicycle and ground. As the name suggests, BRiM is built to implement a rider on the bicycle model. To achieve this, a Rider model can be connected to the Bicycle model via one or more connections. The Rider model can consist of a single child model, for example a lower body that is connected to the bicycle via a seat connection. The modularity of BRiM allows for different configurations of riders to be made. In the core of BRiM, three connections between de Bicycle and Rider exist, the seat between bicycle saddle and buttocks, the pedals between bicycle crank and rider feet, and the handgrips between the handle bars and hands. The latter is for obvious reasons not considered in this research.

B

List of Parameters

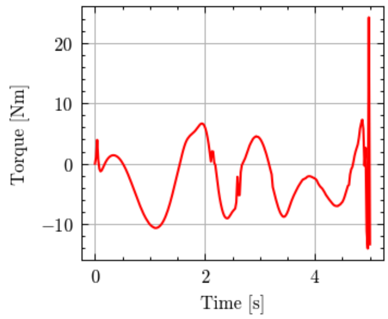
Parameter	Value	Unit
Browser		
IBxx	0.52962890621	kgm^2
IBxz	-0.116285607878	kgm^2
IByy	1.3163960125	kgm^2
IBzz	0.756786895402	kgm^2
IFxx	0.0883826870796	kgm^2
IFyy	0.149221207336	kgm^2
IHxx	0.25335539588	kgm^2
IHxz	-0.0720217263293	kgm^2
IHyy	0.245827908036	kgm^2
IHzz	0.0955686343473	kgm^2
IRxx	0.0904114316323	kgm^2
IRyy	0.152391250767	kgm^2
c	0.0685808540382	m
g	9.81	m/s^2
lam	0.399680398707	m
mB	9.86	kg
mF	2.02	kg
mH	3.22	kg
mR	3.11	kg
rF	0.34352982332	m
rR	0.340958858855	m
w	1.121	m
xB	0.275951285677	m
xH	0.866949640247	m
zB	-0.537842424305	m
zH	-0.748236400835	m
Rider - Jason		
IBxx	9.57707087143	kgm^2
IBxz	-1.02983731033	kgm^2
IByy	9.71438946103	kgm^2

IBzz	2.16361764694	kgm^2
mB	83.5	kg
xB	0.352471843075	m
yB	2.77335755173e-08	m
zB	-1.10674939861	m
Ls0w	0.347	m
Ls2L	0.277	m
Ls4w	0.343	m
Ls5L	0.545	m
Ls6L	0.530	m
Ls8L	0.308	m
Ls7p	0.208	m
Joints		
k_{seat}	0	N/rad
c_{seat}	10	N/(rad/s)
k_{torso}	0	N/rad
c_{torso}	10	N/(rad/s)
k_{neck}	0	N/rad
c_{neck}	10	N/(rad/s)

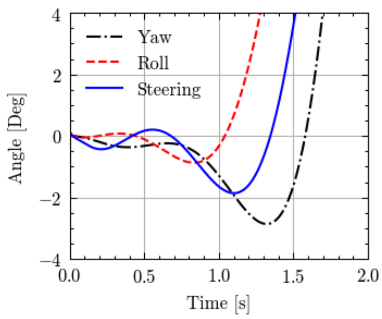
C
Additional Figures

Models	8 km/h	10 km/h	12 km/h	14 km/h	16 km/h	19.5 km/h
Single Pendulum - Pin Seat	6.65	5.63	5.50	4.69	4.99	6.89
Double Pendulum - Pin Seat	7.82	6.63	6.49	5.06	6.46	8.71
Triple Pendulum - Pin Seat	8.07	6.99	6.76	5.82	7.04	9.51
Single Pendulum - Combi Seat	4.59	3.79	3.18	2.34	2.49	3.52
Double Pendulum - Combi Seat	4.49	3.88	3.35	2.63	3.02	9.45
Triple Pendulum - Combi Seat	4.60	4.15	3.60	2.97	3.10	7.84
Single Pendulum - Torso Joint	7.82	6.74	-	5.94	-	-

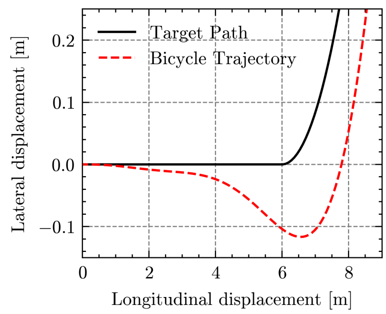
Table C.1: Overview of the average torque [Nm] required to complete the double lane switch for the different rider models at different speeds.



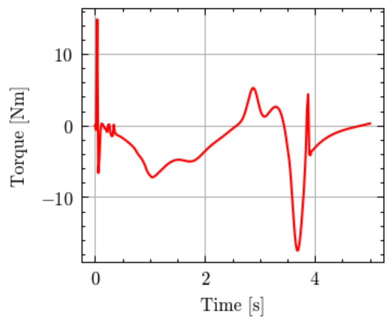
(a) Torque; Single pendulum, CS



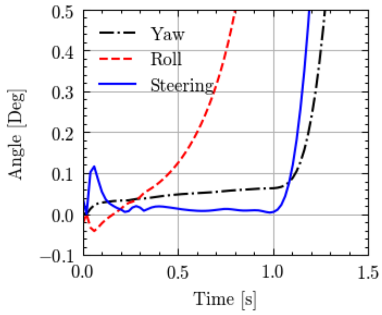
(b) Bicycle angles; Single pendulum, CS



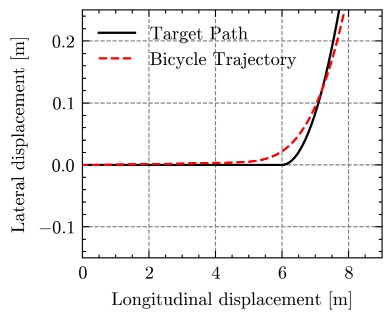
(c) Path trajectory; Single pendulum, CS



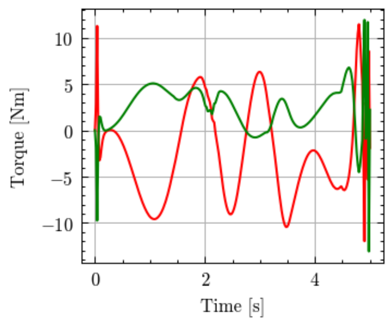
(d) Torque; Single pendulum, No-CS



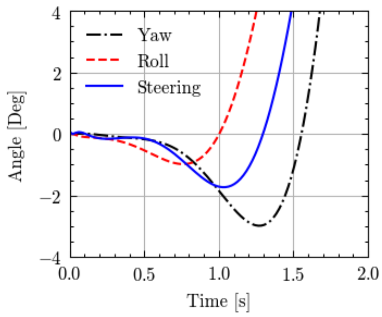
(e) Bicycle angles; Single pendulum, No-CS



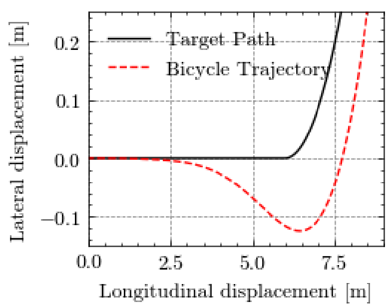
(f) Path trajectory; Single pendulum, No-CS



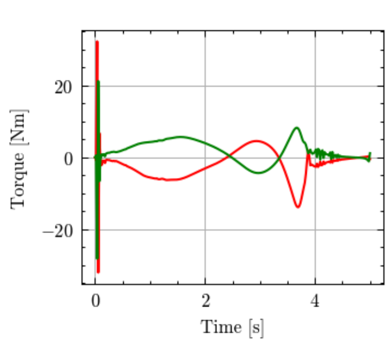
(g) Torque; Double pendulum, CS



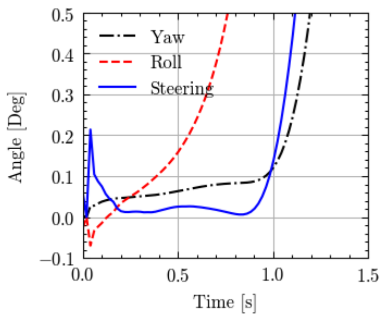
(h) Bicycle angles; Double pendulum, CS



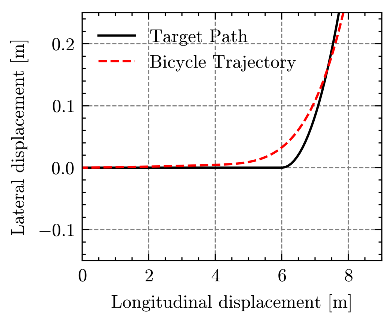
(i) Path trajectory; Double pendulum, CS



(j) Torque; Double pendulum, No-CS

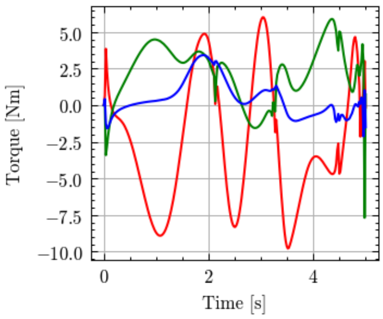


(k) Bicycle angles; Double pendulum, No-CS

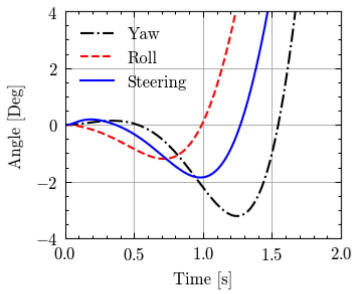


(l) Path trajectory; Double pendulum, No-CS

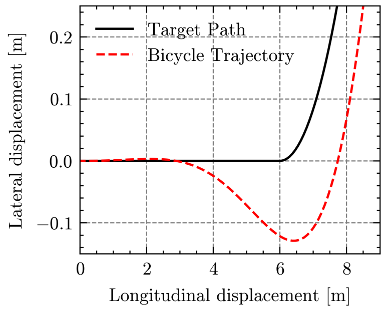
Figure C.1: Exerted rider torque(s), bicycle angles and path trajectory of the three different models, with and without counter-steering.
CS = Countersteering, No-CS = Not Countersteering.
Torques: —: Seat Torque, —: Torso Torque, —: Neck Torque.



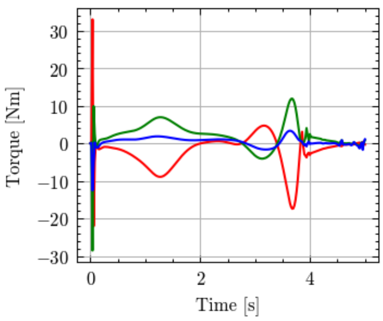
(a) Torque; Triple pendulum, CS



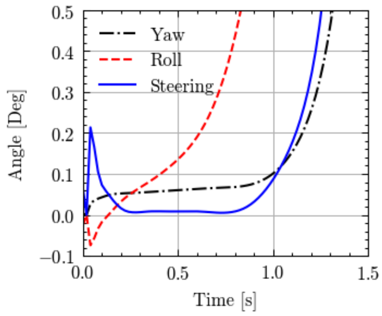
(b) Bicycle angles; Triple pendulum, CS



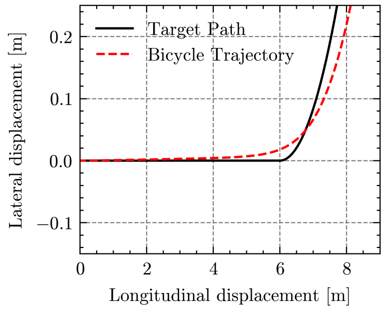
(c) Path trajectory; Triple pendulum, CS



(d) Torque; Triple pendulum, No-CS

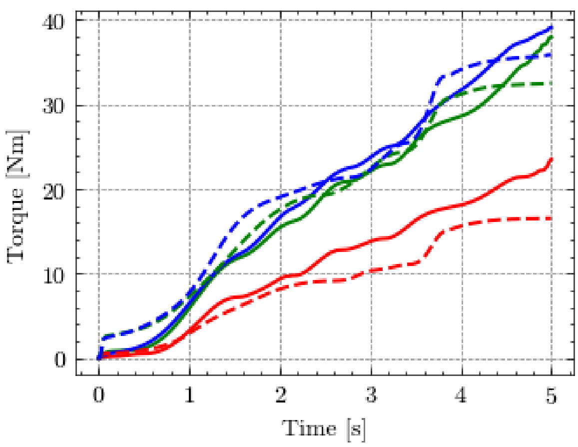


(e) Bicycle angles; Triple pendulum, No-CS

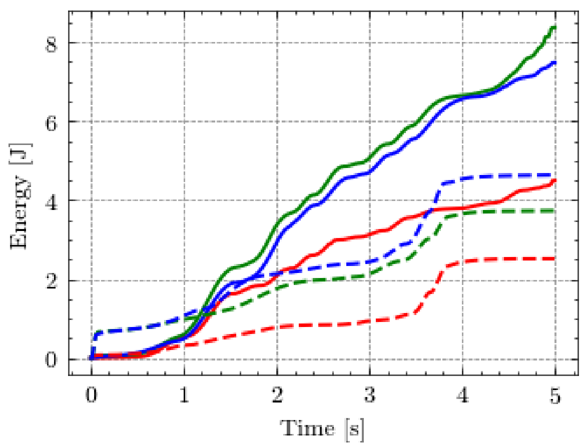


(f) Path trajectory; Triple pendulum, No-CS

Figure C.1: (continued): Exerted rider torque(s), bicycle angles and path trajectory of the three different models, with and without counter-steering.
Torques: —: Seat Torque, —: Torso Torque, —: Neck Torque.

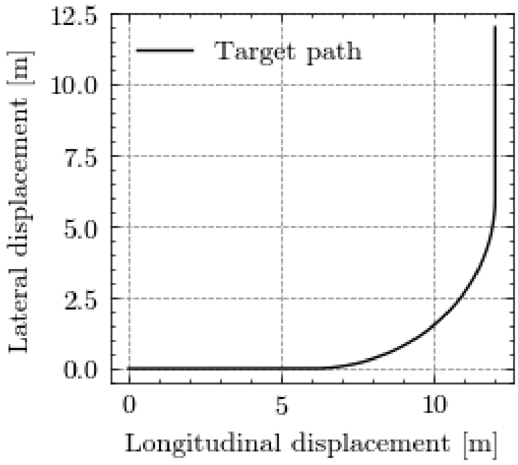


(a) Torque

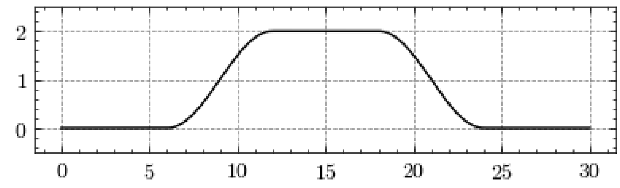


(b) Energy

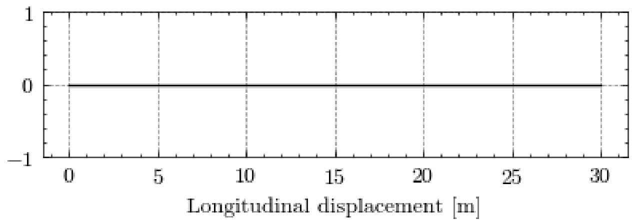
Figure C.2: Torque and Energy comparison of the different models during the $90°$ turn task.
—: Single Pendulum, —: Double Pendulum, —: Triple Pendulum
—: Counter-steering, - -: No Counter-steering



(a) 90° Turn



(b) Double Lane Switch



(c) Perturbed Cycling

Figure C.3: The three different paths trajectories used for this research.

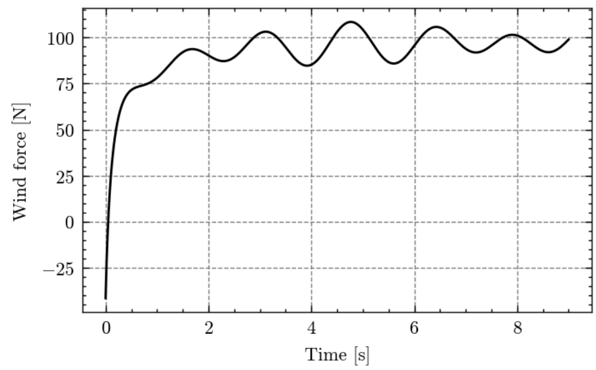


Figure C.4: The wind function $F_w(t)$

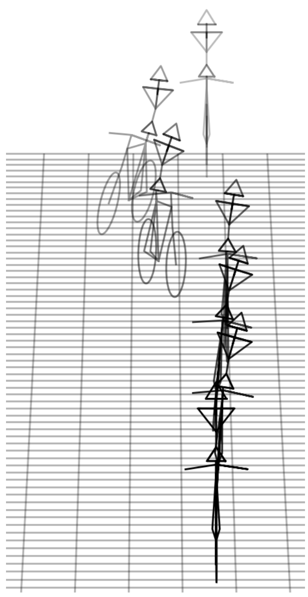


Figure C.5: Timelapse of the Double Pendulum rider with CombiSeat in response to wind perturbations at 15 km/h.