Control of Platooning Mobile Robots: Experimental Validation

Master's Thesis
DC 2017.099

Author

T.P. Beumer

Eindhoven University of Technology
Department of Mechanical Engineering
Dynamics & Control group

Coach

dr.ir. A.A.J. Lefeber

Supervisor

prof.dr. H. Nijmeijer

November 6, 2017

Summary

The research on combined longitudinal and lateral control for platooning is limited and proposed solutions often result in corner cutting. Therefore, an alternative approach is studied in this work, where the lateral and longitudinal control problem are separated. The lateral control problem is approached as a path following problem, where the following vehicle converges to the path of its predecessor. The mapping from the path of the follower to the path of its predecessor, obtained from solving the lateral control problem, can be used to solve the longitudinal control problem. This approach is already illustrated by means of simulations in previous work, but still requires validation by real-time simulations and experiments. The main objective of this research is to implement the separated lateral and longitudinal controller on an e-puck mobile robot platform, to validate the approach. The goal of this real-time controller is not only to overcome the problem of corner cutting, but also to guarantee satisfactory tracking in case of incomplete or disturbed data. First, the transition is made from a theoretical controller design in the spatial domain towards a controller in the time domain. Furthermore, adaptations are made to the controller design to obtain a real-time controller, which is practically feasible. In practice, the position and orientation are not continuously available. An observer is proposed to estimate the orientation of the e-puck mobile robots and filter the noise on the measurements. The performance of the real-time controller is examined within a simulation and experimental environment. The simulation environment is designed as a close representation of the experimental setup, taking into account the hardware and software limitations. The performance of the controller is studied for a platoon of four vehicles, which all have a lateral and longitudinal initial error. Several experiments are performed which show the cornering and longitudinal behaviour of the mobile robots. The results confirm the expectations from simulation results. Further improvements can be obtained by reducing localization errors and hardware limitations, and including unmodelled dynamics.

Acknowledgements

First of all, I would like to express my gratitude to my supervisor prof. dr. H. Nijmeijer for giving me the opportunity to work on this project. I am very grateful for the support, inspiration and valuable input during meetings.

Besides prof. dr. Nijmeijer, I would like to thank my coach, dr. ir. A.A.J. Lefeber. His feedback, guidance, and infinite patience were of great value to me during this project.

A special gratitude goes to the D&C Lab, family, and friends for their valuable discussions and support. Their company and support is very much appreciated.

Finally, I would like to thank my parents, brother and girlfriend for their everlasting support and unconditional encouragement during my studies.

Contents

Nomenclature

Acronyms

CACC	Cooperative Adaptive Cruise Control
ITS	Intelligent Transportation Systems
LES	Local Exponential Stability
UGAS	Uniform Global Asymptotic Stability
WMR	Wheeled Mobile Robot
ZOH	Zero-Order-Hold

Greek Symbols

α	mapping from the path of the follower to the path of the leader	m
κ	curvature	-
ω	angular velocity	rad/s
ϕ	orientation angle	rad
θ	steering angle	rad

Roman Symbols

a	acceleration	m
c	controller gain	m
d	inter vehicle distance	m
h	sample time	s
I	identity matrix	m
K	controller gain	m
L	vehicle length	m
R	radius	m
r	standstill distance	m
s	travelled distance	m
t	time	s
V	Lyapunov candidate function	m
v	velocity	m/s
x	position coordinate	m
y	position coordinate	m

Subscripts

e	error
f	following vehicle
i, j	indices
l	leading vehicle
r	reference vehicle

Other Symbols

$\bar{x}$	virtual reference vehicle
$\ddot{x}$	second time derivative of x
$\dot{x}$	time derivative of x
$\hat{x}$	estimation of x
x^T	transpose of x

Chapter 1

Introduction

1.1 Background

The increasing amount of vehicles on highways [8] and the cost to extend the existing highway system leads to a growing interest in intelligent transportation systems. Intelligent road and vehicle systems could result in improved traffic efficiency, and additionally safety and reduction of the fuel emissions [33, 2]. A widely proposed technique for increasing the throughput on highways is vehicle platooning. Platooning incorporates a string of several fully autonomous vehicles driving with a safe inter-vehicle distance. This method has proven to be an effective way to deal with the increasing highway occupation [31] and reduces the fuel emission due to the decreasing effect of aerodynamic drag forces between vehicles [2]. Adaptive Cruise Control (ACC) is such a control method, based on maintaining this safe inter-vehicle distance. Here, radar or LIDAR measurements are used to calculate the distance and relative speed of the preceding vehicle after which the following vehicle automatically adjusts its vehicle speed to maintain a safe inter-vehicle distance. Obstacle detection and collision avoidance, as well as Cooperative Adaptive Cruise Control (CACC) could further benefit the highway system. The originally designed comfort system of ACC can be extended to CACC. This relaxes the highway traffic and realizes longitudinal automated vehicle control based on Vehicle-to-Vehicle (V2V) communication, providing the following vehicle with information about its predecessor. This method reduces the inter-vehicle distance while maintaining string stability of the vehicle platoon [27], where string stability is interpreted as the disturbance attenuation along the vehicle string.

An important aspect of vehicle platooning is individual vehicle control and the ability of a vehicle to precisely follow another vehicle. The control of an autonomous vehicle towards the other vehicle or reference is commonly separated in longitudinal and lateral control, as was first introduced to fully automated platooning in [28]. Lateral control of autonomous vehicles is generally approached with a lane keeping method or a follow-the-leader approach. The lane keeping method uses intelligent road infrastructure, where a follow-the-leader approach is based on position and orientation of the predecessor. The lane-keeping method has proven to be robust and reliable [28]; however, the method requires magnetic reference markers, which are undesired due to the changes that need to be made to the existing road infrastructure. Tunçer et al. [30] introduces a vision-based lane-keeping system for lateral control. This has the disadvantage of blocking the lane markings when vehicles drive with a small inter-vehicle distance, which is therefore impractical. A more practical solution is the follow-the-leader approach, which uses the position and orientation of the preceding vehicle as a reference for the following vehicle, making use of radar, LIDAR or inter-vehicle communication [21].

Problems concerning the motion control of autonomous vehicles in a platoon are commonly approached as point stabilization, path following or trajectory tracking problems. Several studies provide strategies to achieve fully automated vehicle platooning, but show effects of corner cutting since the follower cuts across the curved path. Lefeber et al. [18] elaborates on the possibilities to overcome this corner-cutting problem, which is the starting point of this research. Lefeber separates the lateral control from longitudinal control as first proposed in [28]. This allows to base the longitudinal control on conventional CACC. From [27], it is known that string stability of the platoon is ensured when conventional CACC is used; stated otherwise, the spacing error is not increasing in upstream direction. Lefeber et al. approaches the lateral control problem as a path following problem. When the preceding vehicle is performing a turn,

signs of corner cutting are no longer apparent. The effectiveness of this particular controller design is illustrates by means of simulations. The approach needs to solve two differential equations; one in the spatial domain and one in the time domain. Therefore, the next step is to implement the controller in an experimental setup. This work is an extension to the work of Lefeber and validates the effectiveness of the controller in an experimental real time environment for mobile robots. The chosen platform for experimentation is an e-puck mobile robot setup where the position is determined using an overhead camera. An e-puck is a small differential wheeled mobile robot, originally designed for micro-engineering education. This experimental platform has proven to be highly effective to investigate coordination control of unicycle mobile robots using a virtual structure approach [32]. Additionally, the setup has proven to be very useful to test different control algorithms for a single or a group of unicycle mobile robots, as shown by [4, 16, 17]. For highway implementation the global coordinates should be determined from for example on-board sensors, V2V or GPS. The experimental environment gives useful insight in the performance and limitation of a real implementation. The next section further elaborates on the relevant literature to this work, which motivate the design choices of Lefeber et al. [18] and the necessity of an experimental validation.

1.2 Literature review

The tracking control problem of a unicycle mobile robot has been widely studied and has resulted in a vast amount of literature available. However, the control of platooning vehicles has been studied mainly in longitudinal direction in the form of CACC. In this section we present the most applicable literature related to the control of platooning vehicles and the occurrence of corner cutting. Before solving the control problem, it is important to properly define the topic we further discuss in this work.

1.2.1 Platoon

A platoon is normally formulated as a string of several fully automated vehicles driving with a safe inter-vehicle distance. In the existing literature, research is often conducted on well-defined platoons, i.e. a platoon leader and all following vehicles are known in advance [26]. This is a very structured environment which cannot be compared to everyday traffic, where vehicles are not identical and constant changes are present. Also the presence of one common leader is very unlikely; therefore, another approach is more likely, being *ad hoc* vehicle following. This approach eliminates the dependency on a natural platoon leader and the following vehicles only require information from themselves and their predecessor. This independency is a significant advantage as it minimizes the distance of wireless communication and gives more flexibility to the solution. Note that the platoon is not limited to a maximum number of participating vehicles. Since the platoon is a combination of multiple leader-follower interactions, this allows to study the motion control of the separate links.

1.2.2 Motion control of wheeled mobile robots

The control of wheeled mobile robots is widely studied due to the interesting challenges concerning the nonholonomic nonlinear system. Motion controllers can be mainly divided in solving three control problems: point stabilization, where the objective is to stabilize the vehicle to a desired target point; trajectory tracking, where the aim is to track a time varying reference trajectory; and path following, which focuses on stabilizing a desired path without dependency on time. Since point stabilization is not of our interest, only trajectory tracking and path following approaches are discussed. Furthermore, the control of an autonomous vehicle is commonly separated in a longitudinal and lateral controller; however, combined longitudinal and lateral controllers are also very common in tracking control. In the next section, different motion controllers are discussed which consider the control of a leader-follower combination. First, trajectory tracking is considered, where corner cutting is first observed. Moreover, the actual problem becomes clear from the proposed solutions and motivates for a path following approach. Second, path following approaches are discussed, which form the fundamentals of the lateral control as designed by Lefeber et al. [18].

1.2.3 Trajectory tracking

The follow-the-leader approach is a specific trajectory tracking problem. It appears to be very effective in regulating vehicles towards the desired trajectory, as for example proposed in [13]. The combined longitudinal and lateral controller is derived using the backstepping technique and stabilizes the error dynamics as proposed in [14], where the error posture is defined in the frame of the follower. These controllers are perfect to solve synchronization, or obstacle avoidance problems [17]; unfortunately, for control of vehicle platooning the problem of corner cutting arises, especially for platoons with a large number of vehicles. The effect of corner cutting was already pointed out by Gehrig et al. [9]. The following vehicle comes with a deviation from the path of the leader, where the current position of both vehicles is used to interpolate a trajectory between the two vehicles. The deviation grows with a larger distance to the predecessor. In [21], it is stated that this deviation is not necessarily a problem for driving on highways since the curvature of highways cannot be larger than 0.00125 1/m. However, for platooning with a large inter-vehicle distance and in an urban environment, this causes serious deviations. The general problem of using tracking controllers to control vehicles in a platoon is the choice of a proper reference point for the follower. However, if the trajectory of the leader can be reconstructed and a reference point for the lateral control of the follower can be selected, using a look-ahead distance, the autonomous vehicle is able to follow the leader according to Gehrig et al. [9]. This significantly increases the precision of the car following system. Another commonly applied method is to introduce a look-ahead distance to detect a trajectory error and predict the future curvature as presented in [10]. Here, the relative position and orientation of the preceding vehicle are measured to provide smooth steering to the desired position using a PD controller with a second order low-pass filter. In [25], an adaptive tracking controller for a two-vehicle convoy is designed, using a look-ahead approach, without the use of information about the road infrastructure or inter-vehicle communication. The proposed controller is able to overcome the problem of corner cutting, which is validated by means of simulations. The main drawback of this approach is the assumption that the predecessor is driving with a constant speed. In the work of Bayuwindra et al. [4], an extended look-ahead point has been introduced, which reduces the effect of corner cutting and is able to maintain a desired inter-vehicle distance. This is a significant improvement since it combines longitudinal and lateral control. A virtual reference for the follower vehicle was introduced as a new tracking objective. As mentioned before, Gehrig et al. [9] generated a trajectory between the predecessor and following vehicle based on stored position coordinates and motion parameters of the predecessor. This motivates to investigate the possibilities of a path following approach, which is discussed next.

1.2.4 Path following

To avoid direct trajectory tracking of autonomous vehicles which results in corner-cutting, the lateral control problem is approached as a path following problem. Path following problems are studied in multiple fields, as for example mobile robotics, marine vessels and aircrafts [6, 3], where the lateral and longitudinal control are again separated. In [22] a virtual target is introduced as a reference for the follower, which moves along the path of the predecessor and has the same kinematics as a unicycle type mobile robot. The errors are expressed with respect to the virtual target from the view of the frame of the follower. However, the proposed controller is only valid if the orientation error between the frames is within $(-\pi/2, \pi/2)$ and the vehicle moves with a constant velocity to the path. The problems that arise when the position of the virtual target is simply a projection of the actual vehicle, are overcome by [29]. This extention to the work of [22] has a virtual target which moves along the path of the leader with an adaptive velocity. This additional velocity allows to converge faster to the desired path, which is done by a backstepping based asymptotically stabilizing control law. The idea of a virtual target moving along the path of the predecessor with its own velocity, considered as an additional input, is adopted by Lefeber et al. [18]. Here, a mapping between the path of the follower and the path of the leader is obtained from solving the lateral control problem. This allows to additionally solve the longitudinal control problem as controlling two points on the same path towards a required inter-vehicle distance, comparable to conventional CACC. As already mentioned in the introduction, the controller of Lefeber et al. [18] is able to completely eliminate the effect of corner cutting. Furthermore, satisfactory behaviour for the follower is achieved in case of a large initial error. This thesis is an extension to the control of platooning vehicles as proposed in [18].

1.2.5 Experimental validation

The main part of the available literature assumes that measurements are available continuously in time and are not corrupted by noise. However, in practice the opposite is true. The number of studies which validate the control of platooning vehicles by experiments is limited. Moreover, the accuracy of the sensors, the choice of sensors, the availability of the states and the scale of the experiment do significantly influence the results. In [15], a group of automated wheeled mobile robots only relying on relative sensor information is addressed and validated by experiments. The objective is again to follow the path of the preceding vehicle precisely and maintain a desired inter-vehicle distance. No inter-vehicle communication is available and the following vehicles rely on the estimated paths of the preceding vehicle. Experiments are performed for a global and local approach for comparison. A dynamic feedback linearized controller is used to force the vehicle to precisely follow the reference path. The best performance is obtained with a global approach, where the tracking error does not accumulate through the number of robots. This is in contrast with the local approach where the error does accumulate. In [4], a similar experimental setup was used to validate the extended look-ahead approach as described earlier. The setup is equipped with a camera to measure the global position of the vehicles. This approach shows appropriate behavior but does not completely eliminate the effect of corner cutting. Experiments on larger scale are often limited to the longitudinal control problem. However, Bom et al. [5] considers a full scale experiment where the vehicles are expected to follow a curved reference path. It immediately becomes clear that wireless communication between vehicles introduces additional challenges concerning network planning and message handling to achieve the required reliability for implementation in everyday traffic. This is out of the scope of the research and therefore the experimental setup as used in Bayuwindra et al [4] is selected to validate the performance of the controller of Lefeber et al. [18]. The so called, e-puck mobile robot setup, is a simplified version of vehicles driving on highways, relying on the same sensor information, where the setup receives the coordinates from an overhead camera. For highway implementation, the global coordinates need to be determined by for example on-board sensors, V2V or GPS.

1.3 Motivation and objectives

From current research it is clear that the controllers which overcome the problem of corner cutting are limited. Additionally, the effectiveness of the designed controller is often shown in a simulation environment rather than validated by real-time experiments. The lateral and longitudinal controller of Lefeber et al. [18] proposes a solution for the corner cutting problem and achieves appropriate following behavior in case of a large initial error in simulations. However, uncertainties and disturbances are not taken into account. This motivates for a practical implementation, which comes with multiple challenges and difficulties. This is especially interesting since the approach is separated in two parts, one in the spatial domain and one in the time domain. Another point of interest is the reliability of the measured and communicated data and the minimum amount of data required for sufficient performance of the platoon. The main objective of this thesis is to investigate the validity of the combined lateral and longitudinal controller for vehicle platooning from Lefeber et al. [18], by means of real time experiments with multiple mobile robots in the platoon. To achieve this, the control law is adapted such that it can be applied and tested on a real-world mobile robot platform. Based on the performance of the controllers, suggestions and improvements can be made.
Taking these objectives into consideration, the main objective is divided in sub-objectives as addressed in this work:

- Derivation of the time based model to make a transition from the theoretical control design towards a practical application.
- Design appropriate state observers which take the limitations of an experimental implementation into account.
- Validate the effectiveness of the proposed real-time controller by simulations and experiments. The goal is to provide an extensive review of the results and provide comments to guarantee a smooth transition to a full scale vehicle application.

1.4 Outline of this thesis

This thesis consists of multiple parts. In Chapter 2, the kinematic model of a mobile car is shortly introduced and it discusses the controller of Lefeber et al. in more detail. Furthermore, the time based model is derived. Chapter 3 discusses the hardware and software components of the experimental setup. Additionally we elaborate on the limitations and methods for a practical implementation of the controller. Next, in Chapter 4 the performance of the real time controller is discussed by simulations and experiments. Special attention is given to the occurrence of corner cutting and error propagation. Furthermore, the behavior of the vehicles is studied during acceleration and deceleration, where the objective is to maintain a desired inter-vehicle distance. Finally, conclusions and recommendations for future research are presented in Chapter 5.

Chapter 2

System and controller design

The nonholonomic constraints associated with the wheeled mobile robot are a perfect motivation for nonlinear control techniques. Commonly, two types of mobile robots are studied; the unicycle mobile robot and the car-like or bicycle type wheeled mobile robot (WMR). The latter is rear wheel driven with front wheel steering, where the unicycle mobile robot has only two wheels which are on a common axis and can be actuated independently. In the remainder of this thesis, the unicycle mobile robot is considered due to the available experimental setup and its simplicity. Another advantage is the zero turning radius of the WMR [22]. The main drawbacks appear when the vehicle drives over an uneven surface and when the vehicle experiences slipping, which are not considered in this work. The chapter is organized as follows: first the kinematic model of the unicycle type mobile robot is introduced and the error definition is explained. Next, we elaborate on the controller design of Lefeber et al. [18] and discuss adaptations for a practical implementation. The lateral and longitudinal control are discussed separately for clarity. The chapter finishes with an observer design to determine the full state, which in general is not available accurately or cannot be measured.

2.1 Kinematic model of a wheeled mobile robot

The advantage of describing the motion of the mobile robot with a kinematic model, and not with a dynamic model, is the simplicity of the kinematic model. The kinematic model does not rely on the knowledge of parameters associated with the vehicle and its actuators, such as, the geometry of the vehicle, masses and mass moments of inertia [22]. This information is not necessarily required for many applications, but can be applied to improve control. However, in this case no reliable characteristic information is present; therefore, the kinematic model is used. Many mobile robots are actuated with electric motors, which receive input commands from low-level controllers based on the kinematic model of a unicycle mobile car. These controllers are based on the principle of controlling only the forward and angular velocity. A good motor-controller combination achieves a small difference between the desired velocities and the actual velocities, even when the desired velocity and the motor load are varying continuously. The nonholonomic kinematic unicycle model is described by the following differential equations:

$$\begin{aligned} \dot{x} &= v\cos\theta \\ \dot{y} &= v\sin\theta \\ \dot{\theta} &= \omega. \end{aligned} \tag{2.1}$$

A schematic representation of the two wheeled mobile robot is presented in Figure 2.1 in a 2D plane. The x and y coordinates describe the position of the center of the mobile robot with respect to the fixed earth frame O. The orientation angle θ is the angle between the heading of the vehicle and the x-axis, taken counterclockwise. The entire sequel of the points (x, y) is referred to as path or trajectory. The forward velocity v and the angular velocity ω are considered to be the controlled inputs. This model assumes that there is no slip, i.e., the lateral velocity is zero, which represents the nonholonomic constraint

$$\dot{x}\sin\theta - \dot{y}\cos\theta = 0. \tag{2.2}$$

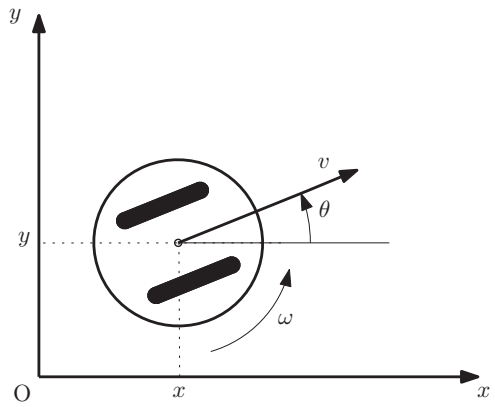

Figure 2.1: Schematic representation of a two wheeled mobile robot.

The control objective is to follow a reference path or trajectory (x_r, y_r) with the corresponding orientations (θ_r), and reference velocities v_r, ω_r. The error between the reference and the vehicle is defined as difference between the postures in the local frame of the current vehicle. A transformation of the reference frame to the frame of the current vehicle needs to be made as proposed by [14] and shown in Figure 2.2. Due to the coordinate transformation, the frame is independent from the global coordinate frame O. This eventually results in the following error coordinates:

$$\begin{bmatrix} x_e(t) \\ y_e(t) \\ \theta_e(t) \end{bmatrix} = \begin{bmatrix} \cos\theta(t) & \sin\theta(t) & 0 \\ -\sin\theta(t) & \cos\theta(t) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_r(t) - x(t) \\ y_r(t) - y(t) \\ \theta_r(t) - \theta(t) \end{bmatrix}. \tag{2.3}$$

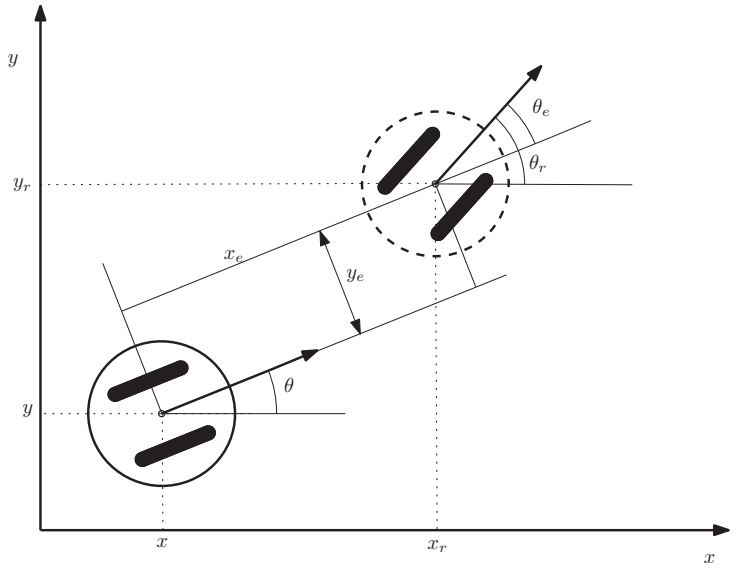

Figure 2.2: The tracking errors x_e, y_e and θ_e as can be found from the coordinates from the reference trajectory and following vehicle.

The error dynamics follow from deriving the error coordinates with respect to time, which results in

$$\dot{x}_e = \omega y_e - v + v_r \cos\theta_e \tag{2.4}$$

$$\dot{y}_e = -\omega x_e + v_r \sin\theta_e \tag{2.5}$$

$$\dot{\theta}_e = \omega_r - \omega. \tag{2.6}$$

For the convergence of a vehicle towards the desired reference, tracking errors x_e, y_e and θ_e must converge to zero over time with a stabilizing control law for $v(t), \omega(t)$. Multiple tracking controllers are available

as for example presented in [13] and [14] as already discussed in the literature review. This is discussed to a greater extent in Chapter 4. The error coordinates as formulated here in the frame of the follower, are used in the remainder of this work. Next, an extension to the lateral and longitudinal controller as originally proposed in Lefeber et al. [18] is presented.

2.2 Controller Design

The combined lateral and longitudinal control problem is studied separately. This chapter elaborates on the lateral control problem first, followed by the longitudinal control problem as proposed by [18].

2.2.1 Lateral controller

The lateral control problem is approached as a path following problem in the spatial domain. In order to use the kinematic model as given in (2.1) for a path following approach, the vehicles have to be expressed in the spatial domain. Introduce the traveled distance along the path of the vehicle $s(t)$. To transform the kinematic equations to the spatial domain, we can use $v(t) = \frac{ds(t)}{dt}$. This characterizes the path by

$$\begin{aligned} \frac{d}{ds}x(s(t)) &= \cos\theta(s(t)) \\ \frac{d}{ds}y(s(t)) &= \sin\theta(s(t)) \\ \frac{d}{ds}\theta(s(t)) &= \kappa(s(t)) \end{aligned} \tag{2.7}$$

where $\kappa(s(t))$ denotes the curvature of the path when the vehicle has traveled a distance $s(t)$. Note that the curvature $\kappa(s(t)) = \frac{1}{R}$, with R the radius of a circle. When $\kappa = 0$, this corresponds to a straight line, and the curve with constant curvature κ is a circle with radius $\frac{1}{\kappa}$.

To correctly formulate the lateral control problem, a leader and follower need to be present. Both vehicles generate a path, which can be described by the characteristics as described in (2.7). In order to distinguish between the two vehicles the leading vehicle is indicated with the subscript l, resulting in

$$\begin{aligned} \frac{d}{ds_l}x_l(s_l) &= \cos\theta_l(s_l) \\ \frac{d}{ds_l}y_l(s_l) &= \sin\theta_l(s_l) \\ \frac{d}{ds_l}\theta_l(s_l) &= \kappa_l(s_l), \end{aligned} \tag{2.8}$$

where the dependency on time t is omitted for readability and $\kappa(s_l)$. The path of the follower is still characterized by (2.7). Consider a feasible trajectory of the leading vehicle $[x_l(s_l), y_l(s_l), \theta_l(s_l), \kappa_l(s_l)]$ and a diffeomorphism α: $\mathbb{R}^+ \to \mathbb{R}^+$, $s_l = \alpha(s)$. This diffeomorphism is an invertible function which is a map between differential manifolds such that both the function and its inverse are smooth. This diffeomorphism is introduced to compare the traveled distance of the following vehicle to the traveled distance of the preceding vehicle since the traveled distance to a particular point is different for both vehicles. For solving the lateral control problem, an appropriate control law $\kappa(s)$ needs to be found, such that

$$\lim_{s_l\to\infty} x_l(s_l) - x(\alpha^{-1}(s_l)) = 0 \tag{2.9}$$

$$\lim_{s_l\to\infty} y_l(s_l) - y(\alpha^{-1}(s_l)) = 0 \tag{2.10}$$

$$\lim_{s_l\to\infty} \theta_l(s_l) - \theta(\alpha^{-1}(s_l)) = 0, \tag{2.11}$$

when the path of the preceding vehicle, $[x_l(s_l), y_l(s_l), \theta_l(s_l), \kappa_l(s_l)]$, is feasible and $\kappa_l(s_l)$ is bounded. For solving this lateral control problem, a virtual vehicle is introduced, which drives along the trajectory of the leader. As mentioned before in the literature review, it is desired to regulate the velocity of this virtual

vehicle as an extra control input. This allows us to position the virtual vehicle closer to the follower, resulting in a stronger convergence to the path of the follower. This virtual vehicle travels a distance $s_l^* = \alpha(s)$ along this path, when the follower has traveled a distance s. Here, s_l^* is a distance on the path of the leader corresponding to the traveled distance of the follower. First, let us define the trajectory of the virtual vehicle by $\bar{x}_l(s) = x_l(\alpha(s))$, $\bar{y}_l(s) = y_l(\alpha(s))$, $\bar{\theta}_l(s) = \theta_l(\alpha(s))$, and $\bar{\kappa}_l(s) = \kappa_l(\alpha(s))$ and let $\bar{v}(s) = \frac{d\alpha(s)}{ds}$ be the extra velocity control of the virtual vehicle. For the characteristics of the virtual vehicle we then obtain

$$\begin{aligned} \frac{d\bar{x}_l(s)}{ds} &= \bar{v}(s)\cos\bar{\theta}_l(s) \\ \frac{d\bar{y}_l(s)}{ds} &= \bar{v}(s)\sin\bar{\theta}_l(s) \\ \frac{d\bar{\theta}_l(s)}{ds} &= \bar{v}(s)\bar{\kappa}_l(s). \end{aligned} \tag{2.12}$$

Consider again the error coordinates as in (2.3), and the visualization in Figure 2.2. The virtual vehicle is the reference for the follower in the spatial domain, which slightly changes the error definition:

$$\begin{bmatrix} x_e(s) \\ y_e(s) \\ \theta_e(s) \end{bmatrix} = \begin{bmatrix} \cos\theta(s) & \sin\theta(s) & 0 \\ -\sin\theta(s) & \cos\theta(s) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \bar{x}_l(s) - x(s) \\ \bar{y}_l(s) - y(s) \\ \bar{\theta}_l(s) - \theta(s) \end{bmatrix}, \tag{2.13}$$

where the dependency on t is again omitted for readability. The error dynamics can be derived and are given by

$$\begin{aligned} \dot{x}_e &= \kappa y_e + \bar{v}\cos\theta_e - 1 \\ \dot{y}_e &= -\kappa x_e + \bar{v}\sin\theta_e \\ \dot{\theta}_e &= -\kappa + \bar{v}\bar{\kappa}_l, \end{aligned} \tag{2.14}$$

where the dependency on s is dropped for the ease of exposition. The Lyapunov function

$$V = \frac{1}{2}x_e^2 + \frac{1}{2}y_e^2 - \frac{1}{c_3}\log(\cos\theta_e), \tag{2.15}$$

is proposed, which is positive definite if $c_3 > 0$. Furthermore it is assumed that the initial conditions satisfy $|\theta_e(0)| < \pi/2$. The derivative of V and the substitution of the error dynamics as given in (2.14) results in:

$$\frac{dV}{ds} = x_e(\bar{v}\cos(\theta_e) - 1) + y_e(\bar{v}\sin\theta_e) + \frac{1}{c_3}\tan\theta_e(-\kappa + \bar{v}\bar{\kappa}_l). \tag{2.16}$$

The control law, given by

$$\bar{v} = \frac{1 - c_1\sigma_1(x_e)}{\cos\theta_e} \tag{2.17}$$

$$\kappa = c_3 y_e(1 - c_1\sigma_1(x_e)) + \bar{v}\bar{\kappa}_l + c_2\sigma_2(\theta_e), \tag{2.18}$$

with $0 < c_1 < 1$ and $c_2 > 0$ results in

$$\frac{dV}{ds} = -c_1 x_e\sigma_1(x_e) - \frac{c_2}{c_3}\sigma_2(\theta_e)\tan\theta_e \leq 0. \tag{2.19}$$

Here, $\sigma(\tau)$ is a continuous monotone function, differentiable at $\tau = 0$, satisfying $\sigma(\tau)\tau > 0$ for $\tau \neq 0$, $|\sigma(\tau)| \leq 1$, and $\sigma'(0) > 0$. The proof, as given by Lefeber et al. [18], that x_e, y_e and θ_e converge to zero is given in Appendix A, resulting in an UGAS and locally exponentially stable (LES) closed-loop system.

2.2.2 Time based model for lateral control

The developed lateral controller is dependent on the traveled distance of the following vehicle. To be able to use the controller for real-time control the trajectories expressed in traveled distance must be time dependent. Differentiate these equation with respect to time, following

$$\frac{d}{dt}x(t) = \frac{d}{ds(t)}x(s(t))\frac{d}{dt}s(t) \tag{2.20}$$

and substitute $\frac{d}{dt}s(t) = v(t)$. This results in the real time model given by

$$\begin{aligned} \dot{x}_l(t) &= v_l(t)\cos\theta_l(t) \\ \dot{y}_l(t) &= v_l(t)\sin\theta_l(t) \\ \dot{\theta}_l(t) &= v_l(t)\kappa_l(t). \end{aligned} \tag{2.21}$$

For the time based model of the following vehicle we have naturally,

$$\begin{aligned} \dot{x}(t) &= v(t)\cos\theta(t) \\ \dot{y}(t) &= v(t)\sin\theta(t) \\ \dot{\theta}(t) &= v(t)\kappa(t) \end{aligned} \tag{2.22}$$

where $v(t)$ is the velocity controlled by the longitudinal controller and $\kappa(t)$ is the control action towards the virtual reference vehicle. The motion of the virtual reference vehicle is described by

$$\begin{aligned} \dot{\bar{x}}(t) &= \bar{v}(t)v(t)\cos\bar{\theta}(t) \\ \dot{\bar{y}}(t) &= \bar{v}(t)v(t)\sin\bar{\theta}(t) \\ \dot{\bar{\theta}}(t) &= \bar{v}(t)v(t)\bar{\kappa}_l(t). \end{aligned} \tag{2.23}$$

Here, it is again used that $\bar{v}(s) = \frac{d\alpha(s)}{ds}$. All information is now transformed to the time domain, which is a first step towards implementation of the controller in real-time. We now proceed to the design of the longitudinal controller.

2.2.3 Longitudinal control

The longitudinal control problem is defined here as the control of two points on the same path towards a required inter-vehicle distance. This 1-dimensional (1D) control problem is comparable to a Cooperative Adaptive Cruise Control (CACC) approach where two points on a straight line are controlled towards the desired inter-vehicle distance [26]. The main difference is the definition of the longitudinal error, which is based on the traveled distance instead of position. The longitudinal distance based error is minimized for all vehicles in the platoon,

$$\lim_{t\to\infty} e_i = 0 \qquad \text{for } i = 1, 2, ..., n \tag{2.24}$$

where n is the number of vehicles in the platoon and where e_i is the error in longitudinal direction with respect to the preceding vehicle $i-1$. A schematic representation of a platoon of vehicles is depicted in Figure 2.3, where L_i is the vehicle length, r_i is the standstill distance, h is the time gap, and v_i is the forward velocity of the vehicle. The inter-vehicle distance d_i is determined according to the constant time gap policy from [26], which is given by

$$d_i = r_i + hv_i. \tag{2.25}$$

From this method it is known that it provides more string stability in the platoon compared to a constant distance policy as in [23]. Note that the first vehicle, has no predecessor and therefore no specific desired

distance policy.

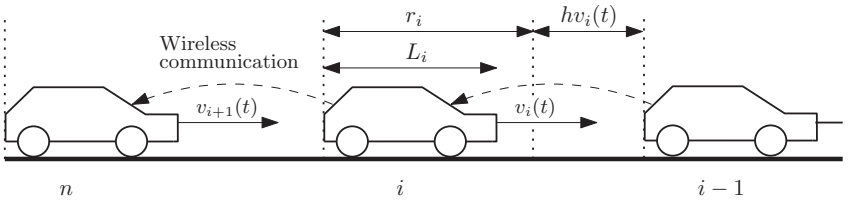

Figure 2.3: Schematic representation of a platoon of vehicles. The reference point is the indicated by the dotted line at the back of the vehicle.

Only the rare case where both vehicles are on the same path, the desired inter-vehicle distance can be considered as a 1D-problem as given by (2.25). Therefore, the distance needs to be defined in 2D, which leaves two options: (1) projecting the leader on the path of the follower, or (2) projecting the follower on the path of the leader. The mapping $s_l = \alpha(s)$, was already introduced to solve the lateral control problem and can now be used to solve the longitudinal control problem. This mapping allows to define the longitudinal error as the difference between the virtual vehicle and the preceding vehicle, $s_l - \alpha(s)$. The same holds for the opposite, i.e., projecting the leading vehicle on the planned path of the following vehicle, $\alpha^{-1}(s_l) - s$, providing the error in the perspective of the following vehicle. The latter is used to define the longitudinal control problem:

$$\lim_{t\to\infty} e(t) = \alpha^{-1}(s_l(t)) - s(t) - hv(t) - r = 0, \tag{2.26}$$

where the velocity profile $v(t)$ needs to be determined. Since this error contains the forward velocity, the kinematic model of a mobile robot is extended with the equation $\dot{v}(t) = a(t)$, which now takes the acceleration $a(t)$ as input. Using this, we obtain

$$\dot{e}(t) = \frac{v_l(t)}{\bar{v}(\alpha^{-1}(s(t)))} - v(t) - ha(t). \tag{2.27}$$

We now take the controller

$$a(t) = \frac{1}{h}\left[\frac{v_l(t)}{\bar{v}(\alpha^{-1}(s(t)))} - v(t) + k\sigma(e(t))\right] \tag{2.28}$$

with $k > 0$, which gives

$$\dot{e}(t) = -k\sigma(e(t)), \tag{2.29}$$

and provides a globally asymptotically stable (GAS) and locally exponentially stable (LES) system.

A more careful look at (2.26) for $t = 0$, where $s_l = s_f = 0$, shows an negative initial error, $e(0) = -hv(0) - r$. This error is always present, and always causes the follower to decelerate. This is undesired and increases the longitudinal error even when the follower is far behind the preceding vehicle. To overcome this problem, the state of the vehicles needs to be defined properly from the start. Stated otherwise, all information required for a correct determination of the control inputs needs to be initially present in a correct way. Figure 2.4 shows the paths of the leader and the follower with the corresponding projections relative to one another.

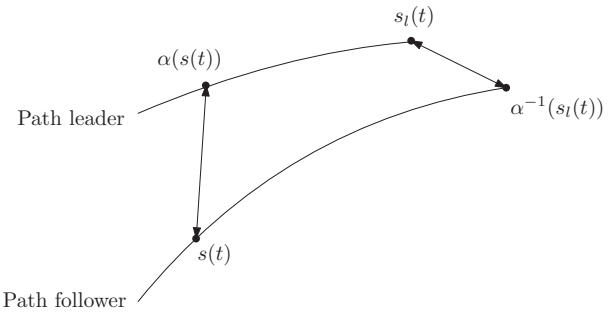

Figure 2.4: Representation of the necessary information to properly define the longitudinal error. The vehicles can be projected on the path of the leader (upper path) or on the path of the follower (lower path). This is only possible if $s_l \gg 0$.

It is clear that the path of the leader must be known to project the follower on this path. Therefore, we propose to introduce an artificial historic path for the leader which ends at the initial position of the leader. The approach is visualized in Figure 2.5. In this case the leader has zero orientation angle, where on the other hand, the follower has an orientation angle θ. The introduction of a historic path comes with possibilities for a free choice of length and direction of this path. A natural choice is to use the longitudinal position based error as the length of the path. This error can be considered in the frame of the leader or in the frame of the follower, giving $x_{e,l}$ and $x_{e,f}$ respectively. For the sake of simplicity, the artificial historic path is here taken as a straight line with the same heading as the vehicle. However, it must be noted that the path is not optimal and can be adapted to overcome limitations or generate a smoother process. Moreover, the sampling distance can be chosen optimal. The last variation is the initial position of the virtual reference vehicle which drives along the path of the leader. In case of an artificial path with length $x_{e,f}$, it is possible to determine an initial position at the path indicated by δ in Figure 2.5. This is causing the vehicle to converge even faster towards the path of the leader.

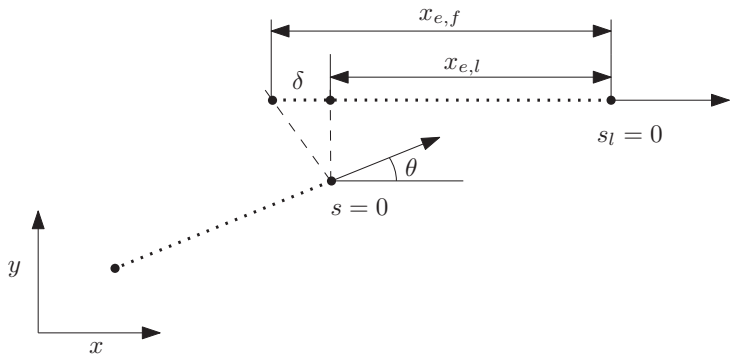

Figure 2.5: Historic path for the leader and follower, indicated with the dotted lines, for a length x_e in the frame of the follower and leader, $x_{e,f}$ and $x_{e,l}$ respectively.

The approach of an artificial historic path behind the leader has the advantage that this artificial historic path is perfectly known and needs no additional calculations after the path is created. This is in contrast with the solution proposed in Lefeber et al. [18], where the leader has planned a path for a distance Δ into the future, which is communicated to the follower using V2V communication. When the predecessor has planned a future path of distance Δ, the associated point on the path of the follower is given by $\alpha^{-1}(s_l(t) + \Delta)$. This point is closer to the path of the predecessor since the trajectory of the follower converges towards the path of the predecessor. This changes (2.26) to

$$\lim_{t\to\infty} e(t) = \alpha^{-1}(s_l(t) + \Delta) - (s(t) + \Delta) - hv(t) - r = 0, \tag{2.30}$$

which defines the state at $t = 0$ properly and overcomes the problem of a negative initial error. However, the associated point on the planned path of the follower must be accurately known for control. It is inevitable that the vehicles deviate from the planned path in a real time experiment, due to inaccurate measurements or neglected dynamics. In Lefeber et al. [18] the position of the vehicles is required for

a distance Δ into the future. The model for the future vehicle is derived, describing the behaviour of the future vehicle. The position of the future vehicle can now be determined using the communicated velocities and the initial position of the vehicles for a distance Δ into the future. Since there is no further feedback on the position and orientation of the future vehicle, the future path deviates from the actual path obtained by the vehicles. It is necessary to re-plan the future path at every time step, which is an expensive solution where good performance is not guaranteed. Therefore, solving the longitudinal control problem using a future path is not feasible from a practical point of view.

2.3 Observer design

The proposed controller design uses the assumption that all states of the kinematic model are available for control. In practice, this is often not the case and an observer is required to estimate the states. Here, we consider the case where the orientation of vehicles is not available, or corrupted with noise and biased, and therefore must be derived from position measurements, which are possible corrupted with noise as well. Therefore, a full state observer is discussed, which estimates the orientation and filters the noise on the position measurements. The problem of estimating the orientation for unicycle type mobile robots is addressed by Noijen et al. [24]. A state feedback controller is combined with an observer that estimates the orientation error using the available position measurements. However, this is undesired since this incorporates a different tracking controller than the controller as proposed here. Jakubiak et al. [11] considers that only two of the tracking error coordinates are measured and develops observers for each of the error coordinates. However, this again results in a combined observer-controller design and therefore cannot be used. Nevertheless, in [12] a full state four dimensional observer is proposed which assures asymptotic stability of the error dynamics. This approach is extended with the proof that the orientation angle θ can be determined from the estimated states and converges to the actual orientation angle.

2.3.1 Problem formulation

Consider the kinematic model as given in (2.1), but with a slightly changed notation. Here, the states are $x, y, \theta = x_1, x_2, x_3$ and the inputs $v, \omega = u_1, u_2$, resulting in

$$\begin{aligned} \dot{x}_1 &= u_1 \cos x_3 \\ \dot{x}_2 &= u_1 \sin x_3 \\ \dot{x}_3 &= u_2 \end{aligned} \tag{2.31}$$

with outputs

$$\begin{aligned} y_1 &= x_1 \\ y_2 &= x_2. \end{aligned} \tag{2.32}$$

Then following Jakubiak [12], the dimension of the system (2.31) is extended by defining new variables s and c, which replace the orientation angle x_3,

$$c = \cos x_3 \tag{2.33}$$

$$s = \sin x_3. \tag{2.34}$$

Using this, the system takes the form

$$\begin{aligned} \dot{x}_1 &= u_1 c \\ \dot{x}_2 &= u_1 s \\ \dot{c} &= -u_2 s \\ \dot{s} &= u_2 c \end{aligned} \tag{2.35}$$

where x_1 and x_2 are still the measured outputs. The system is transformed from a three dimensional to a four dimensional system, which introduces a constrain of the form $s^2 + c^2 = 1$. However, an observer for the system does not necessarily satisfy this constrain.

2.3.2 Observer design

As an observer for the system (2.31), a copy of the system is defined with an additional vector $f(x_1, x_2, \hat{x}_1, \hat{x}_2, \hat{c}, \hat{s}, u_1, u_2)$

$$\begin{aligned} \dot{\hat{x}}_1 &= u_1\hat{c} + f_1 \\ \dot{\hat{x}}_2 &= u_1\hat{s} + f_2 \\ \dot{\hat{c}} &= -u_2\hat{s} + f_3 \\ \dot{\hat{s}} &= u_2\hat{c} + f_4. \end{aligned} \tag{2.36}$$

Jakubiak et al. [12] proposes an observer with appropriate functions f, where f takes into account the estimation error, resulting in the form

$$\begin{aligned} \dot{\hat{x}}_1 &= u_1\hat{c} + l_1\tilde{x}_1 \\ \dot{\hat{x}}_2 &= u_1\hat{s} + l_2\tilde{x}_2 \\ \dot{\hat{c}} &= -u_2\hat{s} + l_3u_1\tilde{x}_1 \\ \dot{\hat{s}} &= u_2\hat{c} + l_4u_1\tilde{x}_2 \end{aligned} \tag{2.37}$$

with gains $l_i > 0$ for $i \in \{1, 2, 3, 4\}$ and observer errors

$$\begin{aligned} \tilde{x}_1 &= x_1 - \hat{x}_1 \\ \tilde{x}_2 &= x_2 - \hat{x}_2 \\ \tilde{c} &= \cos x_3 - \hat{c} \\ \tilde{s} &= \sin x_3 - \hat{s}. \end{aligned} \tag{2.38}$$

If the observer errors converge to zero, this implies that the estimated values converge to the true state of the system (2.31). Furthermore, this results in the following observer error dynamics

$$\begin{aligned} \dot{\tilde{x}}_1 &= u_1\tilde{c} - l_1\tilde{x}_1 \\ \dot{\tilde{x}}_2 &= u_1\tilde{s} - l_2\tilde{x}_2 \\ \dot{\tilde{c}} &= -u_2\tilde{s} - l_3u_1\tilde{x}_1 \\ \dot{\tilde{s}} &= u_2\tilde{c} - l_4u_1\tilde{x}_2. \end{aligned} \tag{2.39}$$

To verify the stability, we differentiate the positive definite Lyapunov function V_1

$$V_1 = \frac{l_3}{2}\tilde{x}_1^2 + \frac{l_4}{2}\tilde{x}_2^2 + \frac{1}{2}\tilde{c}^2 + \frac{1}{2}\tilde{s}^2 \tag{2.40}$$

along the observer error dynamics, resulting in

$$\begin{aligned} \dot{V}_1 &= l_3\tilde{x}_1u_1\tilde{c} - l_3l_1\tilde{x}_1^2 + l_4\tilde{x}_2u_1\tilde{s} - l_2l_4\tilde{x}_2^2 - u_2\tilde{c}\tilde{s} - l_3u_1\tilde{x}_1\tilde{c} + u_2\tilde{c}\tilde{s} - l_4u_1\tilde{x}_2\tilde{s} \\ &= -l_1l_3\tilde{x}_1^2 - l_2l_4\tilde{x}_2^2 = Y_1(\tilde{x}), \end{aligned} \tag{2.41}$$

which is negative semi-definite. We can conclude that $\tilde{x}_1, \tilde{x}_2, \tilde{c}$ and $\tilde{s}$ are bounded since $V_1 > 0$ and $\dot{V}_1 \leq 0$. Additionally, if $\dot{V}_1 \leq Y_1(\tilde{x}) \leq 0$, uniformly globally asymptotic stability can be obtained according to the adapted Matrosov theorem as stated in Appendix A. Differentiating the function

$$V_2 = -\dot{\tilde{x}}_1\tilde{x}_1 - \dot{\tilde{x}}_2\tilde{x}_2, \tag{2.42}$$

along the observer error dynamics results in

$$\begin{aligned}\dot{V}_2 &= -\ddot{\tilde{x}}_1\tilde{x}_1 - \ddot{\tilde{x}}_2\tilde{x}_2 - \dot{\tilde{x}}_1^2 - \dot{\tilde{x}}_2^2 \\ &= -\ddot{\tilde{x}}_1\tilde{x}_1 - \ddot{\tilde{x}}_2\tilde{x}_2 - (u_1\tilde{c} - l_1\tilde{x}_1)^2 - (u_1\tilde{s} - l_2\tilde{x}_2)^2 \\ &= \underbrace{-\ddot{\tilde{x}}_1\tilde{x}_1 - \ddot{\tilde{x}}_2\tilde{x}_2 - l_1^2\tilde{x}_1^2 - l_2^2\tilde{x}_2^2 + 2l_1u_1\tilde{c}\tilde{x}_1 + 2l_2u_1\tilde{s}\tilde{x}_1}_{=0 \text{ if } Y_1=0} - u_1^2(\tilde{c}^2 + \tilde{s}^2).\end{aligned} \tag{2.43}$$

So, under the assumption that $|u_1(t)| \geq \epsilon > 0$ for all t, we obtain Uniform Global Asymptotic Stability (UGAS) of the observer error dynamics. The convergence of $\tilde{c}$ and $\tilde{s}$ to zero can be used to calculate the orientation angle $\hat{x}_3$. This additionally reveals the main disadvantage of the observer, since two variables are used to estimate the orientation angle. The proof that $\hat{x}_3$ converges to the true state is provided next.

2.3.3 Orientation angle

Based on the observer design, the convergence of the estimated orientation angle is studied. Consider the estimated orientation angle,

$$\hat{x}_3 = \text{atan2}(\hat{s}, \hat{c}), \tag{2.44}$$

where $\hat{s}$ and $\hat{c}$ are generated by the observer (2.37). We then obtain the following,

$$\sin\hat{x}_3 = \frac{\hat{s}}{\sqrt{\hat{s}^2 + \hat{c}^2}} \qquad \cos\hat{x}_3 = \frac{\hat{c}}{\sqrt{\hat{s}^2 + \hat{c}^2}} \qquad \tan\hat{x}_3 = \frac{\hat{s}}{\hat{c}}. \tag{2.45}$$

Furthermore,

$$\begin{aligned}\tan(\hat{x}_3 - x_3) &= \frac{\tan\hat{x}_3 - \tan x_3}{1 + \tan\hat{x}_3\tan x_3} \\ &= \frac{\hat{s}/\hat{c} - \tan x_3}{1 + \hat{s}/\hat{c}\tan x_3} \\ &= \frac{\hat{s}\cos x_3 - \hat{c}\sin x_3}{\hat{c}\cos x_3 + \hat{s}\sin x_3} \\ &= \frac{(\sin x_3 - \tilde{s})\cos x_3 - (\cos x_3 - \tilde{c})\sin x_3}{(\cos x_3 - \tilde{c})\cos x_3 + (\sin x_3 - \tilde{s})\sin x_3} \\ &= \frac{-\tilde{s}\cos x_3 + \tilde{c}\sin x_3}{1 - \tilde{c}\cos x_3 - \tilde{s}\sin x_3}.\end{aligned} \tag{2.46}$$

Since $\lim_{t\to\infty}\tilde{c}(t) = \lim_{t\to\infty}\tilde{s}(t) = 0$, we can directly conclude that we also have $\lim_{t\to\infty}\tan\big(\hat{x}_3(t) - x_3(t)\big) = 0$. For the orientation angle this results in converges to the real orientation angle x_3 if the initial estimated orientation error satisfies $|\hat{x}_3(0) - x_3(0)| < \frac{\pi}{2}$.

2.4 Concluding remarks

In this chapter, the lateral controller design is considered as originally designed by Lefeber et al. [18]. With the aim of a practical real time implementation the time based lateral controller is derived. Furthermore, the longitudinal control problem is solved by approaching it as a 1D problem of controlling two points on the same path. The mapping as originated from solving the lateral control problem, is used to project the follower on the path of the leader. However, since the error is defined as a function of the traveled distance, this causes a negative initial error. This causes the vehicles to decelerate from the beginning and is therefore undesired. The solution proposed by Lefeber et al. [18] is based on the assumption that all states and inputs are accurate and are not disturbed over time. However, with the aim of a practical implementation, the proposed solution needs to be adapted since the measurements are inaccurate and the velocities cannot be directly measured, which does not guarantee satisfying performance. Therefore, an alternative solution is proposed which overcomes this problem by introducing an artificial historic path of the leader. Now, the follower is able to project its position on this path and properly define the error.

Since the orientation angle is in general not accurately available, the assumption is made that only position measurements are available. Therefore, a four dimensional observer is considered to estimate the full state as first proposed by Jakubiak [12]. The stability proof is extended with the proof of UGAS of the observer error dynamics. Additionally the orientation angle is constructed from the state estimates. It has been shown that the difference between the real angle and the estimated angle converges to zero.

The continuous time control law and the continuous time observer cannot be directly implemented in a real-time experimental environment. The next chapter elaborates on the implementation of the proposed approach on the e-puck mobile robot platform and the adaptations that need to be made.

Chapter 3

Setup and implementation

This chapter discusses the transition from a theoretical approach to a practical implementation of the controller as proposed in the previous chapter. We want to validate the proposed methods by means of discrete-time simulations and experiments, with the aim to obtain a simulation environment which is a close representation of the experimental environment, including measurement noise and uncertainties. The experimental setup is briefly discussed first to get a clear overview of the hardware components and the control structure. Second, the global control structure of the experimental platform is introduced, which is adopted in the simulation environment. Furthermore, the limitations of the setup are discussed. Eventually we elaborate on the implementation of the controller design.

3.1 Experimental Setup

The experimental setup contains several aspects. First the hardware of the platform is discussed. Second, a global overview of the control structure and the corresponding hardware components are given. Additionally, we elaborate on the hardware platform and software limitations that occur during experiments.

3.1.1 Hardware components

The chosen platform to validate the controller of [18], as described in this thesis, is an e-puck mobile robot setup, which was originally designed to investigate coordination control of unicycle mobile robots using a virtual structure approach [32]. The setup has proven to be very useful to test different control algorithms for a single or a group of unicycle mobile robots, see for example [4, 16, 17]. The e-puck mobile robot setup is shown in Figure 3.1 and consists of a 1.75 $\times$ 1.28 m arena. The mobile robot e-puck, shown in Figure 3.2, is a differential-drive unicycle-type robot developed at the EPFL, Switzerland.

The wheels of the e-puck are driven by stepper motors which are actuated by velocity control commands, i.e., the desired right and left wheel speeds, computed from the desired forward and steering velocity. Furthermore, the setup contains an external PC and an overhead camera which samples at approximately 30 Hz and is directly coupled to the PC by a firewire cable. The overhead camera is used for localization and detects the e-pucks from their unique markers as can be seen in Figure 3.2, which are originally designed in [7]. One marker is used to calibrate the relative position between the camera and the arena. All other markers are unique and directly coupled to a vehicle. Using an image processing algorithm, the relative position of the vehicles can be obtained with respect to the coordinate system, set by the calibration marker. If the calibration marker is not seen by the camera, the last known position of the marker is used. Using the overhead camera and the image processing algorithm, we are able to obtain the position with an accuracy of 0.0019 m in x- and y-direction according to [32]. The measured orientation is far from accurate with an accuracy of 0.0524 rad. It must be noted that the orientation angle cannot be determined when the vehicle is in standstill. Additionally the measurements of the angle are biased and contain significant noise. Accurate results can therefore not be guaranteed when the measured angle is used. This motivates for the use of only position measurements to obtain the orientation of the vehicle. All control is conducted on an external PC since the onboard processor of the e-pucks does not have the computational power to run this locally. The PC is additionally used to process the images from the camera into position measurements and is able to communicate the desired inputs to the e-pucks

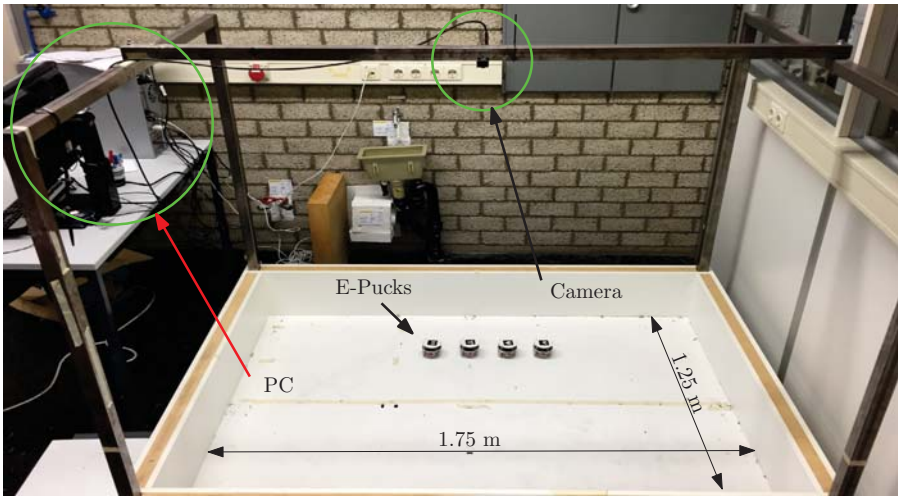

Figure 3.1: The mobile robot experimental setup with the e-pucks in the arena, the external PC, and the camera attached to the frame to measure the position of the vehicles.

using Bluetooth communication. The ability to communicate in such a manner, allows a variety of programming languages for control. In this case MATLAB is used to generate control inputs based on measurements and additional algorithms. The control structure is further discussed in the next section.

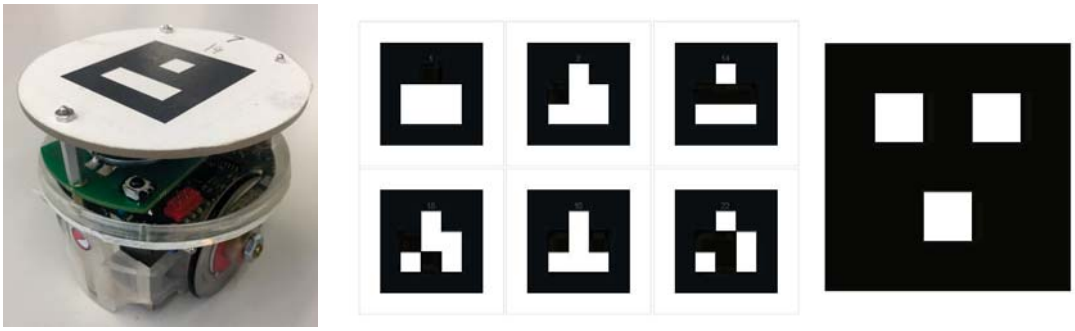

Figure 3.2: The left picture shows an example of an e-puck mobile robot with a marker on top. The right picture shows examples of the markers where the enlarged marker is used for calibration [7].

3.1.2 Control structure

The control structure for the use of the experimental setup is redesigned multiple times for specific projects. However, the guidelines remain mainly unchanged. An overview of the global control structure is presented in Figure 3.3. The camera provides images which are processed by an algorithm on the external PC. This results in position and orientation measurements, which are further processed. The design space incorporates all adaptations and necessary algorithms to determine the correct velocities. These are eventually converted to actual motor commands. It must be clear that the design space is the part where we are free to implement controllers, observers and additional algorithms. Since the measurements are corrupted with noise, and not all states are measured accurately, the states are estimated by a four dimensional observer as discussed in Chapter 2. The position, orientation and curvature of the path are required as a function of the travelled distance, which allows to reconstruct the trajectory of the vehicles. This data history is used to generate a reference point for each individual vehicle for which the controller determines the new velocity commands based on the current position of the vehicles. The next part elaborates on the mentioned events.

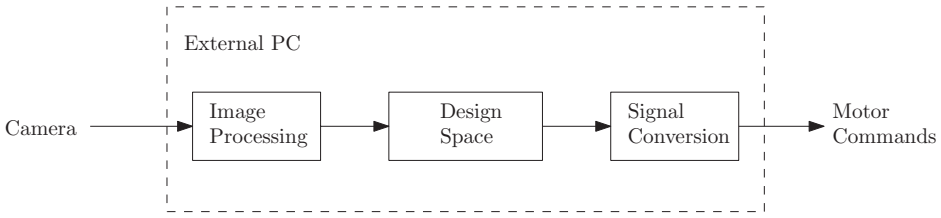

Figure 3.3: Overview of the global control structure within the PC and the interaction with the overhead camera and the e-puck mobile robot.

3.1.3 Measurement noise

For the purpose of realistic simulations, it is desired to take the measurement noise into account. Therefore, the standard deviation of the measurement noise needs to be determined. In [32] it was claimed that camera is able to measure with an accuracy of 0.0019 m in x- and y-direction. However, for completeness an analysis of the measurement noise is provided here. Therefore, the vehicles are positioned on four characteristic positions being approximately the corner points of a rectangle which covers one quarter of the experimental area as shown in Figure 3.4. Position A is located directly below the overhead camera and is therefore expected to be measured the most accurate. Position B is close to the border of the experimental area in y-direction. Position C and D are both located close to the border and far from the camera. The positions were measured for 50 seconds, giving over 1400 measurement points. Table 3.1 shows the corresponding standard deviation of the four vehicles in x- and y-direction. From this we can conclude that the measurements at the edge of the experimental area are poor due to a bended view of the overhead camera. This is especially indicated by the standard deviation at position C where the overhead camera is not able to accurately determine the position. When the vehicles are initially positioned this must be taken into account. The standard deviation of the measurements at position B and D are taken as a measure for the purpose of simulations. Those positions are the most representative, since it is expected that the vehicles approach these positions when following a reference trajectory.

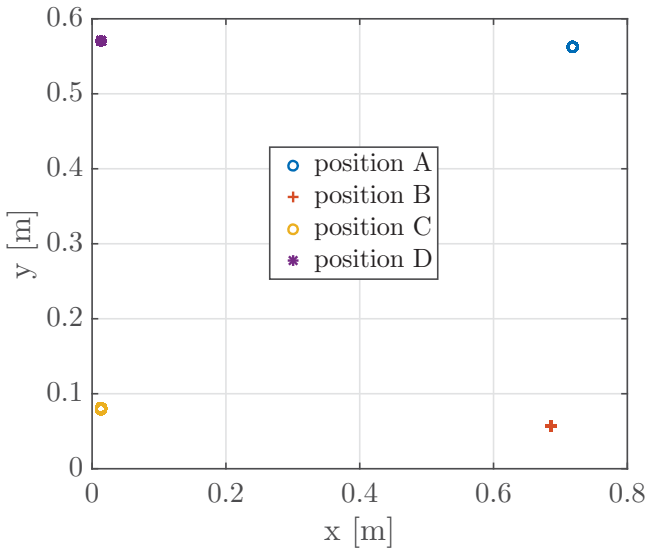

Figure 3.4: Positions of the e-puck mobile robots for which accuracy of the position measurements is determined. Position A is located underneath the overhead camera, where position C is located in the corner of the experimental area.

Before we continue to discuss the structure of the simulation model, we first consider the limitations of the experimental setup.

Table 3.1: Standard deviation of the measurements at the four positions A,B,C and D

Position	x	y	unit
A	0.0100e-3	0.0111e-3	m
B	0.0208e-3	0.0097e-3	m
C	0.1010e-3	0.1616e-3	m
D	0.0127e-3	0.0322e-3	m

3.1.4 Limitations

In practice, the sampling time is limited by the framerate of the camera system, which is approximately 30 Hz, but non-constant. Another limitation is observed at the edge of the arena caused by the camera which captures a more bended image and therefore gives less accurate position measurements. Moreover, the measurement data contains noise and might be biased. This was already discussed and a state estimator is proposed to overcome this problem. It must be noted that the biased orientation angle can be corrected manually by analysis of the e-pucks in steady state. The disadvantage is that this must be done for each individual vehicle, which is undesired. So far the assumption has been made that the states of the vehicles are continuously available. This is not the case since the states are measured in a discretized manner. Additionally, an external PC is used to process the data and communicates the new input command over a Bluetooth connection, which increases the possibility of delays. From the provided Bluetooth dongle it is known that it can only handle up to 7 mobile robots and induces a delay in sending of the control signals, which escalates with an increasing number of robots. Delays also arise due to computation time of the PC, where there is another small delay in getting the necessary signals. The delay is non-constant, since the image processing and computation time of the PC are varying. However, variations are small and therefore an approximation of the delay is considered. Identifying experiments show a delay between sending the motor commands and actual movement of the wheel of approximately 4 to 6 samples, shown in Appendix C. As mentioned, the delay is depending on several aspects: Bluetooth connection, the number of vehicles, computation time of the control signals and image processing. Additionally, the batteries have a significant influence on the performance of the e-pucks. When the batteries are low, the voltage output drops and less torque is provided to the stepper motors. Stepper motors are known for their precise positioning and speed control at low speeds. They have maximum torque at low speeds, but have less torque at high speeds. There is no further feedback from the stepper motors, which required additional sensors for reference tracking. An open loop experiment is executed to investigate if the actual velocities are close to the input velocities. The result for an open loop experiment with $v = 0.08$ and $\omega = 0.2$ are depicted in Figure 3.5. It can be concluded that there is a slight difference in the input command and the actual velocities. However, this is of no concern in closed loop situations.
The e-puck has a maximum forward velocity of approximately 0.13 m/s, which approximately equals 1024 steps in each motor. The reference forward velocity is always set below the maximum value to obtain accurate results. Despite the limitations, the e-puck mobile platform is a very useful setup to test the control algorithm proposed by Lefeber et al. [18] and to investigate limitations of the setup and further improvements. Nevertheless, the limitations cannot be neglected and must be incorporated in the simulations. Furthermore, calculation times need to be minimal.

3.2 Numerical model

In the previous section, a general overview is given for the control structure of the experimental setup and hardware limitations are addressed. These practical issues are taken into account to obtain a realistic simulation environment. In simulations, the inputs are not send to an e-puck. Therefore, the motion of the e-puck mobile robot is described with a kinematic model. An overview of the closed loop model is depicted in Figure 3.6. Additionally the input signals provided from the controller are saturated to meet the restrictions of the mobile robot. In order to analyze the behaviour of the vehicles when measurement data corrupted by noise, zero mean white noise is added to the position determined from the kinematic model. This section further elaborates on the elements shown in Figure 3.6, and the adjustments that need to be made.

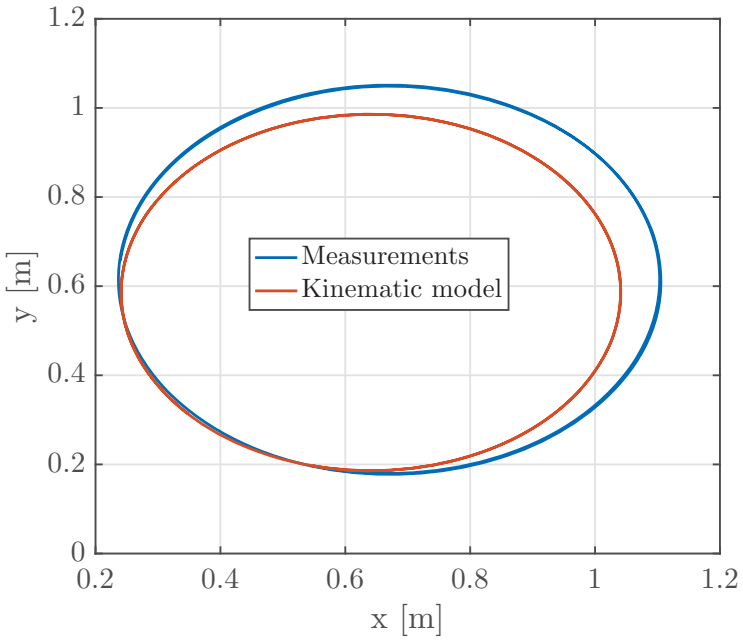

Figure 3.5: Open loop experiment, with $v = 0.08$, $\omega = 0.02$ and initial position (0.55,0.2) m, compared to the expected path as derived from the kinematic model.

3.2.1 Exact discrete time kinematic model

The continuous time kinematic relations of a mobile robot are well known and already given in (2.1). However, for the ease of exposition, the variables are changed, resulting in

$$\begin{aligned} \dot{x}_1 &= u_1 \cos x_3 \\ \dot{x}_2 &= u_1 \sin x_3 \\ \dot{x}_3 &= u_2, \end{aligned} \tag{3.1}$$

where (x_1, x_2) is the position in (x, y)-direction and x_3 is the orientation angle. The input signals u_1 and u_2 represent the forward velocity and the angular velocity respectively. The time interval between two sampling instances is defined as:

$$t(k) = t \in [kh, kh + h) \tag{3.2}$$

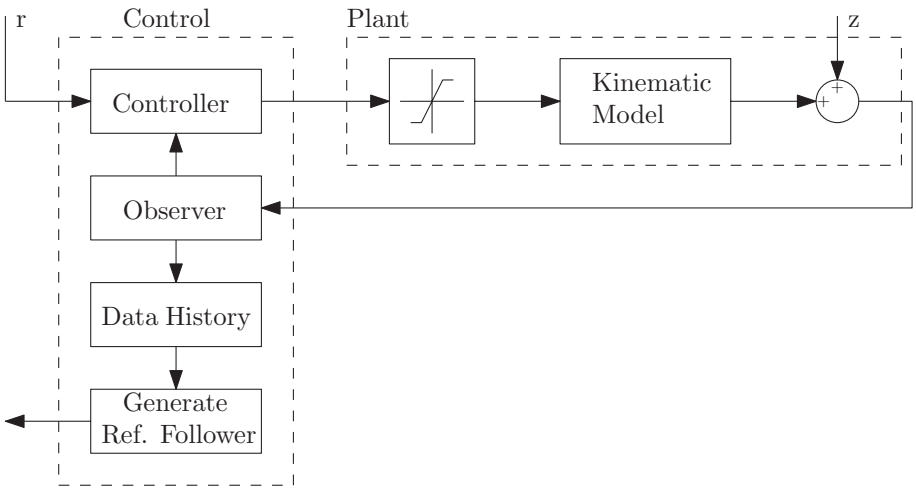

Figure 3.6: Schematic representation of the control structure in a simulation environment.

with sampling period h. For the discretization process it is assumed that we have zero-order-hold (ZOH) for the inputs u_1 and u_2, i.e., $u(t) = u(t(kh))$ is constant for $t \in [t(kh), t(kh+h))$. The discretized model is given by:

$$\begin{bmatrix} x_1(kh+h) \\ x_2(kh+h) \\ x_3(kh+h) \end{bmatrix} = \begin{bmatrix} x_1(kh) \\ x_2(kh) \\ x_3(kh) \end{bmatrix} + \int_t^{kh} \begin{bmatrix} \cos x_3(\tau) & 0 \\ \sin x_3(\tau) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_1(kh) \\ u_2(kh) \end{bmatrix} d\tau. \tag{3.3}$$

The exact discretized time model is obtained from solving the integral, formulated as:

$$\begin{bmatrix} x_1(kh+h) \\ x_2(kh+h) \\ x_3(kh+h) \end{bmatrix} = \begin{bmatrix} x_1(kh) + 2u_1(kh)\phi(u_2(kh)) \cos\gamma(x_3(kh), u_2(kh)) \\ x_2(kh) + 2u_1(kh)\phi(u_2(kh)) \sin\gamma(x_3(kh), u_2(kh)) \\ x_3(kh) + u_2(kh)h \end{bmatrix}. \tag{3.4}$$

where,

$$\phi(u_2(k)) = \begin{cases} \frac{\sin(\frac{h}{2}u_2)}{u_2} & \text{if } u_2 \neq 0 \\ \frac{h}{2} & \text{if } u_2 = 0, \end{cases}, \qquad \gamma(x_3(k), u_2) = x_3 + \frac{h}{2}u_2. \tag{3.5}$$

The exact discretized time model as derived here is now further used in simulations.

3.2.2 Discrete time observer

Before implementation of the observer in a real-time environment, the continuous time observer needs to be discretized. The position measurements are available at each sampling time $t(k)$, $k \in \mathbb{N}$, where the sampling time is set by the framerate of the camera. Moreover, it is known that there are approximately 4–6 samples delay between sending the input signals and actual movement of the vehicle. However, the position measurement from the camera are not delayed. To prevent a mismatch between the estimated values and the measured values, the observer input is delayed by 6 samples. This can be taken into account for the observer proposed in the previous chapter. First the continuous time observer as proposed in [12] is discretized. Recall the differential equations of the observer dynamics:

$$\begin{aligned} \dot{\hat{x}}_1(t) &= u_1(t)\hat{c}(t) + l_1\tilde{x}_1 \\ \dot{\hat{x}}_2(t) &= u_1(t)\hat{s}(t) + l_2\tilde{x}_2 \\ \dot{\hat{c}}(t) &= -u_2(t)\hat{s}(t) + l_3u_1(t)\tilde{x}_1 \\ \dot{\hat{s}}(t) &= u_2(t)\hat{c}(t) + l_4u_1(t)\tilde{x}_2 \end{aligned} \tag{3.6}$$

where, $\tilde{x}_1(t) = x_1(t) - \hat{x}_1(t)$ and $\tilde{x}_2(t) = x_2(t) - \hat{x}_2(t)$. Again the inputs are assumed to be constant between two sampling instances. However, $\tilde{x}_1$, $\tilde{x}_2$, $\hat{c}$, and $\hat{s}$ do not change over time as well. When evaluated at the final time instants and taking into account the six samples delay, this results for the observer in:

$$\begin{bmatrix} \hat{x}_1(kh+h) \\ \hat{x}_2(kh+h) \\ \hat{c}(kh+h) \\ \hat{s}(kh+h) \end{bmatrix} = \begin{bmatrix} \hat{x}_1(kh) + hu_1(kh-6h)\hat{c}(kh) + hl_1\tilde{x}_1(kh) \\ \hat{x}_2(kh) + hu_1(kh-6h)\hat{s}(kh) + hl_2\tilde{x}_2(kh) \\ \hat{c}(kh) - hu_2(kh-6h)\hat{s}(kh) + hl_3u_1(kh-6h)\tilde{x}_1(kh) \\ \hat{s}(kh) - hu_2(kh-6h)\hat{c}(kh) + hl_4u_1(kh-6h)\tilde{x}_2(kh) \end{bmatrix}, \tag{3.7}$$

where, l_1, l_2, l_3, and l_4 are still to be determined. The orientation angle can be determined from

$$\hat{x}_3(kh) = \text{atan2}(\hat{s}(kh), \hat{c}(kh)). \tag{3.8}$$

As shown in Chapter 2, the estimated orientation angle converges to the actual orientation error when $|\hat{x}_3(0) - x_3(0)| < \frac{\pi}{2}$. The convergence to the actual orientation angle is further discussed in the next chapter.

3.2.3 Master reference

The first vehicle needs to generate a path, which must fit within the bounds of the experimental area and has to show the advantages and performance of the control algorithm. Therefore, a master reference is introduced which is only followed by the first vehicle. This enables to include the acceleration and deceleration of the master reference vehicle to show effectiveness of the longitudinal controller of [18]. The master reference path is shown in Figure 3.7, which includes the transition of a straight line to a circular path and vice versa. If corners are cut, this is immediately visible, which motivates for the chosen trajectory.

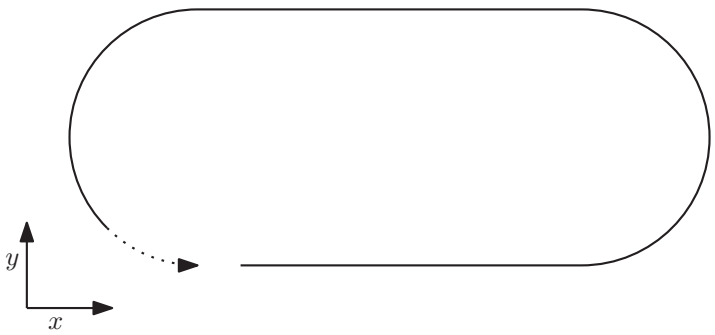

Figure 3.7: Preferred reference trajectory for the first vehicle.

The first vehicle needs to track the proposed master reference, and is therefore equipped with the tracking controller as designed by Jiang and Nijmeijer [13]. However, the tracking controller of [13] was originally designed for the velocity inputs v and ω. Since the longitudinal controller of [18] uses the acceleration as input for the following vehicles the choice is made to rewrite the tracking controller. Consider once more the kinematic model of a mobile car where the model is extended with $\dot{v} = a$. Using Lyapunov analysis the following control laws for the angular velocity and acceleration are obtained:

$$\omega = \omega_r - c_2 v_r x_e \frac{1-\cos\theta_e}{\theta_e} + c_2 v_r y_e \frac{\sin\theta_e}{\theta_e} + c_3\theta_e \tag{3.9}$$

$$a = a_r + c_1 x_e + c_4 v_e, \tag{3.10}$$

where c_1, c_2, c_3, and $c_4 > 0$ are the control gains and x_e, y_e and θ_e are the error coordinates as defined in Chapter 2, and $v_e = v_r - v$. The complete proof that the errors converge to zero is given in Appendix B. Furthermore, $\frac{\sin\theta_e}{\theta_e} = 1 - \frac{1}{6}\theta_e$, and $\frac{1-\cos(\theta_e)}{\theta_e} = \frac{\theta_e}{2} + \frac{\theta_e^3}{24} + \frac{\theta_e^5}{720}$, if $\theta_e = 0$. In [13] it is shown that the error dynamics converge to zero under the assumption that either $v_r(t)$ or $\omega_r(t)$ does not converge to zero. The control algorithm is able to follow a straight line, and has control gains which are easy to tune, making it a very suitable controller for the purpose.

3.2.4 Virtual reference

From Chapter 2 it is known that the path of the virtual reference vehicle is characterised by differential equations as in (2.23). However, these cannot be used to determine the positions of the virtual reference vehicle since there is no feedback on the obtained position. This implies that if the initial position is not on the path of the leader, or if the communicated input velocities are not exact, that virtual reference is no longer located on the path of the leader. Since this is undesired, the virtual reference position is generated from the saved data of the path of the leader to guarantee that it is on the path of the leader. The properties of the path of the leading vehicle, $x_l(s_l(t))$, $y_l(s_l(t))$, $\theta_l(s_l(t))$, $\kappa_l(s_l(t))$ are saved as function of the travelled distance $s_l(t)$. It should be noted that the estimated position and orientation are saved, since the measured information is unreliable. Moreover, the travelled distance of the vehicle is not measured directly and cannot be determined based on odometry. The travelled distance of the vehicle is therefore determined from position measurements, following

$$s(k) = s(k-1) + \sqrt{(\hat{x}(k) - \hat{x}(k-1))^2 + (\hat{y}(k) - \hat{y}(k-1))^2}. \tag{3.11}$$

Another possibility is to derive the travelled distance from direct integration of the velocity. However, since both variables are uncertain, the position based approach becomes the most reliable source. However, when the distance between the measurements increases, interpolation between points is necessary.

The following vehicle is driven towards a virtual reference vehicle. The position and orientation of this virtual reference vehicle are originally defined as a function of the travelled distance of the following vehicle along the path of the leading vehicle, denoted by $\alpha(s)$. The following applies $\bar{v}(s) = \frac{d\alpha(s)}{ds}$ and $\bar{x}_l(s) = x_l(\alpha(s))$, $\bar{y}_l(s) = y_l(\alpha(s))$, $\bar{\theta}_l(s) = \theta_l(\alpha(s))$, and $\bar{\kappa}_l(s) = \kappa_l(\alpha(s))$. However, since we are implementing the controller in real time, the time based equivalents are required as derived in Chapter 4, defined as

$$\begin{aligned} \bar{x}_l(t) &= \bar{v}(t)v(t)\cos\bar{\theta}_l(t) \\ \bar{y}_l(t) &= \bar{v}(t)v(t)\sin\bar{\theta}_l(t) \\ \bar{\theta}_l(t) &= \bar{v}(t)v(t)\bar{\kappa}_l(t). \end{aligned} \tag{3.12}$$

To define the position of the virtual reference vehicle, the travelled distance along the path of the leading vehicle is required, which can be obtained from

$$\alpha(s(t)) = \int_0^t \bar{v}(\tau)v(\tau)d\tau \tag{3.13}$$

or in discrete time

$$\alpha(s(kh+h)) = \alpha(s(kh)) + \bar{v}(kh)v(kh)h. \tag{3.14}$$

Using the historic data of the leader, one can find the corresponding position of the virtual reference vehicle following

$$(\bar{x}(s), \bar{y}(s), \bar{\theta}(s), \bar{\kappa}(s)) = \{(x_l(s_l^*), y_l(s_l^*), \theta_l(s_l^*), \kappa_l(s_l^*)) : s_l^* \in \text{argmin}|s_l - \alpha(s)|\} \tag{3.15}$$

Note, that this is only allowed when the velocity of the vehicles is low and relatively many data points are provided by the sensors. If the velocity increases, this increases the distance between two data points. Interpolation can be used to fit the intermediate points. Since driving at maximum speed increases the possibility of uncertainties, such as slip, and does not allow the vehicle to further accelerate, a forward velocity of 0.04 m/s is chosen as base speed for the e-pucks. Further, the sample rate of the camera is around 30Hz, which provides sufficient data points in combination with the chosen velocity such that interpolation might not be necessary. Since the implementation must be low cost, the interpolation does not take an infinite amount of points.

3.3 Concluding remarks

In this chapter, a short overview of the experimental setup is given. The hardware components are discussed and we elaborated on the limitations. Furthermore, the control structure is shown and adaptations are made to use the control structure in both a simulation and experimental environment. Furthermore, it was shown that the accuracy of the camera varies when the vehicles are positioned at four crucial positions in the experimental area. The amount of measurement noise must be taken into account to obtain a simulation environment that is close to the experimental environment.

The exact kinematic model is derived, which is used in simulations. In practice, the measurements, used by the observer to obtain estimates, are sampled and the input is delayed. In order to improve the estimated position and orientation, the delay is taken into account in the discretized observer design. Furthermore, the reference for the first vehicle is discussed and an adaptation to the tracking controller of Jiang and Nijmeijer [13] is shown. As a final remark, it has been shown how the position of the virtual reference vehicle is determined from the saved data. The next chapter elaborates on the performance of the proposed methods and the performance of the controller is shown through a numerical and experimental analysis.

Chapter 4

Simulations and experiments

In the previous chapters the theoretical approach is adapted for a practical implementation. In this chapter, the controller is first implemented in the simulation environment MATLAB. This allows to determine the control parameters properly for the available experimental setup. Additionally, this allow to investigate the performance and the behavior of the vehicles in case of uncertainties. Next, to check the validity of the proposed assumptions and controller design, the controller is implemented on the e-puck mobile robot. The advantage of using MATLAB is that it can be directly used for the experimental environment to determine the new velocity inputs. This combination of simulation and experimental results provides valuable insights in the performance of the controller and the behavior of the vehicle in case of sensor noise, delays, and model uncertainties. The results of the real time experiment are presented afterwards and can be compared to the simulation results from which conclusions and recommendations are drawn.

4.1 Simulations

The main objective of the simulations is to show the performance of the discretized controller and observer. Additionally, the choice of control gains is motivated. First, the control gains as proposed by Lefeber et al. [18] are discussed by means of simulation. This allows to investigated the characteristics of the path in more detail. The simulation environment takes into account sensor noise and a varying sampling time, to obtain a simulation environment that is a close representation of the experimental setup. The results are therefore a good indication for the performance in actual experiments. Second, the performance of the observer is discussed, where we consider the presence and absence of measurement noise. The simulation part ends with a simulation, where four vehicles participate in a platoon, and where measurement noise and a varying sampling time are taken into account. This provides a first impression of the performance of the vehicles in a real-time experiment.

4.1.1 Controller parameters

In Chapter 2, the lateral and longitudinal controller, as well as the observer design are presented. The control gains, as used for simulation of the lateral and longitudinal controller, are adopted from Lefeber et al. [18] and given by

$$c_1\sigma_1(x_e) = 0.99\text{sat}\left(\frac{2}{0.99}x_e\right) \qquad c_2\sigma_2(\theta_e) = 4\text{sat}(\theta_e) \qquad c_3 = 4,$$

where

$$\text{sat}(x) = \begin{cases} -1 & \text{for} \quad x \leq -1 \\ x & \text{for} \quad -1 \leq x \leq 1. \\ 1 & \text{for} \quad 1 \leq x \end{cases}$$

These are obtained by placing the poles of the linearization of the error dynamics for a straight line at -2. The controller and the corresponding gains are initially designed for full scale highway vehicles. This

can be retrieved from the choice of the saturation values, which are corresponding to saturations when the longitudinal error is larger than 1 m. Since the e-puck mobile robot setup works on a significantly smaller scale, the errors are smaller and therefore the proposed controller gains need to be adapted. To obtain a desired convergence to the path of the leader, a gain or scaling vector is applied to the position errors to influence the effectiveness of the controller. This gain is determined in an iterative manner and is equal for all vehicles. The choice for a proper gain is based on the available experimental area and the desired trajectory. As an extra criteria, it is required to be on the desired path before the vehicles start to turn. This is essential since we are mainly interested in the effect of corner cutting. A gain $K = 10$ is considered, since this shows a sufficient convergence to the desired path and is able to eliminate the lateral error before the vehicles start to turn. The convergence of the follower to its reference path is shown for $K = 10$ in Figure 4.1. A higher gain results in a faster convergence towards the desired path. However, we need to consider that the follower becomes the leader for the next vehicle when multiple vehicle participate in the platoon. With this in mind we can reconsider Figure 4.1. Special attention is given to the case where the follower, initially positioned at (0.3,0.3) m, starts with an orientation angle close to zero. The artificial path of the leader, necessary to overcome the problem of a negative initial error, is based on the initial orientation of the vehicles. This results in a horizontal artificial path of the leader, to be tracked by the follower. As the leading vehicle starts to drive, it immediately reacts to the presence of a lateral error, and turns fast towards the desired path. This results in a transition from the artificial path to the actual generated path which is non-smooth. The virtual reference vehicle for the follower moves along the artificial horizontal path until it reaches the initial position of the leader at (0.4,0.2) m and continues on the path generated by the leader. A closer look suggest that the follower moves away from the reference. However, the 2-norm of the position error $||e_{pos}||_2$ shows that the error is monotonically decreasing, which clarifies that the follower does not move away from the path.

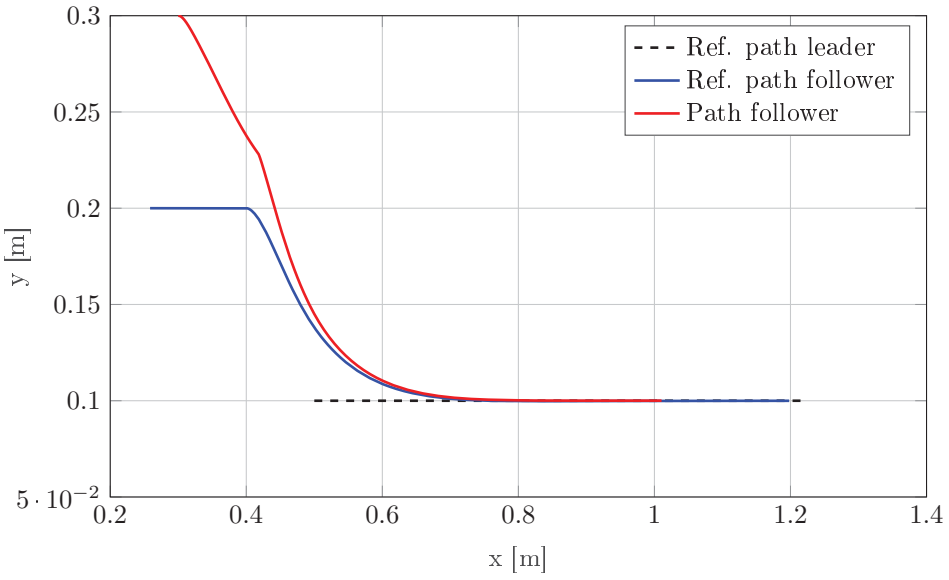

Figure 4.1: The convergence of the follower towards the reference path is shown here from an initial position (0.3,0.3) m. The reference path is generated by the leader from position (0.4,0.2) m and onwards, and is extended with an artificial horizontal path to the left. Note that there is a non-smooth convergence due to this reference path.

So far, the choice of the controller gains for the first vehicle are not discussed. The only requirement for the controller is that the first vehicle is on the desired trajectory before the first turn. The controller of Jiang and Nijmeijer [13] was extended and considers the inputs a and ω as shown in Appendix B. This leaves four gains to tune resulting in the values that are shown in Table 4.1. The gains are tuned manually and result in satisfactory behavior of the first vehicle. The resulting path is shown in Figure 4.1 as the reference path of the follower, indicated in blue. It can be seen that the vehicle smoothly converges towards the desired reference path within 0.4 m.

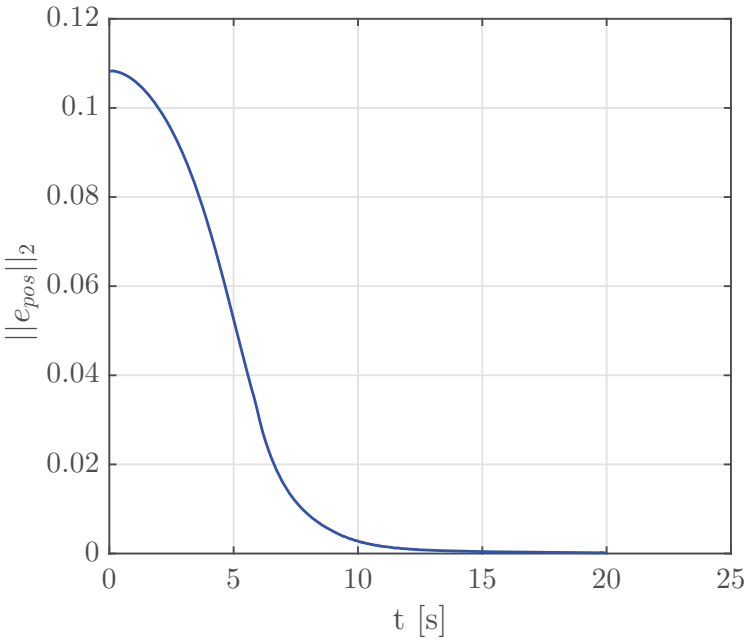

Figure 4.2: Norm of the position error which shows a monotonic decrease.

Table 4.1: Control parameters used in simulation and experiments

Parameter	Description	Value
K	Position error gain	10
c_1	Position gain	1
c_2	Position gain	600
c_3	Orientation gain	1
c_4	Velocity gain	1

4.1.2 Observer gains

In Chapter 2 a full state observer is proposed which estimates the position and orientation of the vehicles. The observer gains are shown in Table 4.2, which are used in simulations to show the convergence of the estimates to the actual values of the position and the orientation angle.

Table 4.2: Control parameter used in simulation and experiments

Parameter	Description	Value
l_1	Observer gain	10
l_2	Observer gain	10
l_3	Observer gain	1000
l_4	Observer gain	1000

Figure 4.3 shows the difference between the measured and estimated position and orientation, with and without the presence of noise. First, a simulation is executed without the presence of noise, indicated in red with initial conditions $\hat{x} = 0$, $\hat{y} = 0$, $\hat{c} = 1$ and $\hat{s} = 0$. As can be seen, the correct x- and y-position are achieved at respectively 5 and 10 seconds. It takes approximately 25 seconds to estimate the exact orientation angle, which is sufficient for the use in real-time experiments. These experiments take on average up to 80 seconds to complete one trajectory. The second simulation considers the presence of noise on the position measurements. As shown in the previous chapter, the worst case scenario for the accuracy of the measurements has a standard deviation of 0.03e-3 m. Therefore, normally distributed noise is added to the values of the x- and y- position with the worst case standard deviation. This significantly affects the orientation estimation. Nevertheless, the error remains within an acceptable range. Special attention is given to the estimation of the orientation angle in Figure 4.4. The orientation θ is determined from unfiltered position measurements using atan2(y, x), indicated in blue. These position

measurements are obtained from a vehicle which accelerates in 4 seconds towards a constant speed of 0.04 m/s . It can be seen that observer is able to provide a better estimate of the orientation angle, $\hat{\theta}$, which contains significantly less noise.

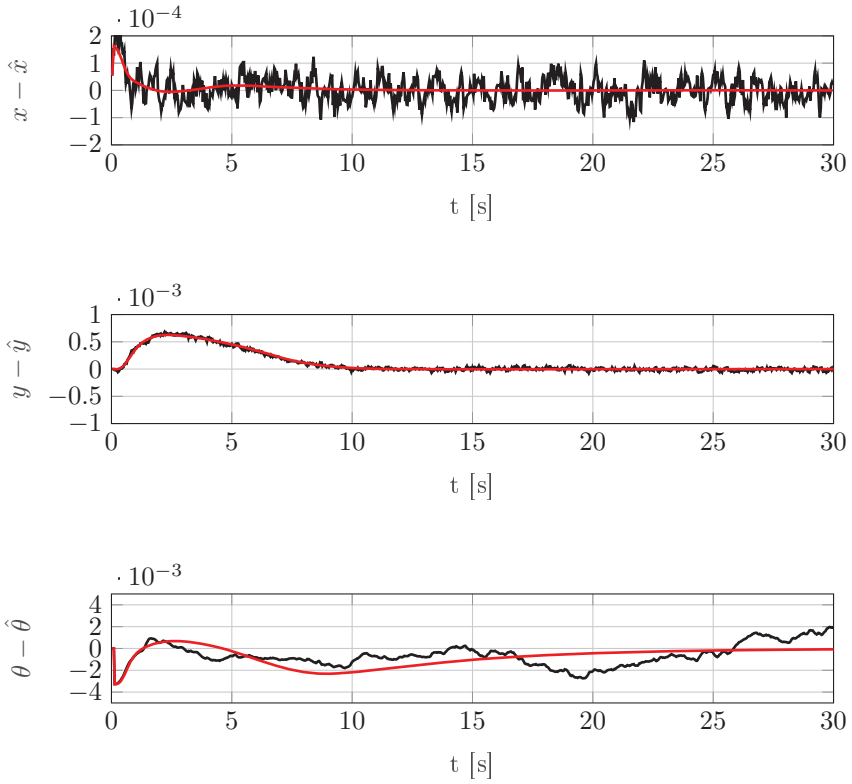

Figure 4.3: Difference between the measured and estimated position and orientation in the presence of noise on the position measurements and without, indicated by the black and red line respectively.

4.1.3 Simulation results

As a final step, a complete simulation is shown, which includes noise in x- and y-measurements, and a varying sampling time. This simulation is divided into several parts for clarity. First, the lateral performance is shown where special attention is given to the beginning of the simulation. Another point of interest is the transition from driving in a straight line to a curvilinear path. The longitudinal control is discussed separately, and investigates the behavior of the vehicles at the start and when the first vehicle accelerates and decelerates.

In order to investigate the performance of the lateral and longitudinal controller, we initially consider a simple trajectory of a repetitive combination of a straight line of 0.4 meters and a semi circle with a radius of 0.4 meters. The first vehicle follows a predefined reference, which is initially positioned at $(x, y) = (0.5,0.1)$ m. This reference vehicle accelerates with 0.01 m/s^2 for 4 seconds to reach a velocity of 0.04 m/s, after which the velocity remains constant. After $t = 40$ s, the reference vehicle starts to accelerate again with 0.01 m/s^2 until $t = 42$ s. After 42 seconds, it drives with a constant velocity. From $t = 48$ to $t = 50$ s this vehicle decelerates with a = -0.01 m/s^2. The first vehicle is controlled by the tracking controller of Jiang and Nijmeijer [13] and is required to minimize the pose and velocity errors with respect to the reference vehicle. It is very important to note that only the first vehicle is controlled by this controller, serving as a reference for the follower.

Consider a platoon of four vehicles where the leader starts at $x = 0.4$ m , $y = 0.1$ m, $\theta = 0$ rad, tracking a reference starting from $x = 0.5$ m , $y = 0.1$ m, $\theta = 0$ rad. The other vehicles start at respectively $(x, y, \theta) = (0.3, 0.2, 0), (0.2, 0.3, 0)$, and $(0.4, 0.1, 0)$. All vehicles are initiated with $v = 0$ m/s. Additionally, a standstill distance is chosen to be $r = 0.1$ m and a time gap $h = 2$ s. Since the velocity is on average $v = 0.04$ m/s this results in an inter-vehicle distance of 0.18 m. The sampling time

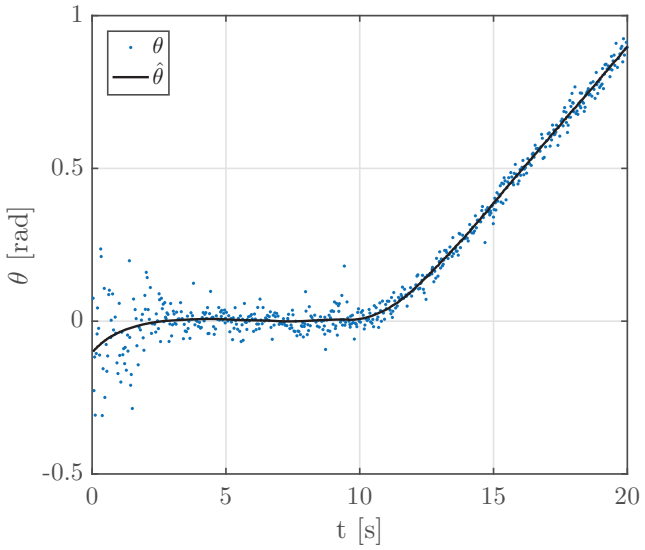

Figure 4.4: Orientation angle θ obtained directly from position measurements and the estimated orientation angle $\hat{\theta}$ by the observer.

is set at $t_s = 1/30s$ with random white noise with a variance of approximately 8e-5 s.

4.1.4 Lateral performance

The resulting paths for the four vehicles is depicted in Figure 4.5. On first sight, the vehicles converge perfectly to the path of the predecessor. However, this cannot be concluded immediately due to the large scale. Therefore, we illustrate the overall performance by the 2-norm of the position error later on. However, first the part is studied until the vehicles are on the desired path. Also the transition from a straight line to a curvilinear path is studied to see if corners are cut. Both cases are shown in Figure 4.6. Consider first the path of vehicles from the initial position towards the path of the preceding vehicle. Again a cornering manoeuvre is visible when the virtual reference vehicles make a transition from the horizontal artificial horizontal path to the actually driven path, as explained before. All vehicles reach the path of the predecessor significantly fast and are on this path when the vehicles start to turn. This is shown in the right plot of Figure 4.6, which is a closer view on the transition point. The paths are not indistinguishable and have a slight deviation. This is mainly caused by the sampling time, which is varying and not infinity small.

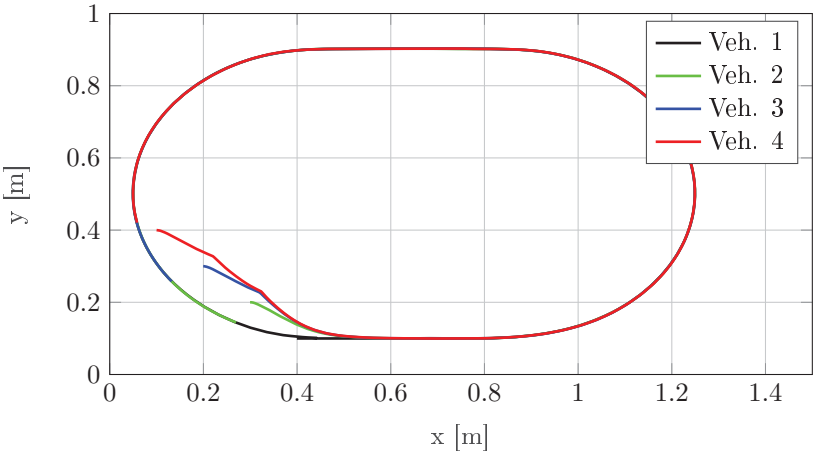

Figure 4.5: The resulting paths of the vehicles as seen from above.

As a measure for the overall performance, the 2-norm of the position error is shown in Figure 4.7. The norm of the position error for vehicle 2, indicated in blue, is taken with respect to the virtual reference

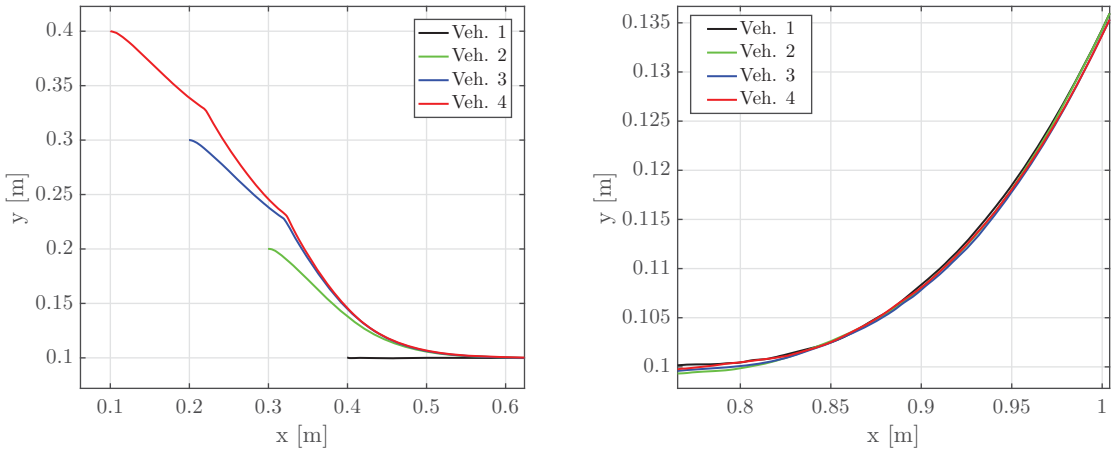

Figure 4.6: The left figure shows the paths of the vehicles from their initial position, $(x, y, \theta) = (0.4, 0.1, 0), (0.3, 0.2, 0), (0.2, 0.3, 0)$, and $(0.4, 0.1, 0)$ respectively. The paths confirm the expectation and are indistinguishable at (0.6,0.1). However, when the vehicle starts corning, shown in the right figure, a slight deviation between the vehicles is noticed.

vehicle. Similarly, this is done for vehicle 3 and 4 as well. The error does not completely converge to zero, but saturates at approximately 2e-3 m. Another point of interest is the noise that remains, which can be simply overcome by taking more intermediate points between two measurement points, providing a smoother curve. The position error is mainly limited by the sampling rate and cannot be further reduced with the current controller design and implementation method.

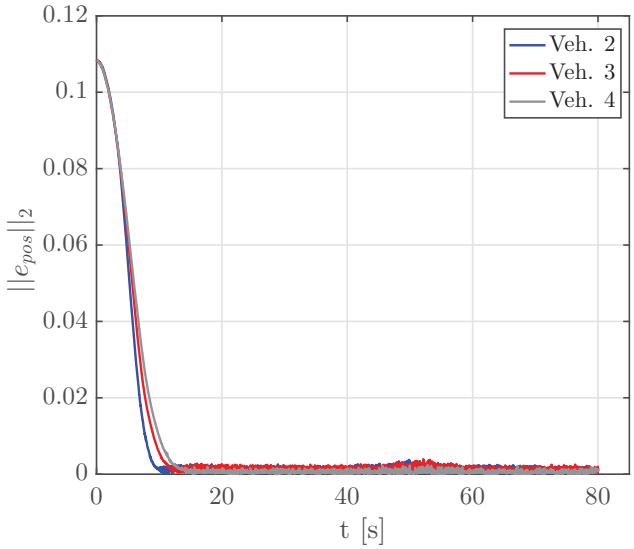

Figure 4.7: Norm of the position error for simulation of four vehicle in a platoon, including measurement noise and a limited varying sampling time.

4.1.5 Longitudinal performance

The performance of the longitudinal controller is evaluated along the trajectory. Here, the trajectory is again divided in two parts. Both events take place during the same simulation, but are separated for clarity. The first part is the beginning of the experiment where the vehicles have a large lateral and longitudinal error. The second part is the part where the first vehicle starts to accelerate and decelerate. Here, all vehicles are already on the path of their predecessor.

First consider the start of the experiment, where the vehicles start from standstill and slowly accelerate to the desired velocity. This acceleration and forward velocity is shown in Figure 4.8. The reference for the first vehicle is depicted as well which shows the performance of the longitudinal control for the first vehicle. The events between $t = 5$ s and $t = 10$ s, particularly stand out. Since the longitudinal controller incorporates the lateral behavior, it is no coincidence that the vehicles adapt their velocity before performing a sharp turn. When the vehicles are close to the path this behavior is damped, as the effect can slightly be seen around $t = 12$ s for the fourth vehicle. It can be concluded from Figure 4.8 that the noise on the input signal a has almost no effect on the velocity v, due to the relatively small change in acceleration. So far, string stability has not been addressed in this work. From [26] it is known that string stability of interconnected vehicles is characterized by the reduction of the velocity in downstream direction. From Figure 4.8 it can be seen that the velocity of the vehicles reduces downstream, implying string stability of the platoon. This is even more clear when the case is studied where the first vehicle accelerates and decelerates as depicted in Figure 4.9. Notice that the acceleration of the first vehicle contains significantly less noise. It is as expected that the behavior of the first vehicle is different from the other vehicles, since a different controller is implemented. The main difference between the two controllers is their definition: where the first vehicle is driven by a tracking controller, the other vehicles have a separated lateral and longitudinal controller with the focus on converging to the desired path. The noise is a result of the varying sampling time and the choice for using a minimum amount of points. Increasing the amount of intermediate points, obtained by interpolation, reduces the noise since reference points can be determined more accurately, which reduces the errors.

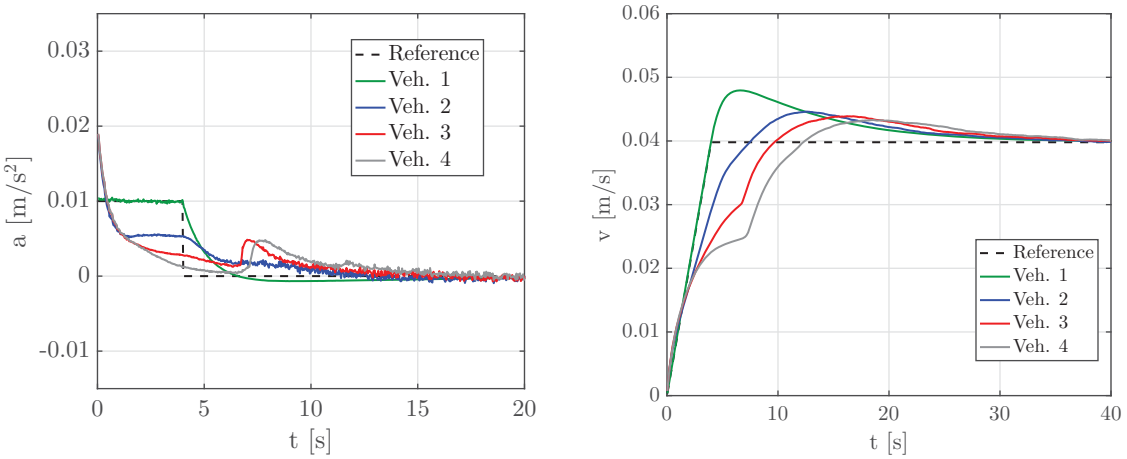

Figure 4.8: Acceleration and velocity with respect to time at the start of the simulation where the vehicles have initial errors in both lateral and longitudinal direction.

4.2 Experiments

So far, the controller has been considered in a simulation environment. We now present the experimental results to validate the assumptions that are made for simulations and to show effectiveness of the controller. Again the analysis is divided in several parts for clarity with the same structure as was done for the simulation results.

4.2.1 Lateral performance

In Figure 4.10 the resulting paths for a platoon of four e-puck mobile robots is depicted. As can be seen, the robots are positioned with both a lateral and longitudinal initial error at the left half of the experimental area.

The heading of the vehicles is approximately zero, which means that the added historic path is a horizontal line. The advantage of positioning the vehicles with an orientation angle of approximately zero, is that no additional experiment needs to be conducted to determine the orientation of the vehicles. A disadvantage is that the orientation angle is not precisely known, resulting in an artificial path which

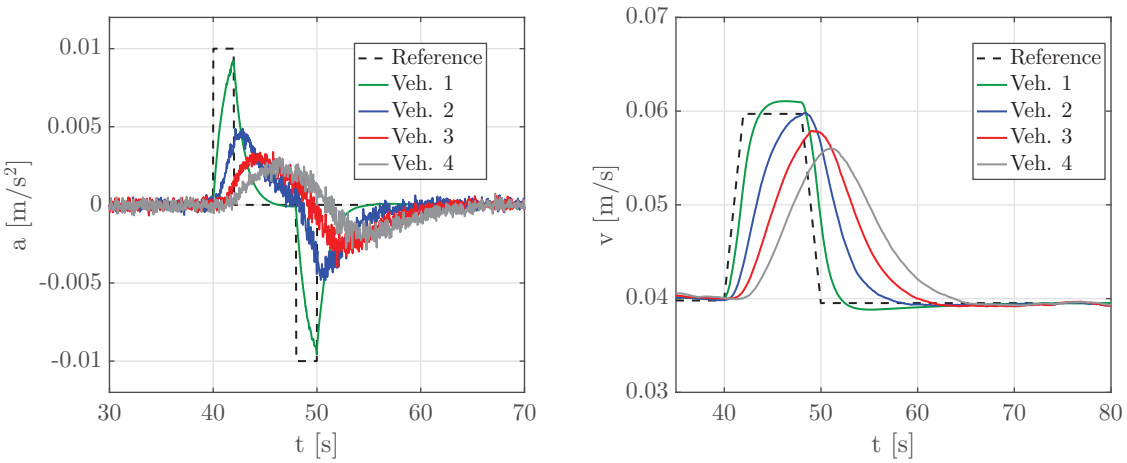

Figure 4.9: Acceleration and velocity with respect to time, where the first vehicle starts accelerates and decelerates. The vehicles are already on the desired path.

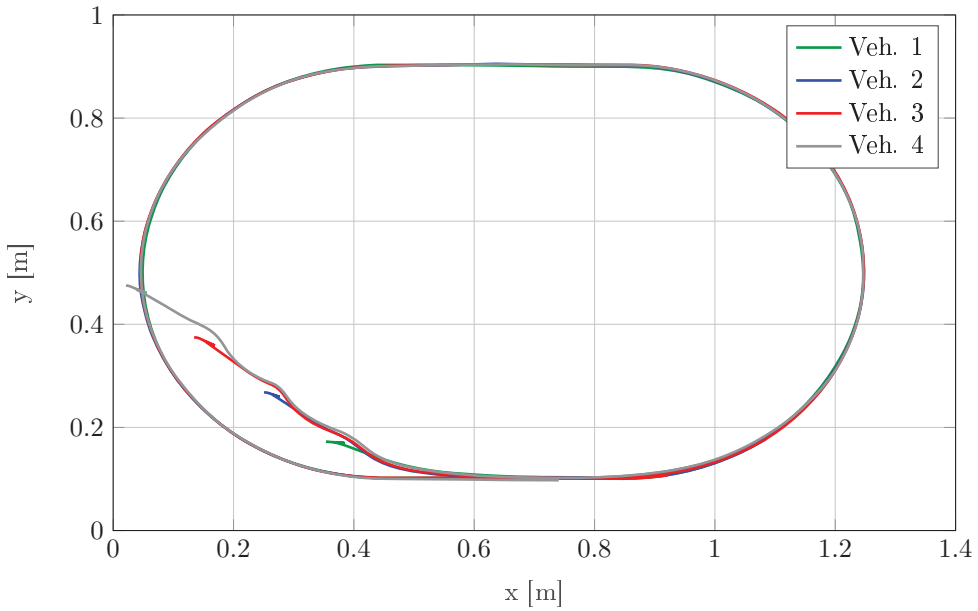

Figure 4.10: Resulting paths for a platoon of four vehicles.

is not exactly aligned with the vehicle. The master reference for the first vehicle remains unchanged. All vehicles start from standstill and move to the path of their predecessor as desired according to this figure. Nevertheless, these results give no clear indication of the precise lateral errors. Therefore, the start and first corner of the vehicles are shown in Figure 4.11.

The left figure shows the estimated path of the vehicles from the start of the experiment. A small deviation is visible at the beginning of each path, caused by an estimation error. It appears that at this specific point, the sampling time significantly increased for only one sample period, which results in an inaccurate estimation of the position. This significantly higher sampling time appears in every experiment after approximately 1 second and is on average 10 times larger than the average sampling time of $t_s \approx 1/30$ s. However, the origin of this higher sampling time is so far unclear. Nevertheless, the convergence to the desired path is as expected with an exception around the transition point from the artificial path to the path generated by the vehicles. The follower shortly deviates from the path before converging towards the path again. This is also visible in Figure 4.12, where the 2-norm of the position error is depicted.

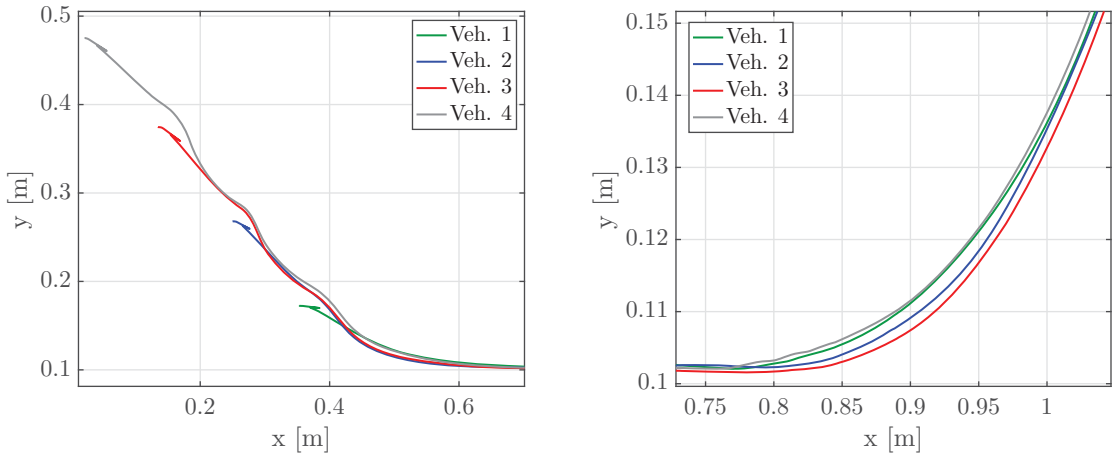

Figure 4.11: The left figure shows the paths of the vehicles from their initial positions. Converge as expected but shortly move away from the desired trajectory. However, when the vehicle starts corning, shown in the right figure, a deviation between the vehicles is noticed.

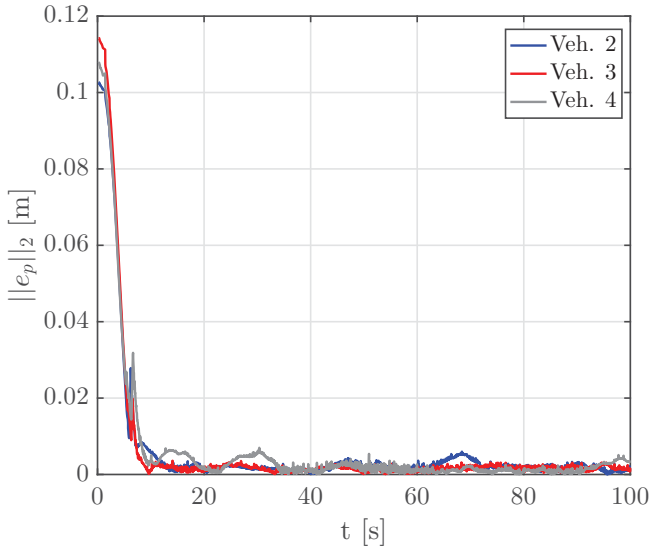

Figure 4.12: Norm of the position error from experiments incorporating four vehicles in a platoon. The error saturates around 3e-3 m.

All vehicles have the effect at approximately $t = 7\ s$, which is explained by the fact that the initial error between the vehicles is similar. Therefore, the vehicles reach the transition point from the artificial historic path to the generated historic path at similar points in time. Moreover, the orientation related to the artificial path was known and set to zero, resulting in a horizontal path. However, this historic path is supplemented with measurements, which are filtered and from which the orientation is estimated. The actual orientation at $t = 0$ is unknown. If initial conditions are far from the actual orientation angle, the observer immediately reacts to obtain a good estimation. Apparently the initial values for $\hat{c}, \hat{s}$ were poor, and the actual orientation angle was different from zero at $t = 0$. The estimated orientation angle of vehicle 3 is shown in Figure 4.13. It can be seen that the estimation reduces the noise on θ significantly. However, it takes more than 5 seconds to obtain a good estimation of the orientation angle. Tuning of the observer gains can improve the performance of the observer. As a final remark, vehicle 4 crosses the desired trajectory twice before staying on the path, explaining the two peaks at $t = 16\ s$ and $t = 30\ s$ respectively in Figure 4.12.

A closer look to the transition from a straight line to a curved path, depicted in the right plot of Figure 4.11, shows that the vehicles are not able to follow their predecessor precisely. However, from the position

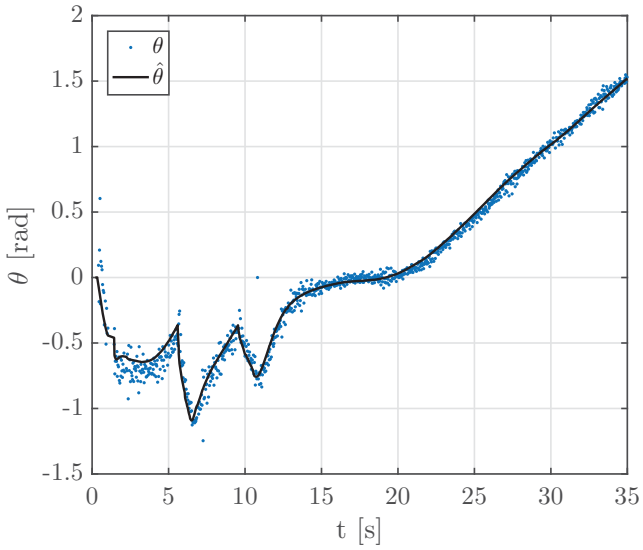

Figure 4.13: Estimated orientation angle in black and the orientation angle directly derived from unfiltered position measurements in blue.

error as depicted in Figure 4.12 it can be concluded that the vehicles eventually converge to the desired path with a deviation of approximately 3e-3 m. The propagation of the posture errors are now further studied in the next section.

4.2.2 Error propagation

In Figure 4.14 the posture errors are depicted for this specific experiment. For clarity, the error evolution is split in two parts; the convergence of the vehicles to the path of their predecessor, and the error when the vehicles are supposed to be on the path of their predecessor, both depicted on a different scale.
The posture errors of the first vehicle are not reviewed here since we are interested in the performance of the longitudinal and lateral controller of Lefeber et al. [18]. As can be seen, all errors converge towards zero within 40 seconds and stay around zero for the remainder of the experiment. It can be seen that errors do not attenuate along the string of the platoon, which implies that errors are similar for a platoon of n-vehicles. Nevertheless, the deviation with respect to the first vehicle can still be significant.

Figure 4.14 shows again the increasing errors between $t = 5$ s and $t = 10$ s. As mentioned before, the observer is not able to instantly estimate the orientation correctly. Additionally, an estimation error is present, caused by the large sampling time. Since the deviating estimation is also used when the reference position is determined, this influences the behavior of the vehicle. It must be noted that the estimation errors can be of less influence when other initial conditions are chosen. However, this experiment clearly shows that the transition from the artificial path to the measured path is of significant influence on the performance of the following vehicle.

Additional effects on the performance can be a delay and a mismatch in the desired velocity and the actual velocity, as was seen in open loop experiments. The controller does not explicitly correct for the delay that is mainly caused by the communication over a Bluetooth network. The overall performance is as expected in case of sampled, delayed and disturbed measurements. Further reduction of the posture errors is limited by the hardware. The observer already increases the performance by filtering the noise on the position measurements and providing a better estimate of the orientation angle. Additionally, the kinematic model does not perfectly describe the motion of the vehicles, since for example it does not take the full dynamic model of the vehicles into account or wheel slip.

4.2.3 Longitudinal performance

In Figure 4.15 the acceleration and forward velocity of each vehicle are depicted. Recall that the first vehicle uses a different controller to determine the velocity of the vehicle and therefore behaves differently.

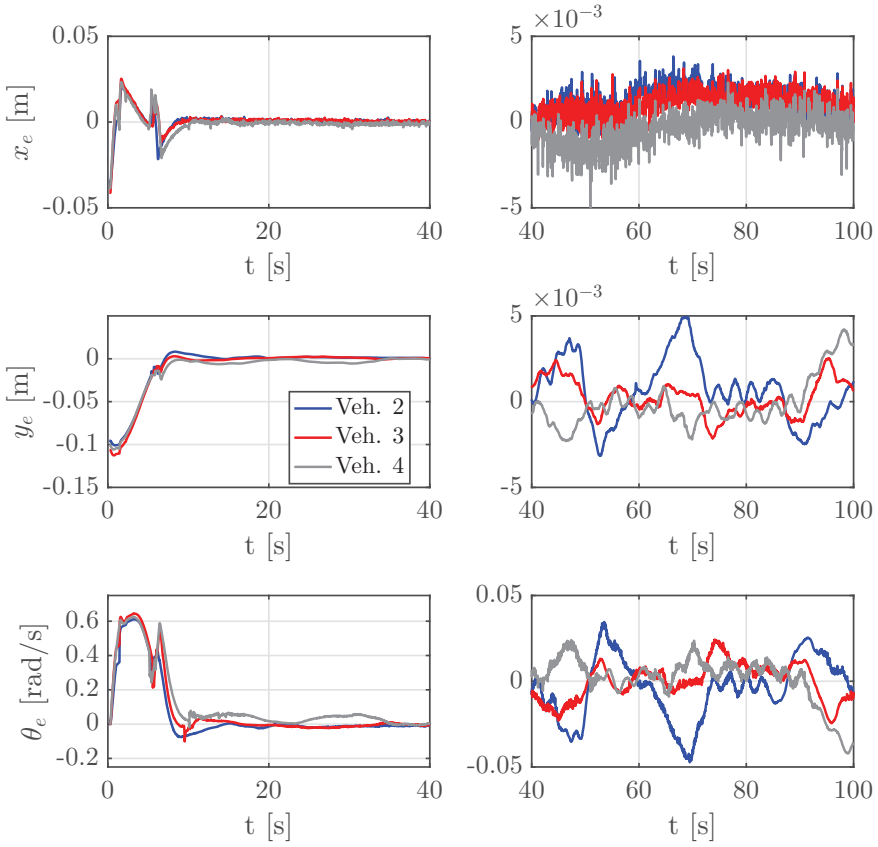

Figure 4.14: Error coordinates for each vehicle as a function of time at the beginning and end of the experiment on a different scale.

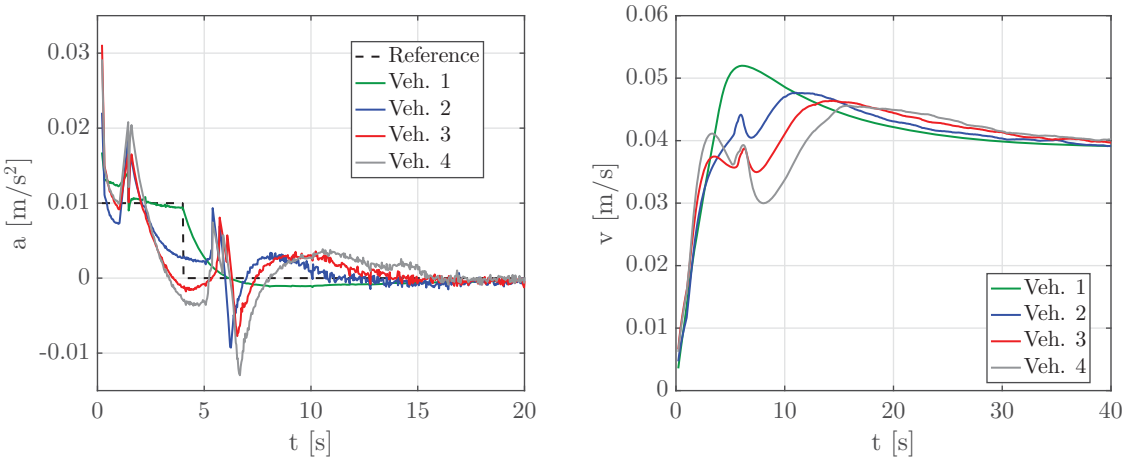

Figure 4.15: Acceleration and velocity with respect to time at the start of the experiment where the vehicles have initial errors in both lateral and longitudinal direction.

In the first 20 seconds of the experiment, the vehicles have a large lateral and longitudinal error. These errors both affect the velocity control at the start of the experiment. The error is well defined from the beginning, and therefore the vehicles are not affected by a negative initial error. The vehicles adapt their forward velocity when the posture errors change, due to the dependency of the controller (2.28) on $\bar{v}(x_e, \theta_e)$. From simulation results it is expected that vehicles adapt their speed before performing

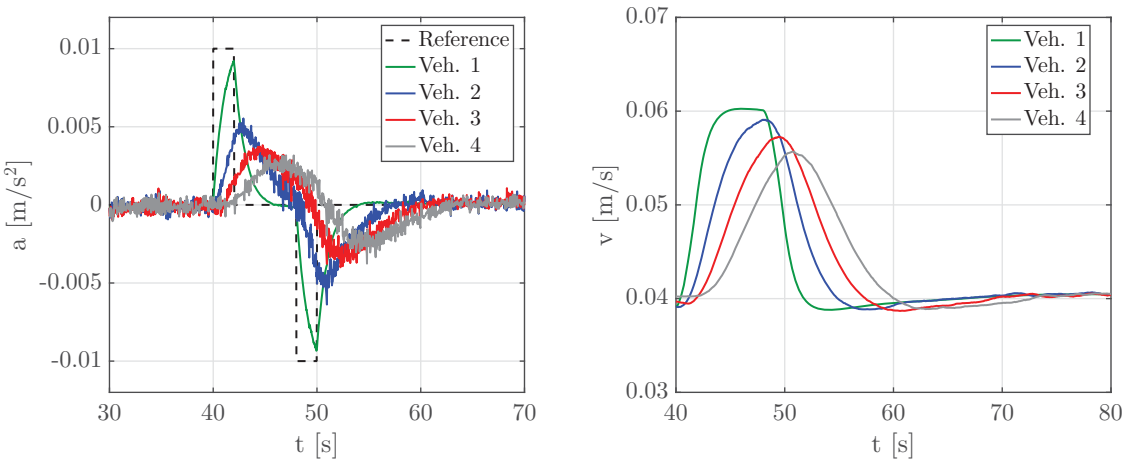

Figure 4.16: Acceleration and velocity with respect to time, where the first vehicle starts accelerates and decelerates. The vehicles are already on the desired path.

a sharp turn; however, in the experimental case an additional deceleration can be seen around $t = 7s$. The behavior can be explained by the divergence from the desired path, which influences the velocity. However, after this perturbation the velocity profile is as expected and shows a reduction of the velocity downstream the platoon. Figure 4.16 shows the acceleration and forward velocity of the vehicles when the first vehicle accelerates for a period of 2 seconds. Again, a reduction of the velocity downstream the platoon can be seen, which is characteristic behavior for string stable platoons. Furthermore, the amount of noise on the input signals is as expected from the simulations.

4.3 Concluding remarks

In this chapter, the proposed longitudinal and lateral controller are validated by simulations and experiments. The experimental results show large similarities to the results from simulation. This confirms the validity of the controller design and the simulation environment. However, the vehicles do show a larger deviation than expected when the vehicles make the transition from a straight line to a curvilinear path. This indicates that corners are still cut, which can be caused by delays, estimation errors or unmodelled dynamics as for example slip. The lateral and longitudinal errors converge to approximately zero where the deviation remains bounded. This deviation does not amplify downstream the platoon.

From experiments it follows that the transition from the artificial path to the actual driven path of the vehicle is undesired, and causes the follower to deviate from the path. This is concerning since this implies that there are additional effects which are not taken into account in simulations. Possible explanations are found in a poor estimation of the states at the start of the experiment and the influence of a deviating estimation. However, the estimated orientation angle converges rapidly to the actual states, after which the vehicle continues to converge to the desired path. Additional effects on the performance of the experiments are the presence of a delay, and the uncertainty of the actual velocity of the vehicles since this is not measured. Moreover, the controller is based on the assumption that there is no slip, and the vehicles are perfectly characterized by the kinematic model.

The response of the longitudinal controller is studied as well and showed satisfactory performance. The vehicles are able to reach and maintain the desired inter-vehicle distance. Furthermore, the vehicles respond as expected on acceleration and deceleration of their predecessors, and shows string stability in the platoon. The next chapter concludes this work and describes recommendations.

Chapter 5

Conclusions and Recommendations

5.1 Conclusions

In this thesis the spatial approach to the control of platooning mobile robots is validated by means of real-time experiments. Special attention is devoted to the problem of corner cutting. This problem has received relatively little attention in literature, but is essential for safe and accurate vehicle following. A transition is made from the theoretical control design to a practical application, taking into account the limitations of an experimental implementation. A simulation environment is created which is a close representation of the experimental environment and gives a good indication of the expected performance.

A literature study has shown that, in case of a follow-the-leader approach, the choice of a reference point for the following vehicle is crucial. Different approaches are studied in literature. Tracking controllers are therefore extended with look-ahead points to reduce the effect of corner cutting. This thesis is based on an alternative approach, where lateral and longitudinal control are separated. This allows convergence to the desired path as a main objective, before maintaining the desired velocity. The controller design considered in this work is able to theoretically eliminate corner cutting and provides the desired velocity behavior for the follower. However, this needs to be validated by experiments in a real-time environment. This further motivates the implementation and analysis of the proposed controller as is the main contribution of this thesis.

The lateral control problem is solved in the spatial domain, which required the derivation of a time dependent solution for real-time implementation. By solving the lateral control problem, a mapping from the path of the follower to the path of the leader is obtained and used to solve the longitudinal control problem as well. This resulted in a distance-based longitudinal error, which is initially not well defined. This problem is solved by introducing an artificial path for the leader, on which the position of the follower can be mapped. In theory it is possible to use a future path of the preceding vehicle. However, implementation of this method has indicated that, due to measurement uncertainties, this method has limited practical value. To overcome the problem of inaccurate angle measurements a four dimensional observer is proposed, driven by position measurements, to estimate the orientation. Using Lyapunov techniques, UGAS of the observer error dynamics has been shown.

The proposed controller is implemented in a simulation environment where the vehicles are represented by an exact discrete time kinematic model of the unicycle type. Simulations are carried out, including the noise on measurements and sampling time, and the behavior of the platoon is analyzed in both lateral and longitudinal direction. The results show sufficient performance and estimates in case of corrupted measurements. This allows to use the controller in a real-time experiment, where the performance is demonstrated for a simple trajectory with multiple cornering transitions. The vehicles start with an initial error in both lateral and longitudinal direction to demonstrate the effectiveness of the controller. The results show similar behavior compared to the results from simulations, which is good. However, the vehicles shortly deviate from the desired trajectory at the transition between the artificial and the actual historic path. The orientation angle is not immediately estimated correctly, causing this short deviation from the path. Overall, a good performance is achieved for the real-time controller with the available hardware.

5.2 Recommendations

Satisfactory performance of the platoon is already achieved by using a low cost controller implementation. However, the kinematic model of a mobile robot that is used assumes that there is no slip and does not consider further dynamic characteristics of the robot, which is not true in practical situations. Therefore, the dynamic model for mobile robots needs to be determined, taking the forces that affect the motion of the mechanical system into account. In order to improve the performance of the follower, a number of measures can be taken. For example, the mobile robots can be equipped with more sensors to determine the position of the vehicles more accurately. Additionally, local sensors can be used to obtain the relative positions and orientation measurements. For the implementation on real-life passenger vehicles this is a conceivable continuation, combining global and local measurement to optimize performance.

This thesis studied the performance at low velocities, providing enough measurement points to achieve an acceptable performance even in the absence of fitting the intermediate points on the path. In this work the intermediate trajectory between two points is achieved by interpolation between the points. However, the amount of points necessary for satisfactory performance in a full scale/real-life environment is not studied separately. Furthermore, the density of the measurements decreases at higher speeds and must be taken into account.

The control gains are chosen based on performance of the mobile robots at the available experimental setup and might not be optimal in all cases. It is therefore recommended to reconsider the chosen controller and observer gains. Appropriate tuning of the tracking controller for the first robot has received even less attention since this is only used to generate the path of the leader.

It is clear that the proposed method to determine the artificial path is far from desired. The transition from the artificial path to the real path is abrupt and results in a non-smooth reference trajectory. This causes undesired behavior for all follower robots. Therefore, a smooth artificial path needs to be determined, taking into account the expected path of the leader. Furthermore, it must be noted that the orientation of the robots is initially unknown, introducing another difficulty in determining the desired artificial path.

Improvements for the lateral controller can be found in the tracking control of marine vessels, where a desired orientation of the following vehicles points towards the virtual vehicle. As mentioned in this work, the initial position of the virtual vehicle influences to the degree of convergence to the desired path. This can be used in combination with an appropriate artificial path to obtain a smooth and fast convergence to the desired path. Finally, for driving comfort, it is important that all paths of the platoon connect properly.

Bibliography

[1] S. Adinandra. *Hierarchical coordination control of mobile robots*. PhD thesis, Eindhoven University of Technology, Eindhoven, The Netherlands,, 2012.

[2] A. Al Alam, A. Gattami, and K. H. Johansson. An experimental study on the fuel reduction potential of heavy duty vehicle platooning. In *Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on*, pages 306–311. IEEE, 2010.

[3] S. A. Al-Hiddabi and N. H. McClamroch. Tracking and maneuver regulation control for nonlinear nonminimum phase systems: Application to flight control. *IEEE Transactions on Control Systems Technology*, 10(6):780–792, 2002.

[4] A. Bayuwindra, Ø. L. Aakre, J. Ploeg, and H. Nijmeijer. Combined lateral and longitudinal CACC for a unicycle-type platoon. In *Intelligent Vehicles Symposium (IV), 2016 IEEE*, pages 527–532. IEEE, 2016.

[5] J. Bom, B. Thuilot, F. Marmoiton, and P. Martinet. Nonlinear control for urban vehicles platooning, relying upon a unique kinematic gps. In *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pages 4138–4143. IEEE, 2005.

[6] M. Breivik and T. I. Fossen. Path following for marine surface vessels. In *OCEANS'04. MTTS/IEEE TECHNO-OCEAN'04*, volume 4, pages 2282–2289. IEEE, 2004.

[7] J. Caarls. *Pose estimation for mobile devices and augmented reality*. PhD thesis, Delft University of Technology, Eindhoven, The Netherlands,, 2009.

[8] J. Dargay, D. Gately, and M. Sommer. Vehicle ownership and income growth, worldwide: 1960-2030. *The Energy Journal*, 28(4):143–170, 2007.

[9] S. K. Gehrig and F. J. Stein. A trajectory-based approach for the lateral control of car following systems. In *IEEE International Conference on Systems, Man, and Cybernetics*, volume 4, pages 3596–3601. IEEE, 1998.

[10] Y. Hayakawa, R. White, T. Kimura, and G. Naito. Driver-compatible steering system for wide speed-range path following. *IEEE/ASME Transactions on mechatronics*, 9(3):544–552, 2004.

[11] J. Jakubiak, E. Lefeber, K. Tchon, and H. Nijmeijer. Two observer-based tracking algorithms for a unicycle mobile robot. *International journal of applied mathematics and computer science*, 12(4):513–522, 2002.

[12] J. Jakubiak, H. Nijmeijer, and E. Lefeber. Observer based tracking controllers for a mobile car. Internship Report, University of Twente, Department of Applied Mathematics, Enschede, The Netherlands,, 1999.

[13] Z.-P. Jiang and H. Nijmeijer. Tracking control of mobile robots: a case study in backstepping. *Automatica*, 33(7):1393–1399, 1997.

[14] Y. Kanayama, Y. Kimura, F. Miyazaki, and T. Noguchi. A stable tracking control method for an autonomous mobile robot. In *IEEE International Conference on Robotics and Automation*, pages 384–389. IEEE, 1990.

[15] G. Klancar, D. Matko, and S. Blazic. Wheeled mobile robots control in a linear platoon. *Journal of Intelligent and Robotic Systems*, 54(5):709–731, 2009.

[16] D. Kostic, S. Adinandra, J. Caarls, and H. Nijmeijer. Collision-free motion coordination of unicycle multi-agent systems. In *Proceedings of the 2010 American Control Conference*, pages 3186–3191. IEEE, 2010.

[17] D. Kostic, S. Adinandra, J. Caarls, N. van de Wouw, and H. Nijmeijer. Collision-free tracking control of unicycle mobile robots. In *Proceedings of the 48th IEEE Conference on Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC.*, pages 5667–5672. IEEE, 2009.

[18] E. Lefeber, J. Ploeg, and H. Nijmeijer. A spatial approach to control of platooning vehicles: separating path-following from tracking. In *Proceedings of the 20th IFAC World Congress, Toulouse, France, 15565-15570*, 2017.

[19] E. Lefeber, S. Van den Eijnden, and H. Nijmeijer. Almost global tracking control of a quadrotor uav on se(3). In *Proceedings of the 56th IEEE Conference on Decision and Control, Melbourne, Australia*, 2017.

[20] A. Loría, E. Panteley, D. Popovic, and A. R. Teel. A nested Matrosov theorem and persistency of excitation for uniform convergence in stable nonautonomous systems. *IEEE Transactions on Automatic Control*, 50(2):183–198, 2005.

[21] G. Lu and M. Tomizuka. A laser scanning radar based autonomous lateral vehicle following control scheme for automated highways. In *American Control Conference, 2003. Proceedings of the 2003*, volume 1, pages 30–35. IEEE, 2003.

[22] P. Morin and C. Samson. Motion control of wheeled mobile robots. *Springer Handbook of Robotics*, 1:799–826, Springer-Verlag Berlin Heidelberg,2008.

[23] G. J. Naus, R. P. Vugts, J. Ploeg, M. J. van de Molengraft, and M. Steinbuch. String-stable cacc design and experimental validation: A frequency-domain approach. *IEEE Transactions on Vehicular Technology*, 59(9):4268–4279, 2010.

[24] S. Noijen, P. Lambrechts, and H. Nijmeijer. An observer-controller combination for a unicycle mobile robot. *International Journal of Control*, 78(2):81–87, 2005.

[25] P. Petrov and O. Boumbarov. Nonlinear adaptive control of a two-vehicle autonomous convoy using a look-ahead approach. In *Proceedings of the 7th WSEAS International Conference on Signal Processing, Robotics and Automation*, pages 55–60. World Scientific and Engineering Academy and Society (WSEAS), 2008.

[26] J. Ploeg. *Analysis and design of controllers for cooperative and automated driving*. PhD thesis, Eindhoven University of Technology, Eindhoven, The Netherlands,, 2014.

[27] J. Ploeg, B. T. Scheepers, E. van Nunen, N. van de Wouw, and H. Nijmeijer. Design and experimental evaluation of cooperative adaptive cruise control. In *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 260–265. IEEE, 2011.

[28] R. Rajamani, H.-S. Tan, B. K. Law, and W.-B. Zhang. Demonstration of integrated longitudinal and lateral control for the operation of automated vehicles in platoons. *IEEE Transactions on Control Systems Technology*, 8(4):695–708, 2000.

[29] D. Soetanto, L. Lapierre, and A. Pascoal. Adaptive, non-singular path-following control of dynamic wheeled robots. In *Proceedings. 42nd IEEE Conference on Decision and Control.*, volume 2, pages 1765–1770. IEEE, 2003.

[30] Ö. Tunçer, L. Güvenç, F. Coşkun, and E. Karsligil. Vision based lane keeping assistance control triggered by a driver inattention monitor. In *IEEE International Conference on Systems Man and Cybernetics (SMC)*, pages 289–297. IEEE, 2010.

[31] A. Vahidi and A. Eskandarian. Research advances in intelligent collision avoidance and adaptive cruise control. *IEEE transactions on intelligent transportation systems*, 4(3):143–153, 2003.

[32] T. Van Den Broek, N. Van De Wouw, and H. Nijmeijer. A virtual structure approach to formation control of unicycle mobile robots. *Eindhoven University of Technology, the Netherlands, Tech. Rep. DCT*, 2009, 2009.

[33] E. van Nunen, R. Kwakkernaat, J. Ploeg, and B. D. Netten. Cooperative competition for future mobility. *IEEE Transactions on Intelligent Transportation Systems*, 13(3):1018–1025, 2012.

Appendix A

Lateral controller:Proof

This appendix provides the preliminaries and proof of the locally exponentially stable lateral controller as shown by Lefeber et al. [18]. It must be clear that this is a copy of the provided proof and is only shown for completeness of this work.

A.1 Preliminaries

Lemma A.1 (Barbălat's Lemma, see Barbălat (1959)). Let $\phi : \mathbb{R}_+ \to \mathbb{R}$ be a uniformly continuous function. Suppose that $\lim_{t\to\infty} \int_0^t \phi(\tau)d\tau$ exists and is finite. Then

$$\lim_{t\to\infty} \phi(t) = 0.$$

Lemma A.2 (Micaelli and Samson, 1993, Lemma 1). Let $f : \mathbb{R}_+ \to \mathbb{R}$ be any differentiable function. If $f(t)$ converges to zero as $t \to \infty$ and its derivative satisfies

$$\dot{f}(t) = f_0(t) + \eta(t) \qquad t \geq 0,$$

where f_0 is a uniformly continuous function and $\eta(t)$ tends to zero as $t \to \infty$, then $\dot{f}(t)$ and $f_0(t)$ tend to zero as $t \to \infty$.

Theorem A.1 ([19], Theorem 2), (cf. [20], Theorem 1): Consider the dynamical system

$$\dot{x} = f(t, x) \qquad x(t_0) = x_0 \tag{A.1}$$

with $f(t, 0) = 0, f : \mathbb{R}^+ \times \mathbb{R}^n \to \mathbb{R}$ locally bounded, continuous and locally uniformly continuous in t. If there exist j differentiable functions $V_i : \mathbb{R}^+ \times \mathbb{R}^n \to \mathbb{R}$, bounded in t, and continuous functions $Y_i : \mathbb{R}^n \to \mathbb{R}$ for $i \in \{1, 2, ..., j\}$ such that

- V_1 is positive definite,
- $\dot{V}_i(t, x) \leq Y_i(x)$, for all $i \in \{1, 2, ..., j\}$,
- $Y_i(x) = 0$ for $i \in \{1, 2, ..., k-1\}$ implies $Y_k(x) \leq 0$,
- $Y_i(x) = 0$ for all $i \in \{1, 2, ..., j\}$ implies $x = 0$,

the the origin $x = 0$ of (A.1) is uniformly globally asymptotically stable (UGAS).

A.2 Lateral controller: Proof

In this appendix the local exponentially stability of the closed-loop system as proven by Lefeber et al. [18] is given for completeness of this work. Recall the error coordinates,

$$\begin{bmatrix} x_e(s) \\ y_e(s) \\ \theta_e(s) \end{bmatrix} = \begin{bmatrix} \cos\theta(s) & \sin\theta(s) & 0 \\ -\sin\theta(s) & \cos\theta(s) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \bar{x}_l(s) - x(s) \\ \bar{y}_l(s) - y(s) \\ \bar{\theta}_l(s) - \theta(s) \end{bmatrix}, \tag{A.2}$$

where the dependency on t is again omitted for readability. The error dynamics can be derived and are given by

$$\begin{aligned} \dot{x}_e &= \kappa y_e + \bar{v}\cos\theta_e - 1 \\ \dot{y}_e &= -\kappa x_e + \bar{v}\sin\theta_e \\ \dot{\theta}_e &= -\kappa + \bar{v}\bar{\kappa}_l. \end{aligned} \tag{A.3}$$

Consider initial conditions satisfying $|\theta_e(0)| < \pi/2$. Differentiating the function $V = \frac{1}{2}x_e^2 + \frac{1}{2}y_e^2 - \frac{1}{c_3}\log(\cos\theta_e)$, which is positive definite for $c_3 > 0$, along solutions of (A.3) results in

$$V' = x_e(\bar{v}\cos\theta_e - 1) + y_e(\bar{v}\sin\theta_e) + \frac{1}{c_3}\tan\theta_e(-\kappa + \bar{v}\bar{\kappa}_l). \tag{A.4}$$

Using the controller

$$\bar{v} = \frac{1 - c_1\sigma_1(x_e)}{\cos\theta_e} \tag{A.5}$$

$$\kappa = c_3 y_e(1 - c_1\sigma_1(x_e)) + \bar{v}\bar{\kappa}_l + c_2\sigma_2(\theta_e) \tag{A.6}$$

with $0 < c_1 < 1$ and $0 < c_2$ results in

$$V' = -c_1 x_e\sigma_1(x_e) - \frac{c_2}{c_3}\sigma_2(\theta_e)\tan\theta_e \leq 0. \tag{A.7}$$

This implies that x_e, y_e, and $-\log(\cos\theta_e)$ are bounded, which implies that $|\theta_e| \leq M < \pi/2$ and $\bar{v}$ is bounded. Since κ_l is bounded by assumption, we have that $\bar{\kappa}_l$ is bounded, and also κ is bounded (from (A.6)) and as a result x_e', y_e', and θ_e are uniformly continuous functions of s. From Lemma A.1 applied to (A.7), we then have that $\lim_{s\to\infty} x_e(s) = \lim_{s\to\infty}\theta_e(s) = 0$.

From (A.2) we then also obtain

$$\begin{aligned} \lim_{s\to\infty} x_l(\alpha(s)) - x(s) &= 0 \\ \lim_{s\to\infty} y_l(\alpha(s)) - y(s) &= 0 \\ \lim_{s\to\infty} \theta_l(\alpha(s)) - \theta(s) &= 0 \end{aligned} \tag{A.8}$$

Furthermore, since $1 - c_1\sigma_1(x_e) \geq 1 - c_1 > 0$ and $|\theta_e| \leq M < \pi/2$, we have that $\bar{v}(s) \geq 1 - c_1 > 0$, and therefore $s_l = \alpha(s)$ is a diffeomorphism, where $\alpha(s)$ is obtained from

$$\frac{d\alpha(s)}{ds} = \bar{v}(s), \qquad \alpha(0) = 0. \tag{A.9}$$

Appendix B

Tracking controller with inputs a and ω

The tracking controller of [13] was originally designed for the velocity inputs v and ω. However, since the longitudinal controller of [18] uses acceleration as input for the following vehicles the choice is made to rewrite the tracking controller. Consider once more the kinematic model of a mobile car where the model is extended by adding the $\dot{v} = a$, resulting in

$$\dot{x} = v\cos\theta \tag{B.1}$$
$$\dot{y} = v\sin\theta \tag{B.2}$$
$$\dot{\theta} = \omega \tag{B.3}$$
$$\dot{v} = a. \tag{B.4}$$

The additional velocity error can be defined as $v_e = v_r - v$ leading to the error dynamics,

$$\dot{x}_e = \omega y_e - v_r(1-\cos\theta_e) + v_e \tag{B.5}$$
$$\dot{y}_e = -\omega x_e + v_r\sin\theta_e \tag{B.6}$$
$$\dot{\theta}_e = \omega_r - \omega \tag{B.7}$$
$$\dot{v}_e = a_r - a. \tag{B.8}$$

Consider now candidate Lyapunov function V and its derivative V'

$$V = \frac{1}{2}x_e^2 + \frac{1}{2}y_e^2 + \frac{1}{2c_2}\theta_e^2 + \frac{1}{2c_1}v_e^2 \tag{B.9}$$
$$V' = \frac{1}{c_2}[-c_2v_rx_e\frac{1-\cos\theta_e}{\theta_e} + c_2v_ry_e\frac{\sin\theta_e}{\theta_e} + \omega_r - \omega]\theta_e + \frac{1}{c_1}[c_1x_e + a_r - a]v_e. \tag{B.10}$$
$$\tag{B.11}$$

Using the controller

$$\omega = \omega_r - c_2v_rx_e\frac{1-\cos\theta_e}{\theta_e} + c_2v_ry_e\frac{\sin\theta_e}{\theta_e} + c_3\theta_e \tag{B.12}$$
$$a = a_r + c_1x_e + c_4v_e, \tag{B.13}$$

one can find that

$$V' = -\frac{c_3}{c_2}\theta_e^2 - \frac{c_4}{c_1}v_e^2 \tag{B.14}$$

where θ_e and v_e approach zero. Using Lemma A.2,

$$\dot{v}_e = -c_1x_e - c_4v_e \tag{B.15}$$
$$\dot{\theta}_e = c_2v_rx_e\frac{1-\cos\theta_e}{\theta_e} + c_2v_ry_e\frac{\sin\theta_e}{\theta_e} - c_3\theta_e \tag{B.16}$$
$$= c_2v_ry_e \tag{B.17}$$

we can conclude that x_e and y_e also converge to zero.

Appendix C

Experimental analysis of delays

This appendix investigates the input delay as detected during experiments. Compensation of the delay requires more knowledge of the delay in terms of accumulation, consistency and composition which are now further investigated. From [1] it was already known that the provided Bluetooth dongle can only handle up to 7 mobile robots, and delays escalate with an increasing number of robots. Delays also arise due to computation time of the PC, where there is another small delay in getting the necessary signals. Another limitation is observed at the edge of the arena caused by the camera which captures a more bended image and therefore gives less accurate position measurements. First, the delay is investigated in situations including multiple vehicles to determine whether the delays are constant or increase along the platoon. To achieve these results, image and signal processing are separated to determine the composition of the delay.

C.1 Delay

An effective experiment to indicate the delay is based on driving in a straight line along the x- or y-axis. Multiple e-pucks are used, receiving input values for a set time after which the vehicles need to stop. The experiment is done several times for reliability. Figure C.1 shows the measurement result for experiment 2, which is discussed since it includes all possible situations that can appear during an experiment. First of all, some e-pucks lose connection during the experiments as happens to vehicle 4, which is not able to follow the given commands at 5 and 15 seconds and therefore holds its last command. Nevertheless, after 25 seconds the Bluetooth connection is restored after which the vehicles start to participate in the experiment. This loss of connection occurs at all vehicles and cannot be fully assigned to low battery life since the vehicles are able to behave perfectly during the follow up experiments. However, low batteries do have a major influence on the driving mechanism of the e-puck and therefore should be taken care of. Table C.1 shows the results of the other experiments. Contrary to experiment 2, 3 and 4, experiment 1 is only executed with one e-puck mobile robot. The delay for this experiment was on average almost two times larger than the delays from experiment 2, 3 and 4. Apparently, conditions were not optimal and caused a significantly larger delay which can also be seen in experiments from the past. Lastly, a careful reader might notice that vehicle 1 stops earlier than the other vehicles, but starts driving at the same time. This would imply that the vehicle receives the signal to stop earlier, which is unlikely since it starts driving again at the same moment in time. Regarding the result of the experiments using four vehicles, it can be concluded that the delay is not constant, but on average has a time period of 0.133 seconds. The experiments clearly show that there are a lot of uncertainties to take into account.

This delays can mainly be explained by the fact that the e-pucks are remotely controlled over a Bluetooth communication network which induces a transport and processing delay. To identify if this delay is mainly caused by the PC/e-puck communication, simple experiments are performed. Firstly, the image processing and input calculations are excluded. Only one input is communicated to the e-puck and the behaviour of the e-puck is recorded by a camera. Frame analysis of the video resulted in an average delay of 0.133 seconds, confirming the conclusion from the previous experiments. The influence of image processing and calculation time are respectively 0.02 and 0.01 [s] and can therefore be neglected.

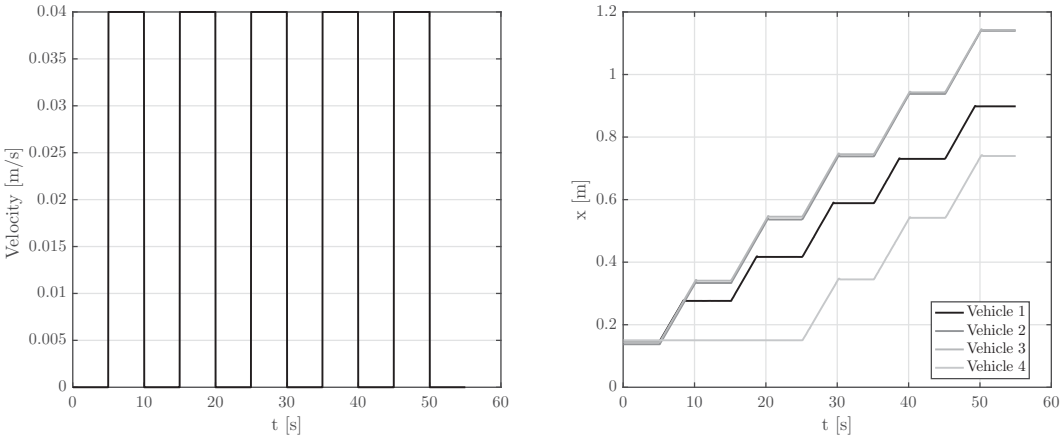

Figure C.1: Velocity profile and measurements of the e-pucks in x-direction of experiment 2.

Table C.1: Delay of the vehicles in seconds for four different experiments.

Step	Exp. 1	Exp. 2	Exp. 3	Exp. 4
5	0.339	0.148	0.154	0.133
15	0.28	0.15	0.13	0.16
25	0.34	0.13	0.16	0.1
35	0.18	0.16	0.17	0.14
45	0.25	0.16	0.17	0.13
Mean	0.2778	0.1496	0.1568	0.1326

C.2 Conclusions

The delay is varying during the experiment, but within the same range and there appears to be no accumulation of the delay along time. Despite the varying delay, it is decided to take a delay of 4 time steps into account since the delay is on average around 0.133 seconds, equal to approximately 4 time steps.