



HAL
open science

High-speed path following control of skid-steered vehicles

Goran Huskić, Sebastian Buck, Matthieu Herrb, Simon Lacroix, Andreas Zell

► **To cite this version:**

Goran Huskić, Sebastian Buck, Matthieu Herrb, Simon Lacroix, Andreas Zell. High-speed path following control of skid-steered vehicles. *The International Journal of Robotics Research*, 2019, 38 (9), pp.1124-1148. 10.1177/0278364919859634 . hal-02994317

HAL Id: hal-02994317

<https://laas.hal.science/hal-02994317>

Submitted on 7 Nov 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

High Speed Path Following Control of Skid-Steered Vehicles

Goran Huskić¹, Sebastian Buck¹, Matthieu Herrb², Simon Lacroix², and Andreas Zell¹

¹: Chair of Cognitive Systems, University of Tübingen, Sand 1, 72076, Tübingen, Germany

²: LAAS-CNRS, Université de Toulouse, CNRS, Toulouse, France

Corresponding author: Goran Huskić goran.huski@uni-tuebingen.de

Abstract—We present a robust control scheme for skid-steered vehicles that enables high speeds path following on challenging terrains. First, a kinematic model with experimentally identified parameters is constructed, in order to describe the terrain-dependent motion of skid-steered vehicles. Using the Lyapunov theory, a nonlinear control law is defined, guaranteeing the convergence of the vehicle to the path. To allow smooth and accurate motion at higher speeds, an additional linear velocity control scheme is proposed, which takes actuator saturation, path following error, and reachable curvatures into account. The combined solution is experimentally evaluated and compared against two state-of-the-art algorithms, by using two different robots on several different terrain types, at different speeds. A Robotnik Summit XL robot is tested on three different terrain types and two different paths at speeds up to $\approx 2.5\text{m/s}$. A Segway RMP 440 robot is tested on three different terrain types and two different path types at speeds up to $\approx 6\text{m/s}$.

Index Terms—path following, control, high speed, rough terrain, skid steering

I. INTRODUCTION

Navigation of autonomous or automated vehicles should be safe, accurate, and robust, which is especially challenging at high speeds and on rough terrain. After a collision-free path has been planned, the vehicle should be able to follow the planned route as exactly as possible. This task of path following can be defined as minimizing the distance and the orientation error between the current vehicle configuration and a desired configuration on the reference path. If the planned path is only a geometric reference, with no timing law assigned to it, the convergence of the vehicle to the path and the vehicle's speed can be controlled independently.

In the literature, a distinction between path following and trajectory tracking is usually made, such as in [Aguilar et al., 2004] or [De Luca et al., 1998]. As opposed to the path following task, trajectory tracking is not a pure geometric tracking, since for each reference point there is a time point assigned to it. Nevertheless, since each set of points can be parameterized in an arbitrary fashion, the difference between a path and a trajectory lies in the parameterization, whether it is done by using some timing law, or e.g. the arc length.

The work presented in this paper deals with the problem of path following, and the goal is to follow a geometric curve as accurately as possible, while driving close to maximum speed on different terrain types, especially outdoors. This ability can

benefit applications like agriculture and surveillance, as well as exploration or search and rescue.

For driving at higher speeds and on varying terrain, both indoors and outdoors, skid-steered vehicles are especially suitable. They are relatively cheap to produce and very robust, since there are no moving parts for the steering system. Furthermore, they are highly manoeuvrable and have high traction. However, skid-steered vehicles are difficult to model and to control, since the motion is terrain-dependent and always includes skidding, hence it violates the nonholonomic constraint, which assumes pure rolling.

Path following at higher speeds on rough terrain is still an ongoing and challenging research topic. Some work on aggressive driving with car-like vehicles based on Model Predictive Control (MPC) is presented in [Williams et al., 2016] and [Williams et al., 2017]. Off-road control of car-like vehicles at high speeds using observer-based control can be found in [Lenain et al., 2011], [Lenain et al., 2010] and [Deremetz et al., 2017]. Similar work on double-steering vehicles can be found in [Lucet et al., 2009]. Reactive high speed navigation on rough terrain is tackled in [Shimoda et al., 2007] and [Spenko et al., 2004]. A comprehensive study on mobile robots on rough terrain can be found in [Iagnemma and Dubowsky, 2004]. High speed path following based on pure pursuit and receding strategy is proposed in [Elbanhawi et al., 2016], while backstepping steering control can be found in [Xin and Minor, 2012]. All of these approaches offer interesting solutions for driving at higher speeds on rough terrain, but none of them deals with skid-steered vehicles.

Trajectory tracking control of skid-steered vehicles can be found in [Caracciolo et al., 1999], [Kozłowski and Pazderski, 2004], or [Yi et al., 2007]. Similar work on this topic can be found in [Kozłowski and Pazderski, 2006], [Yi et al., 2009], and [Miller and Murphey, 2012]. However, these approaches do not offer experimental evaluation at higher speeds. One solution to path following using skid-steered vehicles is proposed in [Pentzer et al., 2014b], where a unicycle control strategy is used to control a skid-steered robot with the help of a kinematic mapping. This mapping is based on the parameter estimation proposed in [Pentzer et al., 2014a]. Their proposed path following solution is experimentally evaluated at speeds up to 1m/s . Path following which takes wheel slip into account was proposed in [Rajagopalan et al., 2016], where

the maximum speed of the robot in the experiments reaches 1m/s. In [Ostafew et al., 2016], tracking of manually defined paths with constraints using a learning-based Nonlinear Model Predictive Control (NMPC) is done with skid-steered vehicles at speeds up to 2m/s. In [Indiveri et al., 2007] a unicycle control strategy from [Soetanto et al., 2003] is extended for higher speeds and experimentally evaluated on a skid-steered vehicle.

In [Huskić et al., 2017c] a robust path following algorithm for skid-steered vehicles is proposed, and experimentally compared against the approaches proposed in [Pentzer et al., 2014b] and [Indiveri et al., 2007] at speeds up to 2.5m/s. This paper is based on the work proposed in [Huskić et al., 2017c], providing the theoretical background in more detail, as well as extending the experimental evaluation, where the maximum speed reaches 6m/s. The proposed algorithm is compared against the two already mentioned state-of-the-art algorithms on two different robots: Robotnik Summit XL and Segway RMP 440. With the Robotnik Summit XL, the comparison is made on three different terrain types, and on two different paths. Using the Segway RMP 440, the comparison is made on three different terrain types and two different path types.

This paper is organized as follows. In Section II the properties of the skid steering kinematics are described, and a model based on Instantaneous Centres of Rotations is introduced. Then, the proposed path following algorithm is introduced in Section III, and the proposed longitudinal control in Section IV. The experiments are described in Section V, and Section VI concludes the paper.

II. SKID STEERING

A skid steering mechanism is similar to the one of a differential drive, but not identical. While a differential drive has two non-steerable wheels on one axle, a skid steered vehicle has $2 \cdot k$ non-steerable wheels, where k is greater than 1. Instead of wheels, a skid-steered vehicle can also have tracks, while the principle stays the same. In the rest of the text, one track, or one side of the wheels, will be referred to as a *tread*. Furthermore, it is assumed that differential drives have only contact points between the wheels and the ground, while skid-steered vehicles have contact patches between the treads and the ground.

In order for such a vehicle to steer, one tread has to move faster than the other, which means the other one needs to skid, hence the name of the mechanism. Since these vehicles always skid, it means that the nonholonomic constraint, which assumes pure rolling, is violated. Such a motion highly depends on the contacts between the treads and the ground, which involves complex interactions such as torsion of both treads and the ground surface. This makes it difficult to model such a behaviour without having a full dynamical model which includes the friction between the treads and the ground. This kind of models can be very complex and their parameters difficult to estimate. Such models can be found in [Wong, 2001] and [Yu et al., 2010]. Furthermore, the majority of commercial robots only offer velocities as control

commands, while a real dynamical model would need torques (which are usually assumed to be linearly proportional to the motor currents). Controlling a dynamic model with velocity inputs would then require transformations such as the one proposed in [Martins et al., 2017].

Driven by these reasons, for the work presented in this paper we have embraced a type of kinematic modelling which implicitly takes dynamics into account, and can be easily configured for different robots with different dynamic characteristics. The idea for such a model comes from [Martínez et al., 2005], and it was further developed in [Mandow et al., 2007].

A. ICR kinematics

If a skid-steered vehicle is observed as a rigid body with planar motion, there is a point in the plane in which the motion of the vehicle can be represented by pure rotation. This point is the Instantaneous Center of Rotation (ICR). It can be seen in Fig. 1, where the ICR is expressed in vehicle's local coordinates (x, y) . The angular velocity is denoted with ω , and the linear velocity vector is $v = (v_x, v_y)$. Control inputs for this system are the speeds of the left and the right tread, V_l and V_r .

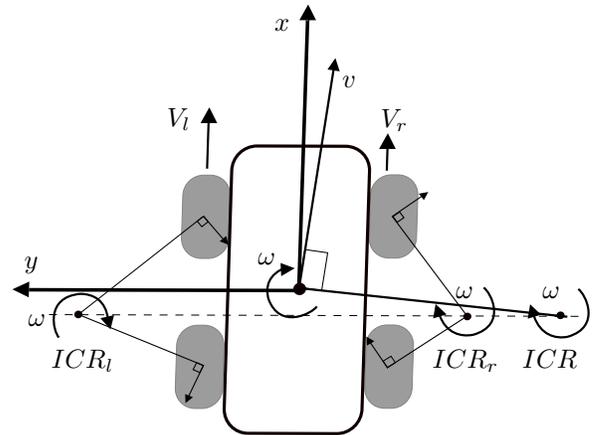


Fig. 1. Skid steering kinematics based on Instantaneous Centres of Rotation.

The motion of a particle on the tread is composed of the input speed of the tread and the component that comes from the motion of the entire vehicle. This means that the treads have their own ICRs, different from the ICR of the entire vehicle. As opposed to differential drives, where the ICRs of the left and the right wheel coincide with the ground contact points, in the case of skid-steered vehicles, the tread ICRs usually lie outside of the treads. This offset is a direct consequence of the slippage between the treads and the ground, and it is terrain-dependent. In Fig. 1, the ICRs of the left and the right tread are denoted as ICR_l and ICR_r , respectively.

As stated in [Martínez et al., 2005], because of Kennedy's ICR theorem (details can be found in [Shigley and Uicker, 1991]), all the three ICRs lie on the same line, and have the same x -coordinate x_{ICR} . The ICR y -coordinates of the vehicle, the left, and the right tread, are denoted as y_{ICR} , y_{ICR_l} , and y_{ICR_r} , respectively. Furthermore, all three ICRs have the same angular velocity.

Now, the following expressions can be derived:

$$\begin{aligned} v_x &= y_{ICR}\omega, \\ v_y &= -x_{ICR}\omega, \end{aligned} \quad (1)$$

which describe the motion of the entire body. The speeds of the left and the right tread can be written as

$$\begin{aligned} V_l &= \frac{v_x - y_{ICRl}\omega}{\alpha_l}, \\ V_r &= \frac{v_x - y_{ICRr}\omega}{\alpha_r} \end{aligned} \quad (2)$$

where α_l and α_r are parameters which encompass mechanical characteristics such as tire inflation or belt transmission, as proposed in [Mandow et al., 2007].

Using Eq. 1 and Eq. 2, the kinematic equations of skid-steered vehicles based on ICR parameters can be expressed as:

$$\begin{aligned} v_x &= \frac{\alpha_l V_l y_{ICRr} - \alpha_r V_r y_{ICRl}}{y_{ICRr} - y_{ICRl}}, \\ v_y &= x_{ICR} \frac{\alpha_r V_r - \alpha_l V_l}{y_{ICRr} - y_{ICRl}}, \\ \omega &= \frac{\alpha_l V_l - \alpha_r V_r}{y_{ICRr} - y_{ICRl}}. \end{aligned} \quad (3)$$

If $\alpha_l = \alpha_r = 1$, $x_{ICR} = 0$, $y_{ICRl} = -y_{ICRr} = w$, where $2w$ is the distance between the right and the left tread, then Eq. 3 represents the kinematic model of a differential drive. The ICR parameters ($x_{ICR}, y_{ICRl}, y_{ICRr}, \alpha_l, \alpha_r$) of skid-steered vehicles can be seen as a deviation from an ideal differential drive. This allows a control law developed for skid-steered vehicles to be applied to differential drives, when the ICR parameters are adjusted accordingly.

As proposed in [Caracciolo et al., 1999], skid-steered vehicles have a nonholonomic constraint which can be expressed as

$$x_{ICR}\omega + v_y = 0. \quad (4)$$

By taking this nonholonomic constraint into account while describing the vehicle's kinematics in global coordinates, the following expression can be derived:

$$\begin{bmatrix} \dot{X}_w \\ \dot{Y}_w \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos \theta & x_{ICR} \sin \theta \\ \sin \theta & -x_{ICR} \cos \theta \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_x \\ \omega \end{bmatrix}, \quad (5)$$

Here, \dot{X} and \dot{Y} are the linear velocity components in global coordinates, while $\dot{\theta} = \omega$, where θ is the global orientation of the vehicle. The boundedness of x_{ICR} is discussed in [Caracciolo et al., 1999], [Kozłowski and Pazderski, 2004] and [Kozłowski and Pazderski, 2006]. In our work, we experimentally identify the ICR parameters ($x_{ICR}, y_{ICRl}, y_{ICRr}, \alpha_l, \alpha_r$) for different terrain types, by using evolutionary algorithms, similar as in [Mandow et al., 2007]. This kind of identification provides realistic values for x_{ICR} which are always bounded.

III. PATH FOLLOWING

In order to define and solve the path following problem, one practical approach is to transform the model of the vehicle to path coordinates. The model transformation and the control law derivation presented here are mostly based on the work of [Soetanto et al., 2003], and the pioneering work in this area made by [Kanayama et al., 1990] and [Samson and Ait-Abderrahim, 1991].

A. Transformed kinematic model

If we observe Fig. 2, a skid-steered vehicle is depicted similarly as in Fig. 1, having its centre Q and orientation θ in the world frame. The geometric reference path is denoted with \mathbf{P} and parameterized by the path parameter $s : [0, \infty) \rightarrow [0, \infty)$, a free control parameter which can be arbitrarily specified. For the path following problem, this parameter is usually chosen as the arc length of the path. The world frame $\{W\}$ is defined by the coordinates (X_w, Y_w) and zero O , while there is a Serret-Frenet frame $\{F\}$ moving along the path, defined by its coordinates (x_e, y_e) , and having its origin in the point P . This moving reference frame is rotated in the world frame by the angle θ_t , which is the angle of the path tangent in the point P . The path following problem can now be defined as making the (x_e, y_e) coordinates, as well as the difference $\theta_e = \theta - \theta_t$, converge to zero.

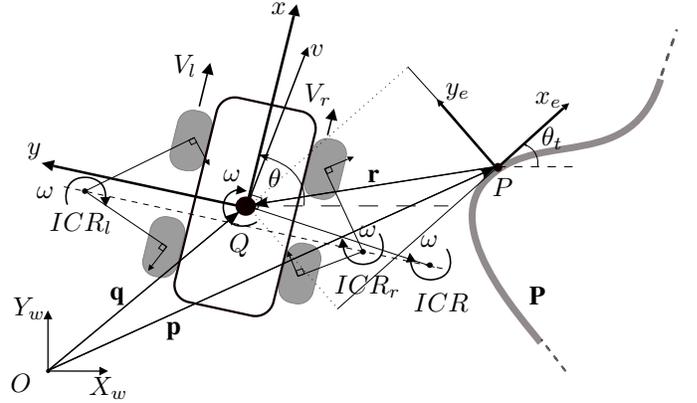


Fig. 2. Path following geometry with a skid-steered vehicle.

Let us now define a vector \mathbf{q} , which is the robot's position vector in world coordinates, pointing from the point O to the point Q , and a vector \mathbf{p} , pointing from O to P , which is the position vector of the Serret-Frenet frame. If we now define a vector \mathbf{r} in the Serret-Frenet frame, pointing from P to Q , the following relation holds

$$\dot{\mathbf{q}} = \dot{\mathbf{R}}(\theta_t)\mathbf{r} + \mathbf{R}(\theta_t)\dot{\mathbf{r}} + \dot{\mathbf{p}}, \quad (6)$$

where $\mathbf{R}(\theta_t)$ is the rotation matrix from $\{F\}$ to $\{W\}$, around the angle θ_t .

If we consider that

$$\mathbf{r} = \begin{bmatrix} x_e \\ y_e \end{bmatrix}, \quad \mathbf{R}(\theta_t)\dot{\mathbf{p}} = \begin{bmatrix} \dot{s} \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{q} = \begin{bmatrix} X_w \\ Y_w \\ 0 \end{bmatrix}, \quad (7)$$

we can derive the dynamics of the error coordinates:

$$\begin{aligned}\dot{x}_e &= [\cos \theta_t \quad \sin \theta_t] \begin{bmatrix} \dot{X}_w \\ \dot{Y}_w \end{bmatrix} - \dot{s}(1 - c(s)y_e), \\ \dot{y}_e &= [-\sin \theta_t \quad \cos \theta_t] \begin{bmatrix} \dot{X}_w \\ \dot{Y}_w \end{bmatrix} - c(s)\dot{s}x_e.\end{aligned}\quad (8)$$

Using the global velocities from Eq. 5, we derive the skid steering model in path coordinates:

$$\begin{aligned}\dot{x}_e &= v_x \cos \theta_e + x_{ICR}\omega \sin \theta_e - \dot{s}(1 - c(s)y_e), \\ \dot{y}_e &= v_x \sin \theta_e - x_{ICR}\omega \cos \theta_e - c(s)\dot{s}x_e, \\ \dot{\theta}_e &= \omega - c(s)\dot{s}, \quad \dot{\theta}_t = c(s)\dot{s}.\end{aligned}\quad (9)$$

B. Kinematic control

The kinematic model of skid-steered vehicles in path coordinates from Eq. 9 can now be used to develop a path following control law. Since the derived kinematic model is consisting of error coordinates, it is convenient to use the Lyapunov stability theory (some basics can be found in [Khalil and Grizzle, 1996] and [Slotine et al., 1991]), and to propose the Lyapunov function candidate

$$V = \frac{1}{2} \left(x_e^2 + y_e^2 + \frac{1}{\sigma} |\sin(\theta_e - \psi(y_e, v_x))| \right), \quad (10)$$

where σ is a positive parameter. The function

$$\psi(y_e, v_x) = -\text{sign}(v)\theta_a \tanh y_e \quad (11)$$

is chosen as in [Micaelli and Samson, 1993] and [Soetanto et al., 2003], in order to ensure smooth transient manoeuvres, having the property $\psi(0, v_x) = 0$, and being bounded, together with its first time derivative. Here, θ_a is a positive parameter chosen as $\pi/4$.

The function $\psi(y_e, v_x)$ makes sure that the manoeuvres of the vehicle approaching the path remain smooth, tending to a maximum heading of θ_a . By choosing a different function or a different parameterization we change the controller sensitivity. For instance, if θ_a would be chosen as $\pi/2$, the controller would become more sensitive, i.e. aggressive. If no function $\psi(y_e, v_x)$ is defined, however, the controller becomes too sensitive, and the controller performance drops.

The Lyapunov candidate function time derivative is now

$$\begin{aligned}\dot{V} &= \frac{1}{\sigma} \frac{\sin(\theta_e - \psi(y_e, v_x)) \cos(\theta_e - \psi(y_e, v_x))}{|\sin(\theta_e - \psi(y_e, v_x))|} \cdot \\ &(\dot{\theta}_e - \dot{\psi}(y_e, v_x)) + \\ &x_e(v_x \cos \theta_e + x_{ICR}\omega \cos \theta_e + x_{ICR}\omega \sin \theta_e - \dot{s}) + \\ &y_e v_x \sin \theta_e - y_e x_{ICR}\omega \cos \theta_e.\end{aligned}\quad (12)$$

In order for \dot{V} to be negative semi-definite, the following expressions need to be defined:

$$\begin{aligned}\dot{s} &= v_x \cos \theta_e + x_{ICR}\omega \sin \theta_e + \gamma x_e, \\ \dot{\theta}_e &= \dot{\psi}(y_e, v_x) + \frac{|\sin(\theta_e - \psi(y_e, v_x))|}{\sin(\theta_e - \psi(y_e, v_x)) \cos(\theta_e - \psi(y_e, v_x))} \cdot \\ &\left(-\sigma y_e v_x \sin \theta_e + \sigma y_e x_{ICR}\omega \cos \theta_e - \right. \\ &\left. \zeta(\theta_e - \psi(y_e, v_x))^2 \right),\end{aligned}\quad (13)$$

where γ and ζ are positive parameters, and $\theta_e - \psi(y_e, v_x) \neq \pm(2k+1)\frac{\pi}{2}, k \in \mathbb{N}$. The latter expression is important to be considered when defining initial conditions. This condition is violated if:

- $|\theta_e| \geq \theta_a (= \frac{\pi}{4})$ and y_e needs to be large (giving $|\psi(y_e, v_x)| \approx \theta_a$ in the extreme case), which means that the robot is diverging from the path, or
- $\theta_e \approx \pm(2k+1)\frac{\pi}{2}$ and $\psi(y_e, v_x) \approx 0$, which means that the robot is on the path, but with a wrong heading.

Because of the controller's convergence, this shouldn't happen if the initial conditions are defined correctly. However, to account for unpredictable problems, such as numerical instabilities or communication errors, which could lead to divergence, it is important to consider these effects in the controller implementation. In our open-source implementation of this controller (<https://github.com/cogsys-tuebingen/gerona>) we limit all the angles and angle differences, as well as the commanded rotational velocity. In this way, even unpredictable errors in a real system are caught.

Keeping in mind that $\theta = \theta_e + \theta_t$, it is now possible to control the dynamics of the moving reference frame on the path, and the rotation of the vehicle in path coordinates, i.e. Eq. 13 presents the path following control for skid-steered vehicles, based on the ICR kinematic model in path coordinates.

C. Stability Analysis

In this section we analyse the convergence properties of the closed loop system, when Eq. 13 is applied to Eq. 9.

Proposition 1. *Let us assume that the linear velocity follows the desired profile, does not tend to zero, as time t tends to infinity, and is bounded, together with its first derivative. The path curvature is also bounded, where $c(s) \in [c_{min}, c_{max}]$, with c_{max} being the maximum, and c_{min} the minimum allowed curvature. By applying the control from Eq. 13 to the model in Eq. 9, the error coordinates $x_e(t)$, $y_e(t)$, and $\theta_e(t)$ asymptotically tend to zero, as t goes to infinity. This means that the vehicle will converge to the geometric reference path.*

Proof. The first derivative of the proposed Lyapunov function candidate defined in Eq. 12 will become negative semi-definite when the control from Eq. 13 is applied, which yields

$$\dot{V} = -\gamma x_e^2 - \frac{\zeta}{\sigma} (\theta_e - \psi(y_e, v_x))^2 \leq 0. \quad (14)$$

This means that the Lyapunov function candidate defined in Eq. 10 is monotonically decreasing, hence it reaches a finite limit as the time tends to infinity. This means that x_e , y_e and $\theta_e - \psi(y_e, v_x)$ are also bounded. Since $\theta_e - \psi(y_e, v_x)$ is bounded, and $\psi(y_e, v_x)$ is defined as bounded, then θ_e is bounded as well. By experimentally identifying a bounded x_{ICR} , the lateral velocity v_y is also bounded. Since v_x and $\psi(y_e, v_x)$ are also chosen to be bounded, and θ_e is bounded, \dot{s} and $\dot{\theta}_e$ are also bounded. This implies that \dot{x}_e and \dot{y}_e are bounded as well. It can be seen that

$$\ddot{V} = -2\gamma x_e \dot{x}_e - 2\frac{\zeta}{\sigma} (\theta_e - \psi(y_e, v_x)) (\dot{\theta}_e - \dot{\psi}(y_e, v_x)) \quad (15)$$

is bounded, since $\dot{\psi}(y_e, v_x)$ is bounded as well. Because of \ddot{V} being bounded, \dot{V} is uniformly continuous. Now, by using Barbalat's lemma (see [Slotine et al., 1991] or [Khalil and Grizzle, 1996] for details), \dot{V} converges to zero, as t goes to infinity, hence $\theta_e - \psi(y_e, v_x)$ and x_e tend to zero as well. Since the boundedness of $\dot{\theta}_e - \dot{\psi}(y_e, v_x)$ can be derived, $\dot{\theta}_e - \dot{\psi}(y_e, v_x)$ is uniformly continuous and converges to zero. This means that the right hand side of equation

$$\dot{\theta}_e - \dot{\psi}(y_e, v_x) = \frac{|\sin(\theta_e - \psi(y_e, v_x))|}{\sin(\theta_e - \psi(y_e, v_x)) \cos(\theta_e - \psi(y_e, v_x))} \cdot \left(-\sigma y_e v_x \sin \theta_e + \sigma y_e x_{ICR} \omega \cos \theta_e - \zeta(\theta_e - \psi(y_e, v_x))^2 \right), \quad (16)$$

needs to tend to zero as well. The two-sided limit

$$\lim_{\theta_e - \psi(y_e, v_x) \rightarrow 0} \frac{|\sin(\theta_e - \psi(y_e, v_x))|}{\sin(\theta_e - \psi(y_e, v_x)) \cos(\theta_e - \psi(y_e, v_x))} \quad (17)$$

does not exist, in this case the convergence needs to be analysed from the left and from the right. To simplify the notation, if we replace $\theta_e - \psi(y_e, v_x)$ with u , we will have

$$\begin{aligned} \lim_{u \rightarrow 0^-} \frac{|\sin u|}{\sin u \cos u} &= \lim_{u \rightarrow 0^-} \frac{1}{\cos u} \lim_{u \rightarrow 0^-} \frac{|\sin u|}{\sin u} \\ &= \lim_{u \rightarrow 0^-} \frac{|\sin u|}{\sin u} = -1, \end{aligned} \quad (18)$$

and similarly

$$\lim_{u \rightarrow 0^+} \frac{|\sin u|}{\sin u \cos u} = 1. \quad (19)$$

Since $\theta_e - \psi(y_e, v_x)$ tends to zero, the expression $\zeta(\theta_e - \psi(y_e, v_x))^2$ tends to zero as well. Now the expressions $E_1 = \sigma y_e x_{ICR} \omega \cos \theta_e$ and $E_2 = -\sigma y_e v_x \sin \theta_e$ need to tend to zero. Since we assume that the velocity profile follows the desired one and does not tend to zero, in order for E_2 to tend to zero, y_e or θ_e needs to tend to zero, or both.

If we assume that only θ_e tends to zero, the only way for E_1 to tend to zero is that ω tends to zero, which then means that $\dot{\theta}_e$ and \dot{s} tend to zero, as well. Now it can be seen that \dot{s} cannot tend to zero, since its first term $v_x \cos \theta_e$ needs to be nonzero, because of the nonzero velocity profile. Hence, the assumption that y_e does not tend to zero is invalid.

If we now assume that only y_e tends to zero, both E_1 and E_2 tend to zero. Furthermore, by recalling Eq. 11, we can see that ψ tends to zero as well. This also implies that θ_e tends to zero. Now it becomes clear that both y_e and θ_e need to tend to zero. Since all the error coordinates (x_e, y_e, θ_e) converge to zero, as t tends to infinity, the vehicle will asymptotically converge to the reference path. \square

IV. SPEED CONTROL

Recalling Eq. 3 and the expression for curvature using linear and angular velocity $c = \omega/v$, the reachable curvature for a skid-steered vehicle can be expressed as

$$c = \frac{\alpha_r V_r - \alpha_l V_l}{\sqrt{(\alpha_r y_{ICRl} V_r - \alpha_l y_{ICRr} V_l)^2 + (-\alpha_r x_{ICR} V_r + \alpha_l x_{ICR} V_l)^2}}. \quad (20)$$

If we want to forbid turning on the spot while driving, similarly as in [Indiveri et al., 2007], we need to restrict the wheel speeds to be of the same sign, limited with a maximum value V_m , i.e. $V_l, V_r \in [0, V_m]$.

To find the curvature extrema, we need to observe two cases. In the first case we have $V_l = 0$, and $V_r = V_m$, and the maximum curvature can be derived as

$$c_{max} = \frac{1}{\sqrt{y_{ICRl}^2 + x_{ICR}^2}}. \quad (21)$$

In this case, the vehicle's speed can be computed as

$$v_x = \frac{\alpha_r y_{ICRl} V_m}{y_{ICRl} - y_{ICRr}}. \quad (22)$$

In the second case, we have $V_r = 0$, and $V_l = V_m$, and the minimum curvature is expressed as

$$c_{min} = -\frac{1}{\sqrt{y_{ICRr}^2 + x_{ICR}^2}}, \quad (23)$$

while the vehicle's speed is

$$v_x = -\frac{\alpha_l y_{ICRr} V_m}{y_{ICRl} - y_{ICRr}}. \quad (24)$$

When planning the reference path, it should be ensured that $c(s) \in [c_{min}, c_{max}]$.

If we now consider the maximum linear velocity of $v_{max} = 2.5m/s$, and the maximum curvature for a skid-steered robot of $c_{max} = \frac{1}{\sqrt{y_{ICRl}^2 + x_{ICR}^2}} = \frac{1}{\sqrt{0.39^2 + 0.28^2}} = 2.083m^{-1}$, the maximum rotational velocity is $\omega_{max} = 5.2075rad/s$. Similarly, for the maximum velocity speed of $v_{max} = 2.5m/s$, and the minimum curvature of $c_{min} = \frac{1}{\sqrt{y_{ICRr}^2 + x_{ICR}^2}} = \frac{1}{\sqrt{(-0.49)^2 + 0.28^2}} = -1.77m^{-1}$, the minimum rotational velocity is $\omega_{min} = -4.425rad/s$. Given the computed curvature limits, as well as the maximum linear velocity that the actuators can exhibit, ω_{min} and ω_{max} represent the absolute physical limitations. More details on the parameters are found in Section V. Further work on considering physical limitations in derivation of kinematic controllers and paths can be found in [Kim and Minor, 2007], [Boucher, 2016] and [Belkhouche, 2009].

If the loop is closed with the control law from Eq. 13, the curvature in the closed loop can be expressed as

$$c_{cl} = \frac{\dot{\theta}_e + c\dot{s}}{\sqrt{v_x^2 + [x_{ICR}(\dot{\theta}_e + c\dot{s})]^2}}. \quad (25)$$

The speeds of the left and right tread in the closed loop are then

$$\begin{aligned} V_l &= \frac{v_x - y_{ICRl}(\dot{\theta}_e + c\dot{s})}{\alpha_l} \\ &= \frac{v_x - y_{ICRl} \cdot c_{cl} \cdot \sqrt{v_x^2 + [x_{ICR}(\dot{\theta}_e + c\dot{s})]^2}}{\alpha_l}, \\ V_r &= \frac{v_x - y_{ICRr}(\dot{\theta}_e + c\dot{s})}{\alpha_r} \\ &= \frac{v_x - y_{ICRr} \cdot c_{cl} \cdot \sqrt{v_x^2 + [x_{ICR}(\dot{\theta}_e + c\dot{s})]^2}}{\alpha_r}. \end{aligned} \quad (26)$$

The maximum speeds of the left and the right tread can now be expressed as

$$\begin{aligned}\max\{V_l\} &= \frac{v_x + |y_{ICRl}c_{cl}| \sqrt{v_x^2 + [x_{ICR}(\dot{\theta}_e + c\dot{s})]^2}}{\alpha_l}, \\ \max\{V_r\} &= \frac{v_x + |y_{ICRr}c_{cl}| \sqrt{v_x^2 + [x_{ICR}(\dot{\theta}_e + c\dot{s})]^2}}{\alpha_r}.\end{aligned}\quad (27)$$

Since the convergence of the closed system variables has been proven, the following properties can be used:

$$\begin{aligned}\lim_{t \rightarrow \infty} c_{cl} &= \frac{c}{\sqrt{1 + (x_{ICRC})^2}}, \\ \lim_{t \rightarrow \infty} \dot{\theta}_e &= 0, \\ \lim_{t \rightarrow \infty} \dot{s} &= v_x,\end{aligned}\quad (28)$$

in order to compute Eq. 27 when $t \rightarrow \infty$. The maximum speeds of the left and the right tread are now

$$\begin{aligned}\max\{V_l\} &= \frac{v_x(1 + |y_{ICRl}c|)}{\alpha_l}, \\ \max\{V_r\} &= \frac{v_x(1 + |y_{ICRr}c|)}{\alpha_r}.\end{aligned}\quad (29)$$

Assuming that the vehicle's motors exhibit the same maximum speed, we can write $\max\{V_l\} = \max\{V_r\} = V_m$. The speed of the entire vehicle can then be expressed as

$$\begin{aligned}v_x &= \frac{\alpha_l V_m}{1 + |y_{ICRl}c|}, \\ v_x &= \frac{\alpha_r V_m}{1 + |y_{ICRr}c|},\end{aligned}\quad (30)$$

depending whether we use the left or the right tread to compute it.

Now, the following speed control scheme can be introduced:

$$\begin{aligned}\omega \geq 0 : v_x &= \begin{cases} \frac{-\alpha_r y_{ICRl} V_m}{y_{ICRr} - y_{ICRl}}, & V \geq \epsilon. \\ \frac{\alpha_r V_m}{1 + |y_{ICRr}c|}, & V < \epsilon. \end{cases} \\ \omega < 0 : v_x &= \begin{cases} \frac{\alpha_l y_{ICRr} V_m}{y_{ICRr} - y_{ICRl}}, & V \geq \epsilon. \\ \frac{\alpha_l V_m}{1 + |y_{ICRl}c|}, & V < \epsilon, \end{cases}\end{aligned}\quad (31)$$

where ϵ is a positive parameter chosen empirically as 0.5. If ϵ is chosen too small, switching between different computations of v_x could often occur, and the motion would become jerky. On the other hand, if ϵ is too large, the switching could perhaps never happen. Therefore it is necessary to choose ϵ in a way that the switch is only active if there is a larger deviation from the path, so that the robot can slow down and steer back to the path. As soon as it approaches the path again, the speed should be increased according to the switch.

As stated in [Huskić et al., 2017c], since skid-steered vehicles behave asymmetrically, depending on the mechanical construction and on the terrain structure, the speed should be adjusted accordingly. This is the reason why a distinction is made whether the vehicle is turning right or left.

If the vehicle is turning left, the right tread speed V_r is dominant, and V_m can be applied, in order to maximize the speed, and to take the actuator saturation into account. The

exhibited speed should never be greater than V_m . Furthermore, the Lyapunov function candidate from Eq. 10 is used here as a path following error measure. If the error is greater than a positive threshold ϵ , the expression from Eq. 22 is used. On the other side, if the error is smaller than ϵ , the speed of the entire vehicle is computed using Eq. 30. Similar analysis can be made if the left tread speed V_l is dominant.

By using this speed control scheme, asymmetric behaviour of the vehicle is taken into account, together with the path following error, and the actuator saturation. The vehicle will not turn on the spot while driving, and the speed is maximized.

The path following control law defined in Eq. 13, combined with the speed control defined in Eq. 31, presents an integral solution for path following of skid-steered vehicles at higher speeds on different terrain types.

V. EXPERIMENTAL EVALUATION

The proposed algorithm is implemented in C++/ROS (detailed in [Quigley et al., 2009]), and integrated into the GeRoNa framework, a generic and modular navigation framework for any wheeled mobile robot. Details about GeRoNa can be found in [Huskić et al., 2017b]. The whole framework, together with the algorithm proposed in this paper, is open-source available at: <https://github.com/cogsystuebingen/gerona>.

The experimental evaluation of the proposed algorithm consists of two parts. First, the algorithm is evaluated on a Robotnik Summit XL robot, up to its maximum speed of ≈ 2.5 m/s, on three different terrain types. The proposed approach is experimentally compared against two state-of-the-art algorithms. The first state-of-art-algorithm is the one originally proposed in [Soetanto et al., 2003], and then extended in [Indiveri et al., 2007]. We refer to it as *SLPINL*, which is an acronym of the authors' surnames. The second algorithm is the one proposed in [Pentzer et al., 2014b], which is based on the work proposed in [Pentzer et al., 2014a]. This algorithm we refer to as *PBR*, using the same naming scheme. Finally, the algorithm proposed in this paper, based on the work in [Huskić et al., 2017c], is referred to as *HBZ*.

The second part of the evaluation is done on a Segway RMP 440 robot, up to the speeds of 6m/s. Similar as with Robotnik Summit XL, an experimental comparison between *SLPINL*, *PBR* and *HBZ* is made on three different terrain types.

For both experimental evaluations, an offline estimation of the ICR parameters is made for different terrain types, similar as in [Madow et al., 2007], by using evolutionary algorithms. For the implementation, two software tools were used: CS::APEX, detailed in [Buck et al., 2016], and EvA2, detailed in [Kronfeld et al., 2010].

A. Experiments with Robotnik Summit XL

With a maximum speed of ≈ 2.5 m/s, and a weight of ≈ 50 kg, Robotnik Summit XL is a medium-sized skid-steered robot with good performance. The robot used in the experiments presented in this paper is shown in Fig. 3.

The experiments with this robot were conducted on the following terrain types, shown in Figure 4:



Fig. 3. The Robotnik Summit XL robot used in the experiments.

- (i) relatively flat, grassy terrain;
- (ii) very smooth vinyl floor;
- (iii) macadam - crushed stones mixed with dust and sand.

For each experiment, PBR and SLPINL were commanded with the same desired speed, in order to achieve similar mean speeds. The proposed approach was commanded with even higher speeds, in order to emphasize its advantages. A video demonstrating the experiments can be found at: <https://youtu.be/plaoHHEfM3o>.

Grass terrain: On the grassy terrain, a 159.83m long path was used, having long segments, and abrupt curves. The path was covering a small football field behind our institute building, hence the rectangular-like shape. For the ground truth, a fusion was made using an internal gyroscope and the odometry coming from Hall effect sensors. The localization error was measured while following a planned path at different speeds, by using CS::APEX, proposed in [Buck et al., 2016]. The resulting error at the average speed of 0.89m/s was $(x_{err}, y_{err}, \theta_{err}) = (0.015m, 0.69m, 2.9 \text{ deg})$. At the average speed of 2.15m/s, the error was $(x_{err}, y_{err}, \theta_{err}) = (0.62m, -0.65m, 3.85 \text{ deg})$.

Path following performance of SLPINL, PBR and HBZ at some representative speeds on grass can be seen in Fig. 5 and Fig. 6. A direct comparison between the three algorithms at different speeds can be observed in Fig. 7.

The mean and maximum values of the speed and error, when driving at higher speeds, can be seen in Table I. For SLPINL, when the desired speed is set to 2m/s, the maximum measured value is 2.25m/s, and the system reaches critical oscillations, where the error reaches 1.83m. Because of the inherent speed control of the algorithm, the overall mean speed is then 1.55m/s. For the case of PBR, when the desired speed is set to 2m/s, the mean speed is 1.94m/s, the error reaches 3.61m, and the system is on the border of stability. The mean error in this case is 83cm. When the HBZ controller is commanded with 2.5m/s, the mean speed is 2.15m/s. The maximum measured error is 22cm in this case, while the mean



Fig. 4. The three terrain types on which the Robotnik Summit XL was tested: grass, vinyl floor, macadam.

error is 7cm.

The Robotnik Summit XL robot can theoretically reach a velocity of 3m/s, if it moves on a perfect straight line. Since there has to be always some correction control, i.e. the vehicle needs to slightly turn, then the left or the right wheels need to drive faster than the commanded speed, which is not feasible. This is the reason why the practical maximum speed lies at $\approx 2.5m/s$, so that the left and right wheels can drive at speeds up to 3m/s, in order to steer.

Vinyl floor: On vinyl and macadam floors, a lemniscate was used as a reference path, overlaid five times, so the robot needed to follow the curve five times in a row in order to finish one run. This way, the repeatability of the algorithms was tested. Furthermore, the curvature of the lemniscate was chosen to be very high, when compared to the dimensions of the robot, in order to make the path following task more challenging.

This floor type is typical for indoor environments, having a very smooth surface, which makes the motion of the vehicle prone to slipping. For the ground truth in this case, three special pillars were used as landmarks for localization. The pillars were easily recognized with a 3D laser scanner mounted on top of the robot, and a Kalman filter was used to estimate their positions, if some of them were not seen in the current

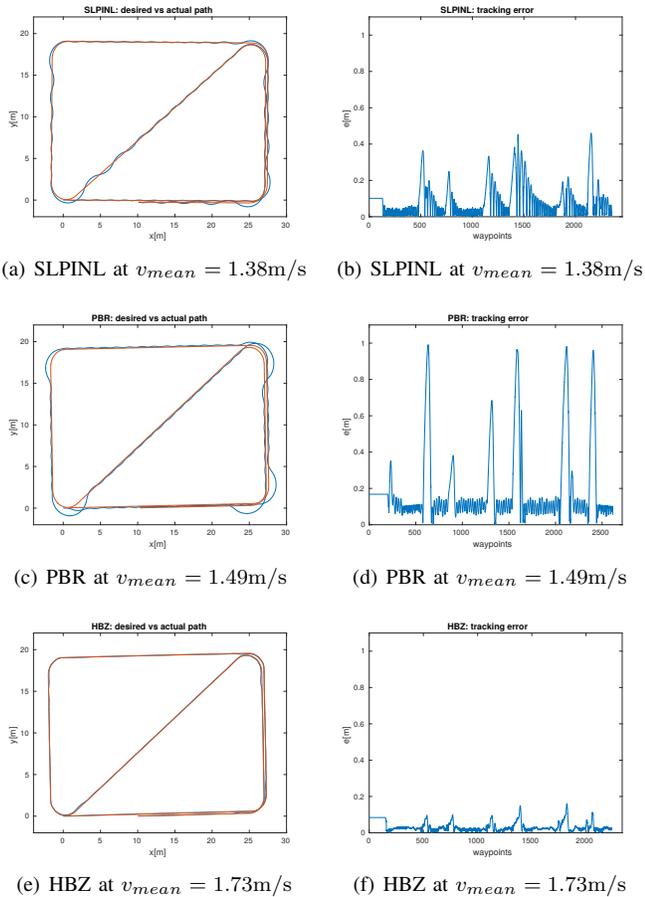


Fig. 5. Path following performance at representative speeds with Robotnik Summit XL on grass, following a 159.83m long path.

time step. Otherwise, a simple triangulation was used to determine the robot's pose.

Some results at representative speeds can be seen in Fig. 8 and Fig. 9.

The results in Fig. 8 show the performance of SLPINL and PBR, when commanded with a desired speed of 1m/s. In both of the cases, the error reaches the value of $\approx 31\text{cm}$. The HBZ controller is commanded with 2m/s, having the average speed of 1.34m/s, and the maximum error stays below 10cm. In Fig. 9 the results are seen, when SLPINL and PBR are commanded with 1.5m/s. In both cases, the error reaches values above 1m. The experiments were conducted in a closed room, and hence had to be stopped, in order to avoid colliding with walls. The HBZ controller was commanded with 2.5m/s, reaching speeds of 2.3m/s, and having an average speed of 1.64m/s. The maximum error in this case reaches 46.6cm, while the average error is 8.5cm.

Macadam terrain: This terrain type was particularly difficult, since the surface made of crushed stones and dust move under the robot wheels, which increases the effect of skidding. The ground truth was computed in the same way as for the vinyl terrain type, and the reference path was the same, as well. Path following results at representative speeds, together with the error profiles, can be observed in Fig. 10

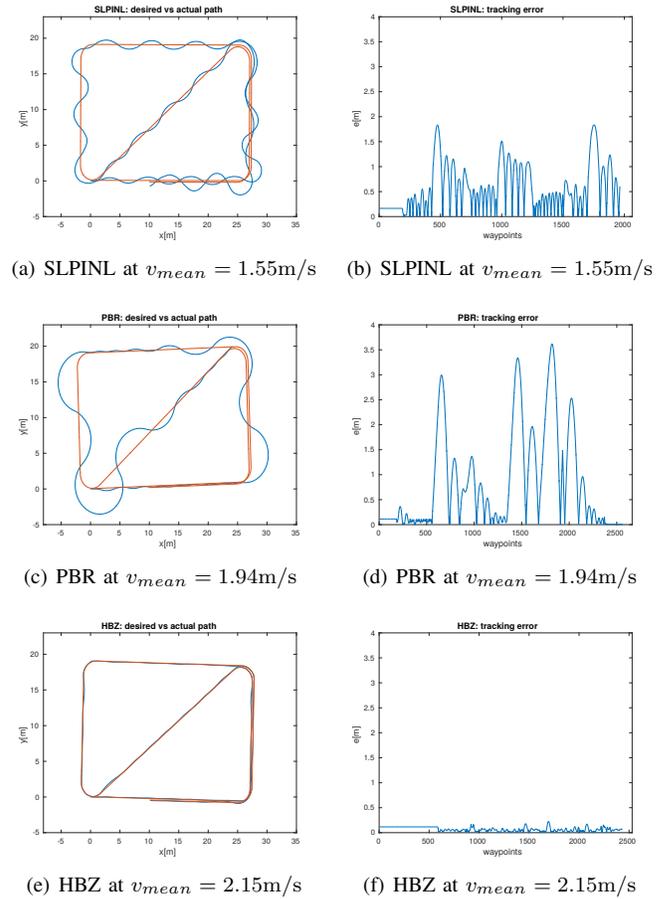


Fig. 6. Path following performance at higher speeds with Robotnik Summit XL on grass, following a 159.83m long path.

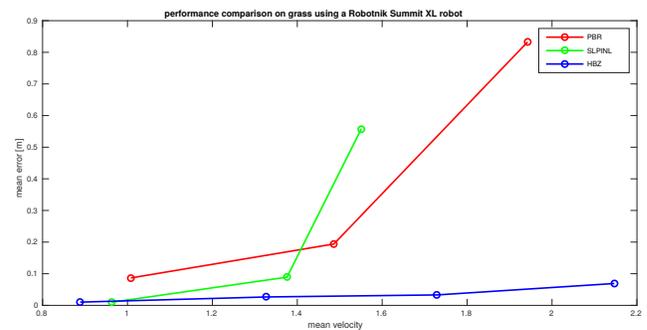


Fig. 7. Performance comparison on grass with the Robotnik Summit XL robot.

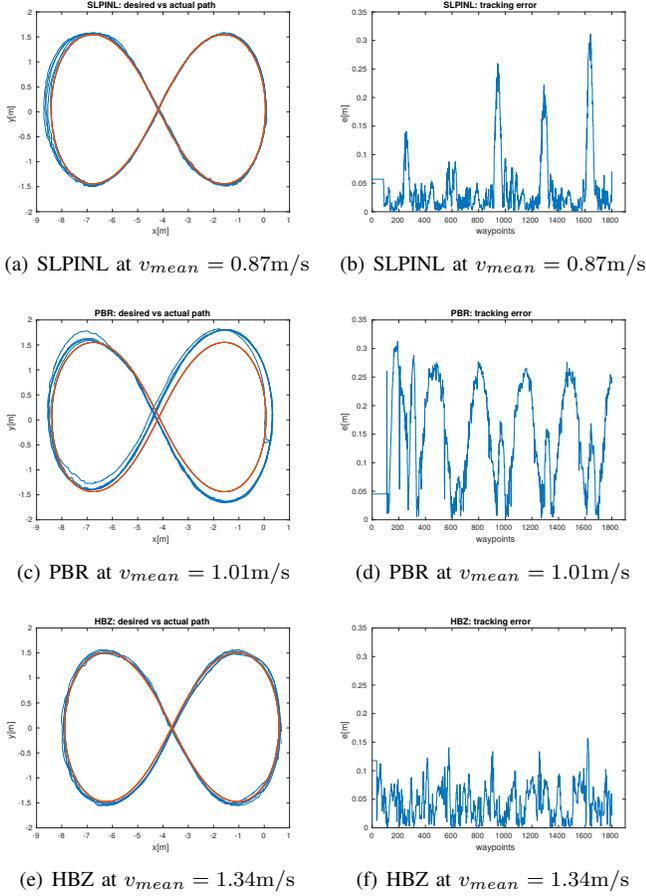


Fig. 8. Path following performance at representative speeds with Robotnik Summit XL on vinyl, following a 110.77m long path.

and Fig. 11.

In Fig. 10(e) a localization error as a sudden jump of 88cm can be noticed, while driving with HBZ, but the robot does not lose track. After the localization jump, the maximum path following error remains below 14cm, and the mean error is 5.2cm. For the case of SLPINL and PBR in Fig. 10, there were no localization jumps, but the path following error reaches the value of 42.7cm with SLPINL, and 45.3cm with PBR. SLPINL and PBR were commanded with 1m/s, and HBZ with 1.5m/s.

While driving at higher speeds, which can be observed in Fig. 11, SLPINL loses track when commanded with 1.5m/s, where the mean speed is then 0.94m/s. The error reaches 1.86m in this case, and only one lap is performed, instead of five. The PBR controller was also commanded with 1.5m/s, having the mean speed of 1.48m/s. The maximum error is 2.12m, and the mean error 85.6cm. The HBZ controller was commanded with 2.5m/s, and the mean speed was 1.68m/s. The maximum error is 95.1cm, and the mean error 18.3cm.

Parameters: In all of the presented experiments conducted with Robotnik Summit XL, the parameters for the HBZ controller were as follows: $\gamma = 8$, $\zeta = 40$, $\sigma = 1$. The parameter γ determines the speed of the virtual vehicle depending on the longitudinal error. This parameter was set to

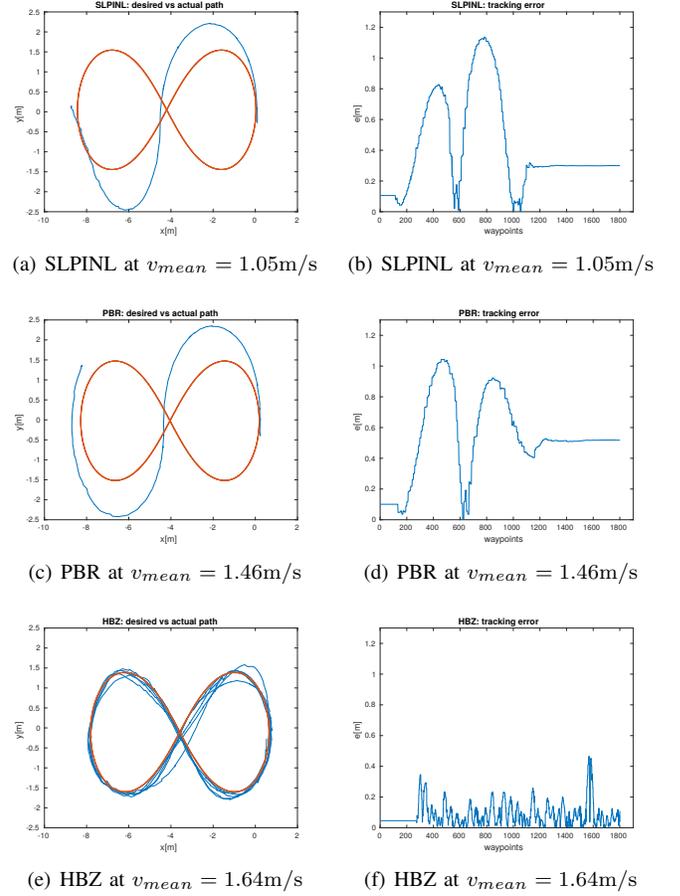


Fig. 9. Path following performance at higher speeds with Robotnik Summit XL on vinyl, following a 110.77m long path.

a high value, in order for the robot to move faster when there is a larger longitudinal deviation. The parameter ζ influences the rotation of the robot, depending on the angle difference between the current robot orientation in path coordinates θ_e , and the desired angle ψ . This parameter was also set to a high value, since the angle difference is usually small, yet it is an important control factor. Finally the parameter σ influences the rotation of the robot depending on the lateral error. Both terms directly influenced by this factor can contain larger values, since they are multiplied by the current linear and angular velocity of the robot. Therefore, this factor needs to be chosen in the range of small numbers. We have chosen a high value in this range for the control to be more sensitive to lateral error.

The ICR parameters identified for grass are

$$(x_{ICR}, y_{ICRl}, y_{ICRr}, \alpha_l, \alpha_r)_{grass} = (0.28, 0.39, -0.49, 0.9, 0.91). \quad (32)$$

The ICR parameters for vinyl are

$$(x_{ICR}, y_{ICRl}, y_{ICRr}, \alpha_l, \alpha_r)_{vinyl} = (0.26, 0.49, -0.35, 0.8, 0.83). \quad (33)$$

The ICR parameters for macadam are identified as

$$(x_{ICR}, y_{ICRl}, y_{ICRr}, \alpha_l, \alpha_r)_{macadam} = (0.22, 0.48, -0.47, 0.88, 0.9). \quad (34)$$

TABLE I
EXPERIMENTS WITH THE ROBOTNIK SUMMIT XL ROBOT ON GRASS, AT THE MAXIMUM SPEED AT WHICH EACH ALGORITHM COULD PERFORM. FOR THE CASE OF HBZ, THE MAXIMUM SPEED OF THE ROBOT IS REACHED.

Algorithm	Mean Speed [m/s]	Max Speed [m/s]	Mean Error [m]	Max Error [m]
SLPILN	1.55	2.25	0.56	1.83
PBR	1.94	2.29	0.83	3.61
HBZ	2.15	2.53	0.07	0.22

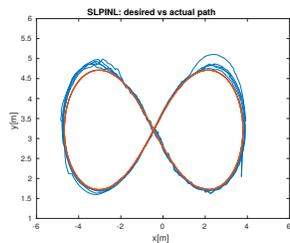
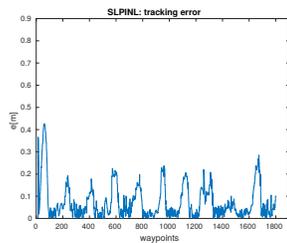
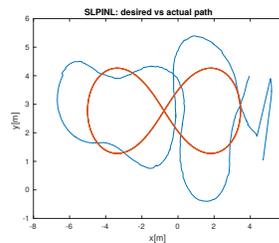
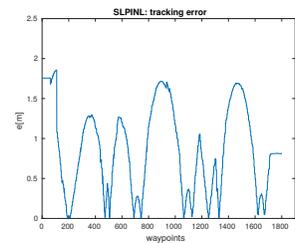
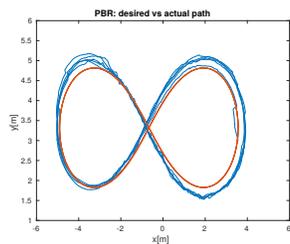
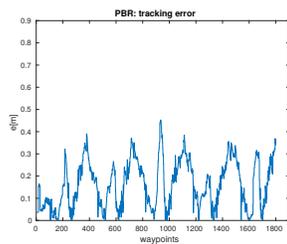
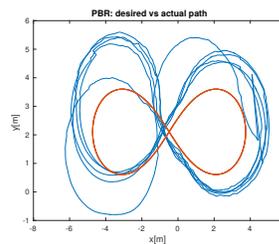
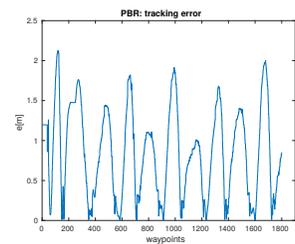
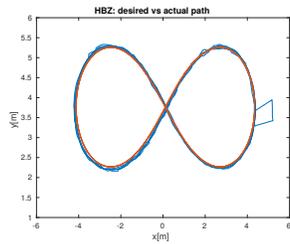
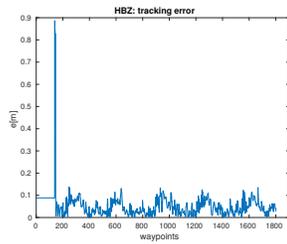
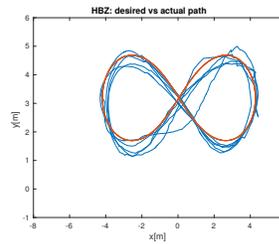
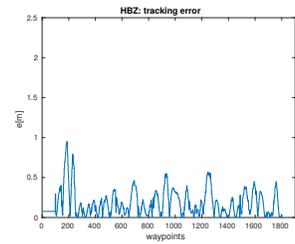
(a) SLPILN at $v_{mean} = 0.87\text{m/s}$ (b) SLPILN at $v_{mean} = 0.87\text{m/s}$ (a) SLPILN at $v_{mean} = 0.94\text{m/s}$ (b) SLPILN at $v_{mean} = 0.94\text{m/s}$ (c) PBR at $v_{mean} = 1.0\text{m/s}$ (d) PBR at $v_{mean} = 1.0\text{m/s}$ (c) PBR at $v_{mean} = 1.48\text{m/s}$ (d) PBR at $v_{mean} = 1.48\text{m/s}$ (e) HBZ at $v_{mean} = 1.1\text{m/s}$ (f) HBZ at $v_{mean} = 1.1\text{m/s}$ (e) HBZ at $v_{mean} = 1.68\text{m/s}$ (f) HBZ at $v_{mean} = 1.68\text{m/s}$

Fig. 10. Path following performance at representative speeds with Robotnik Summit XL on macadam, following a 110.77m long path.

For every experiment, the appropriate parameter set was used, depending on the terrain. However, it is worth to examine the influence of the ICR parameters to the path following performance. In order to do so, an additional experiment on vinyl was made, where the HBZ controller was commanded with 2m/s to follow the five time overlaid lemniscate, like before. This time, three different runs were made, every time with different ICR parameters. The results can be seen in Table II.

By observing Table II it is obvious that the maximum error is significantly smaller when vinyl parameters are used, but the mean error in all three cases is less than 5cm. This leads to the conclusion that the HBZ controller is robust to changes

Fig. 11. Path following performance at higher speeds with Robotnik Summit XL on macadam, following a 110.77m long path.

of the ICR parameters. The reason for this lies in the fact that the ICR parameters, being estimated offline, are a good approximate guess of the kinematics, and implicitly of the dynamics. The fast feedback control then drives the system in order to minimize the error, even if the ICR parameters are not optimally chosen. Furthermore, the most influential ICR parameter for the feedback control is x_{ICR} , which in this example has similar values for all three runs, and the control parameters γ , ζ , and σ are kept the same.

In [Huskić et al., 2017a], an extension of the HBZ controller was evaluated using a Robotnik Summit XL robot for person following. The feedback control, however, is the same as the one presented in this paper. Many long-run tests

TABLE II
EVALUATION OF THE HBZ CONTROLLER ON VINYL FLOOR WITH THREE DIFFERENT ICR PARAMETER SETS.

Parameters	Mean Speed [m/s]	Max Speed [m/s]	Mean Error [cm]	Max Error [cm]
grass	1.45	1.96	4.9	44.8
vinyl	1.34	1.74	4.5	15.6
macadam	1.41	1.92	4.6	34.2

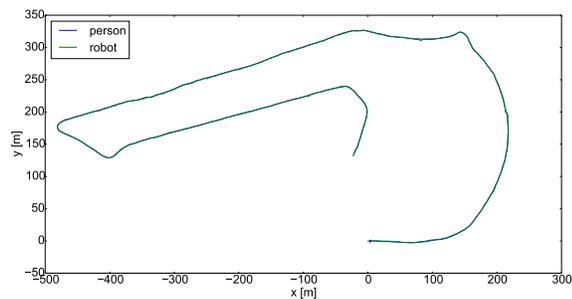


Fig. 12. An extension of the HBZ controller tested on a 1791.6m long path, over various terrain types, using only one parameter set.

on various different terrain types and under many different conditions, such as on ice and snow, were performed to further evaluate the approach. In all of those experiments, the grass parameters shown in Eq. 32 were used, together with the previously mentioned controller parameters: $\gamma = 8$, $\zeta = 40$, $\sigma = 1$. This further emphasizes the robustness of the approach. A video demonstrating the experiments in various terrain conditions with the same parameter set can be found at: <https://youtu.be/4O7twdWFm4s>. In Fig. 12 a 1791.6m long trajectory excerpt from the video can be seen, using the same parameter set on many different terrain types, including asphalt, grass, gravel, dirt, etc.

B. Experiments with Segway RMP 440

The Segway RMP 440 is a robust and relatively big skid-steered robot, weighing around 150kg and reaching speeds of $\approx 8\text{m/s}$. The robot used for the experiments presented in the following text can be seen in Fig. 14.

The experiments were conducted on three different terrain types, shown in Figure 13:

- (i) relatively flat mixture of grass and dirt;
- (ii) very uneven terrain with relatively high grass;
- (iii) asphalt partially covered with crushed stones.

On the first terrain type, a detailed comparison between SLPINL, PBR and HBZ is made at speeds up to 2m/s. On the second terrain type, the HBZ controller is thoroughly tested at speeds up to $\approx 6\text{m/s}$. On the third terrain type, some experiments with the HBZ controller were made at moderate



Fig. 13. The three terrain types on which the Segway RMP400 was tested: grass, uneven ground, old asphalt

speeds. In all of the experiments conducted with the Segway RMP 440 robot, the following parameter set was used:

$$(x_{ICR}, y_{ICRl}, y_{ICRr}, \alpha_l, \alpha_r)_{RMP440} = (0.6, 0.74, -0.7, 0.96, 0.94). \quad (35)$$

The controller parameters for the terrain (i) and (iii) were chosen as $\gamma = 10$, $\zeta = 15$, $\sigma = 0.1$, whereas for the terrain type (ii) they were chosen as $\gamma = 10$, $\zeta = 5$, $\sigma = 0.1$. When comparing these parameters to the ones used for Summit XL, it can be seen that γ is slightly higher, thus the linear velocity of the virtual vehicle is now more sensitive to the

longitudinal error. In this way, RMP 440 is commanded to drive faster, since it is a faster robot than Summit XL. On the other hand, both ζ and σ are lower than the ones used for Summit XL, since RMP 440 has a higher angular velocity, and if ζ and σ would be higher, the control behaviour would be too aggressive and might lead to instability. This is also why these two parameters are even lower for the very uneven terrain type (ii).

The ground truth for all the experiments with the Segway RMP 440 robot was determined by fusing the odometry measurements with a high grade fiber optic gyroscope. A video demonstrating the conducted experiments can be found at: <https://youtu.be/SCv0bu-Zuq8>.

Grass/Dirt terrain: On this terrain type, a comparison between SLPINL, PBR and HBZ is made on two different reference paths: a three times overlaid oval-like path, and a three times overlaid lemniscate. The terrain was relatively flat, with some parts covered with short grass, and some parts made of dirt mixed with gravel. Some results obtained on an oval reference path, including the desired and the actual path, as well as the error profiles, can be observed in Fig. 15 and Fig. 16. A direct comparison between the three algorithms at different speeds can be seen in Fig. 17.

As it can be observed in Fig. 15, Fig. 16, and Fig. 17, the SLPINL controller is very accurate at speeds lower than 1m/s, while the accuracy decreases, as the speed increases above 1m/s. At 1.53m/s, the error reaches the maximum of 1.3m, while the mean error is 38.4cm. The PBR controller is fairly accurate at lower speeds, and the error increases relatively slowly, as the speed increases. However, the error increases faster, when the speed is higher than 1.5m/s. When driving with a mean speed of 1.49m/s, the mean error is 17.6cm. At 1.98m/s, the error reaches its maximum of 1.29m, while the mean error is 39.4cm. The HBZ controller has a mean error of 7.3cm, when driving at 1.6m/s. When driving with a mean speed of 2m/s, the mean error is 13.1cm, and the maximum error 50cm.

Further experiments on the same terrain were conducted by using a lemniscate reference path overlaid three times, such that one run includes three laps around the path. Some results are shown in Fig. 18 and Fig. 19. A direct comparison of SLPINL, PBR and HBZ on a lemniscate at different speeds can be observed in Fig. 20.

By observing Fig. 18, Fig. 19, and Fig. 20, similar conclusions as in the case of the oval-shaped path can be made. At speeds lower than 1m/s, the SLPINL controller has a higher accuracy than the PBR controller. As the speed increases, the error quickly increases as well. At 1.32m/s, the maximum error is 81.7cm, and the mean error 28.4cm. When driving with the mean speed of 1.36m/s, the maximum error reaches 1.1m, and the mean error is 42.1cm. The reason for such a big error increase at similar mean speeds, is that the desired speeds are different.

In the first case, the commanded speed is 1.5m/s, and in the second case 1.75m/s. Because of the inherent speed control of the SLPINL algorithm, the actual speed never reaches the commanded input. Obviously, for this path, the speed of ≈ 1.3 m/s is the maximum at which this algorithm can perform,



Fig. 14. The Segway RMP 440 robot used in the experiments.

since the value is not exceeded, even if the commanded speed increases. The PBR controller has a fair accuracy at lower speeds, but when the speed is increased, the error increases slower than in the case of the SLPINL controller. When driving at 1.74m/s, the error reaches the maximum of 1.3m, where the mean error is 45.9cm. The HBZ algorithm has a very high accuracy through the whole spectrum of speeds. When driving with a mean speed of 1.6m/s, the mean error is 7.9cm, while at 2m/s, the mean error increases to 16.3cm.

In order to test the sensitivity to parameter changes, an additional experiment was conducted on this terrain by following the oval path, in a similar way as before. This time, all the parameters were the same as for the Robotnik Summit XL robot on grass, as described in Section V-A. The desired and the actual path, together with the error profile, can be seen in Fig. 21. When driving with a mean speed of 0.81m/s, the maximum error is 19.3cm, and the mean error 5.9cm. The results are similar to the ones obtained with HBZ at 1.6m/s, with the proper ICR parameter set. The fact that the HBZ algorithm was in this case transferred from one vehicle to another very different one, without any parameter tuning, and having a relatively small tracking error, demonstrates the robustness of the approach.

Uneven ground: Further experiments were conducted at higher speeds on a very uneven ground with relatively high grass. A lemniscate path was used as a reference again, this time, however, having a larger scale. The path was partially covering a shallow ditch in the ground, which made the experiments even more challenging. In these experiments, only the HBZ algorithm is evaluated, since all the previous experiments showed that SLPINL and PBR become rather unstable at ≈ 2 m/s.

In Fig. 22 the desired and the actual paths at some representative speeds are shown, together with the error profiles. In Fig. 23 the performance of the HBZ algorithm at different

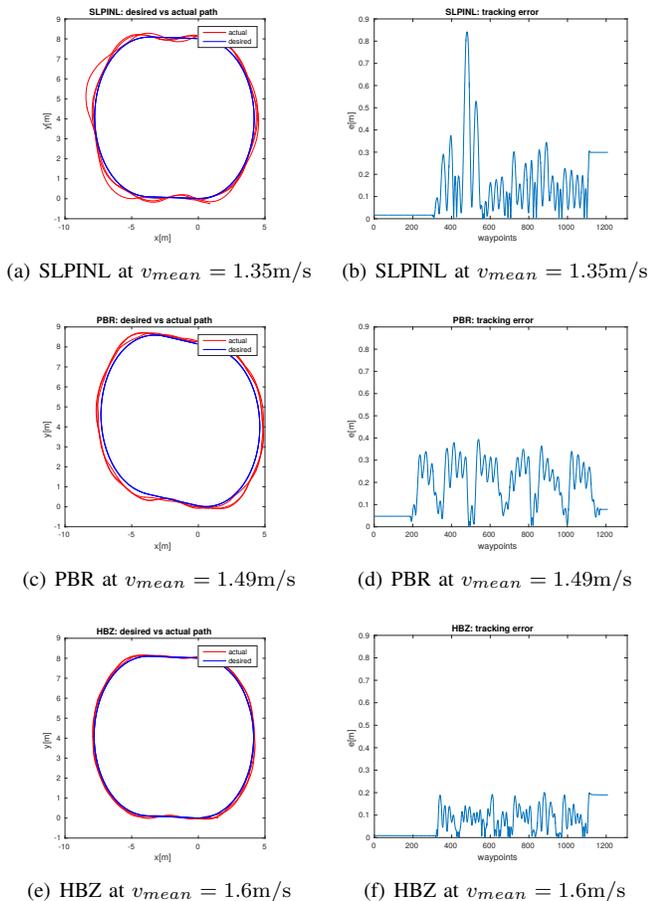


Fig. 15. Path following performance at representative speeds with Segway RMP 440 on grass/dirt, following a 98.65m long path.

speeds can be observed. When driving with a mean speed of 2.25m/s , the maximum error is 38.96cm , while the mean error is 9.8cm . If the mean speed increases to 3.86m/s , the maximum error is 1m , and the mean error 25.4cm . The main reason for the increased error at high speeds is the relatively deep ditch, which was located in the lower right part of the lemniscate. This can be observed in Fig. 22(e), since there are some visible oscillations at this spot.

Since the terrain was very difficult to cope with, the speed of 3.86m/s was the highest mean speed achieved. Driving faster on a reference path of this scale, and on this kind of a terrain, would probably lead to tipping over. In order to drive faster while staying safe as well, the terrain should be relatively flat, and the reference path should have a bigger radius of curvature. However, the experiments described here are more challenging, since the path is on the border of being feasible, both because of the terrain, and the curvature of the path, relative to the speed of the vehicle.

If the speed control from Eq. 31 is properly set, the robot is limited by the maximum velocity, even if the commanded speed is higher than being feasible. If the commanded speed is 3.5m/s on the reference lemniscate on this uneven terrain, the mean speed is 3.12m/s , since the speed control regulates the motors so that the tracking error stays as small as possible.

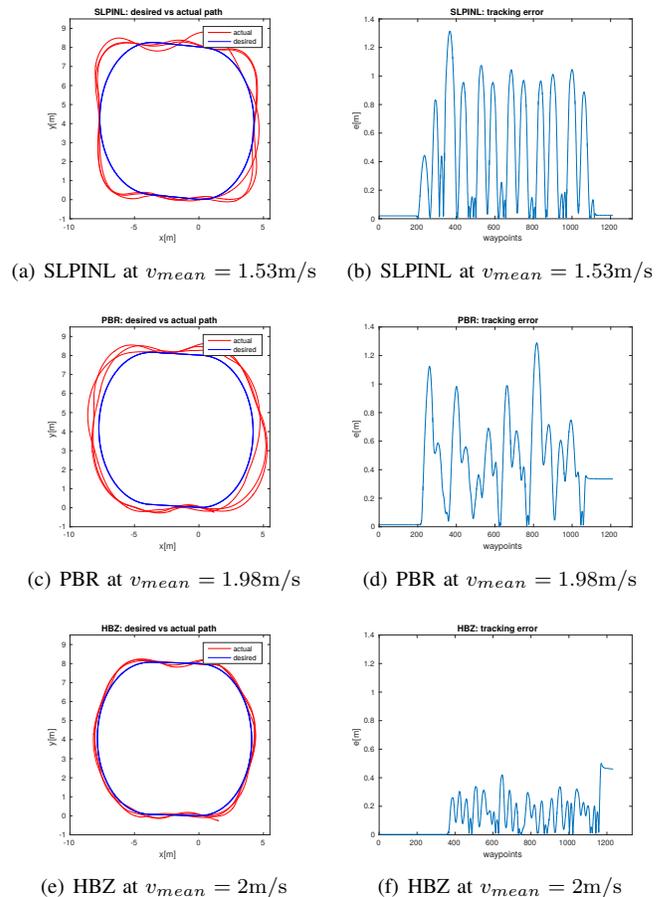


Fig. 16. Path following performance at higher speeds with Segway RMP 440 on grass/dirt, following a 98.65m long path.

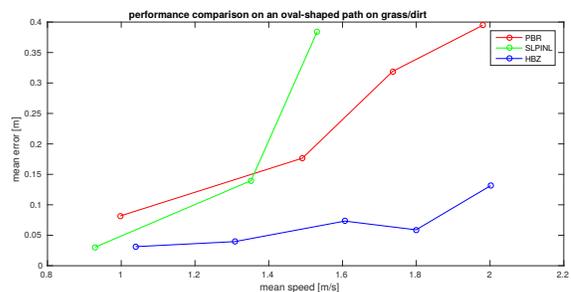


Fig. 17. Performance comparison on an oval-shaped path on grass/dirt using the Segway RMP 440 robot.

Furthermore, if the curvature of the path increases, the speed decreases. If the commanded speed is 6m/s , the maximum speed will reach 6.16m/s indeed, but Eq. 31 will reduce the overall mean speed to 3.19m/s . This means that the vehicle is allowed to accelerate up to the allowed maximum, if the tracking error and the curvature are small enough, otherwise, the speed is reduced to a lower value. The comparison between these two cases can be seen in Fig. 24.

This effect can be best observed in Fig. 24(g) and Fig. 24(h). When the desired speed is 3.5m/s , the speed profile is smooth.

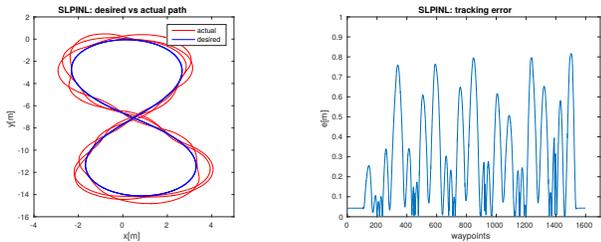
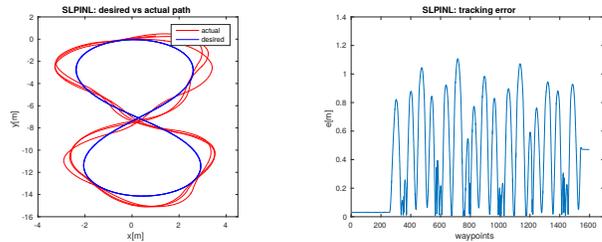
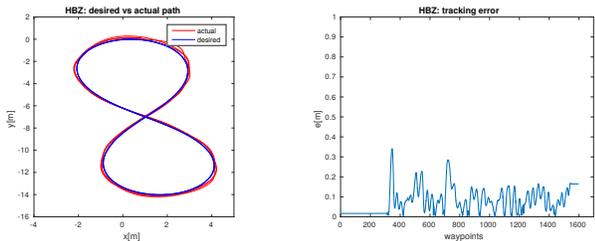
(a) SLPINL at $v_{mean} = 1.32\text{m/s}$ (b) SLPINL at $v_{mean} = 1.32\text{m/s}$ (c) PBR at $v_{mean} = 1.49\text{m/s}$ (d) PBR at $v_{mean} = 1.49\text{m/s}$ (e) HBZ at $v_{mean} = 1.6\text{m/s}$ (f) HBZ at $v_{mean} = 1.6\text{m/s}$

Fig. 18. Path following performance at representative speeds with Segway RMP 440 on grass/dirt, following a 98.65m long path.

When commanding 6m/s , the peaks of $\approx 6\text{m/s}$ show up while driving on the straight line segments of the lemniscate, and the rest of the time the speed remains around $\approx 3\text{m/s}$. When entering a straight line segment and accelerating, the vehicle would sometimes even lose contact between the front wheels and the ground, and drive only on the rear wheels for short periods. The angular velocities are similar in the both cases, and the error increases in the case of the high commanded speed, as it can be observed in Fig. 24(d).

Old asphalt: Some further experiments at moderate speeds were conducted on an old asphalt with small cracks and areas covered with crushed stones. Using a skid-steered vehicle on asphalt usually leads to unstable behaviour, since the friction between the ground and the wheels is very high. In order to skid, large forces need to be applied, which then exhibit discontinuous impulses. This can be observed as a swinging behaviour of the vehicle, which could even lead to tipping over, if e.g. the centre of mass is high. This effect is especially noticeable when the vehicle is turning on the spot.

Since the asphalt used for this experiments was covered with crushed stones at some areas, it was possible for the vehicle to exhibit a relatively smooth motion. An oval-shaped path was used as a reference, similar as before. The desired and the actual paths, the error profile, together with the angular and longitudinal velocities for one representative experiment

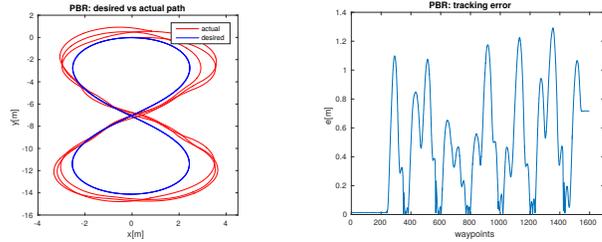
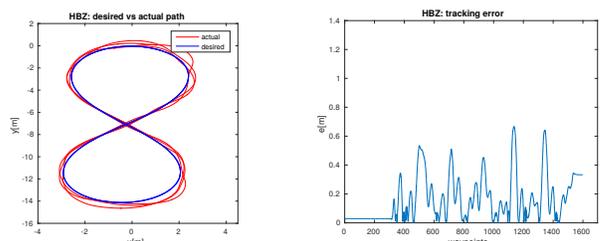
(a) SLPINL at $v_{mean} = 1.36\text{m/s}$ (b) SLPINL at $v_{mean} = 1.36\text{m/s}$ (c) PBR at $v_{mean} = 1.74\text{m/s}$ (d) PBR at $v_{mean} = 1.74\text{m/s}$ (e) HBZ at $v_{mean} = 1.99\text{m/s}$ (f) HBZ at $v_{mean} = 1.99\text{m/s}$

Fig. 19. Path following performance at higher speeds with Segway RMP 440 on grass/dirt, following a 98.65m long path.

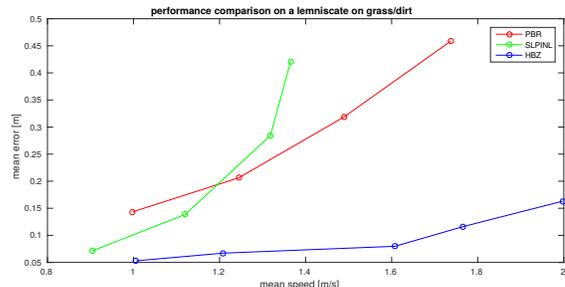


Fig. 20. Performance comparison on a lemniscate on grass/dirt using the Segway RMP 440 robot.

can be observed in Fig. 25. The mean speed in this case is 1.85m/s , the maximum error 14cm , and the mean error 6.2cm .

VI. CONCLUSIONS

In this paper, the approach proposed in [Huskić et al., 2017c] is presented in an extended manner. The theory is explained more thoroughly, especially detailing the convergence properties of the closed loop system. The experiments are extended, and additionally conducted on another vehicle with different dynamic characteristics, at speeds more than two times higher than before.

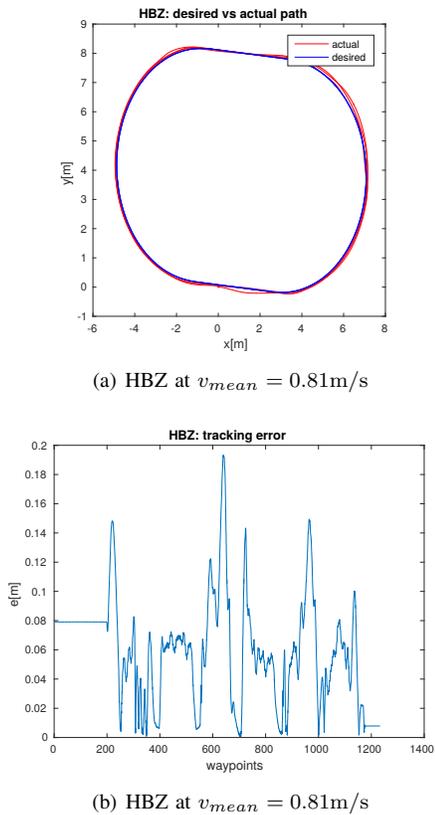


Fig. 21. Path following performance with Segway RMP 440 on grass/dirt, using the ICR parameters from the Robotnik Summit XL robot.

The proposed algorithm is a combined solution of a nonlinear control law designed for path following, and a control scheme for the longitudinal velocity. The path following control guarantees that the vehicle converges to the path. The speed control maximizes the speed, while minimizing the tracking error. Both control solutions are based on an experimentally identified kinematic model for skid-steered vehicles.

A comprehensive experimental comparison between the proposed solution and two other state-of-the-art algorithms is made, on two different vehicles and on various terrain types and paths. The experiments include two major difficulties for the path following task: rough terrain and high speeds. The maximum speed reached is 6.16m/s .

In the experimental evaluation, it is clear that the proposed approach outperforms the two state-of-the-art algorithms on both vehicles, for each terrain and path type. The difference gets more obvious as the speed increases. The SLPINL algorithm, however, shows very good performance at lower speeds. At higher speeds, the error increases rapidly. The PBR algorithm usually has a small offset even at lower speeds, since this approach relies heavily on the convergence of the Extended Kalman Filter (EKF), which is needed for the online estimation of the ICR parameters. The EKF does not necessarily find the optimal values, which leads to an offset. This approach, however, performs better than SLPINL at higher speeds, since it takes ICR parameters into account, even though it uses a simple control law for unicycles.

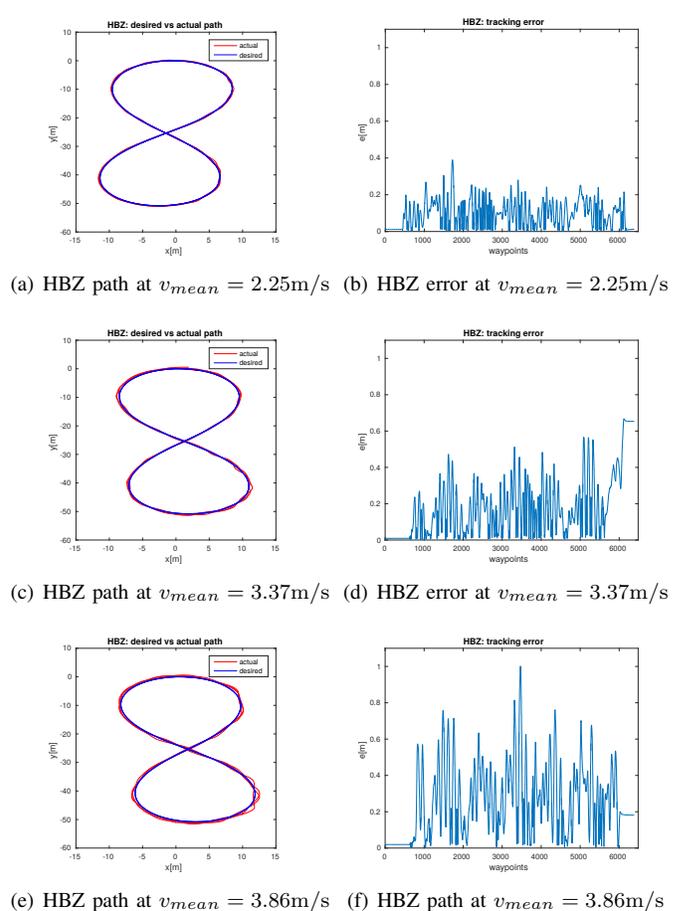


Fig. 22. Path following performance at high speeds with Segway RMP 440 on a very uneven terrain with high grass, following a 400.14m long path.

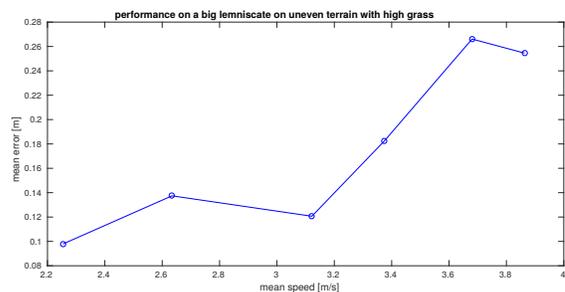


Fig. 23. HBZ performance on a very uneven terrain using the Segway RMP 440 robot.

The proposed approach shows very stable performances even at high speeds and rough terrain. In the experiments it is shown that it recovers from localization errors, and that it is robust to parameter changes. The complete solution was cloned from one vehicle to another, a very different one, without any parameter tuning, and still gave satisfying results. Furthermore, in [Huskić et al., 2017a] the proposed approach is extended for person following and obstacle avoidance, and tested with the same parameter set on various different terrain types, on large-scale paths.

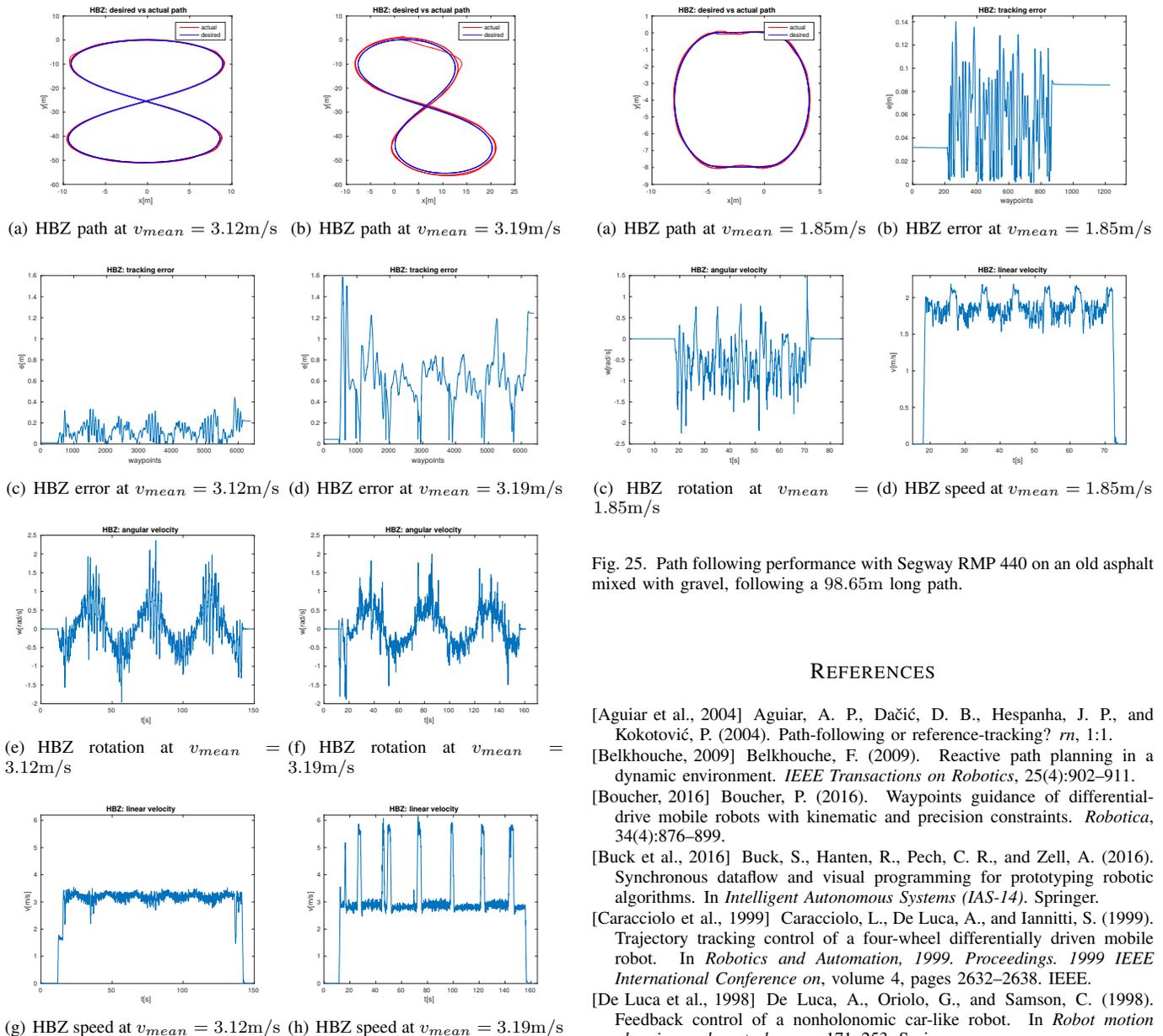


Fig. 24. Effects of the speed control when following a path with commanded speeds that are higher than feasible.

Since the model used for the control is a kinematic model and only takes the dynamics into account implicitly, it would be reasonable to extend this work by explicitly modelling the dynamics and the wheel slip. Furthermore, it would be important to examine physical limitations and safety at higher speeds.

ACKNOWLEDGEMENT

Goran Huskić would like to thank the German Academic Exchange Service (DAAD) for his Ph.D. scholarship, as well as the Laboratory for Analysis and Architecture of Systems (LAAS-CNRS) in Toulouse, France, for the hospitality and openness to academic exchange.

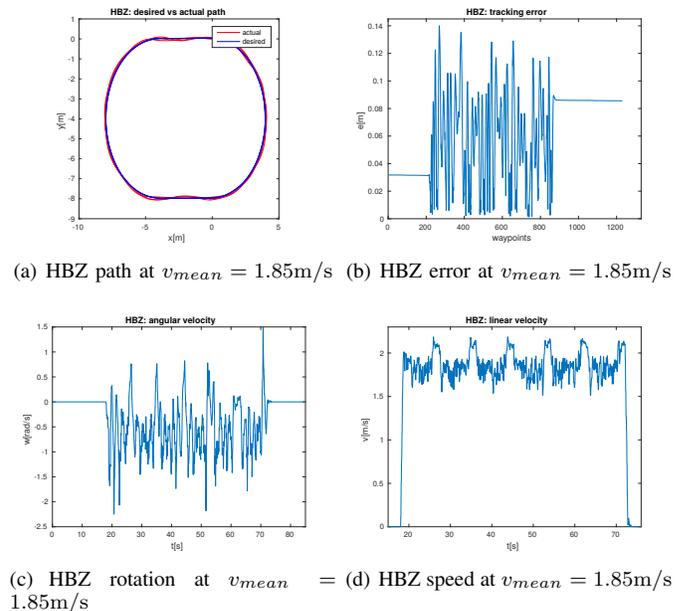


Fig. 25. Path following performance with Segway RMP 440 on an old asphalt mixed with gravel, following a 98.65m long path.

REFERENCES

- [Aguiar et al., 2004] Aguiar, A. P., Dačić, D. B., Hespanha, J. P., and Kokotović, P. (2004). Path-following or reference-tracking? *mn*, 1:1.
- [Belkhouche, 2009] Belkhouche, F. (2009). Reactive path planning in a dynamic environment. *IEEE Transactions on Robotics*, 25(4):902–911.
- [Boucher, 2016] Boucher, P. (2016). Waypoints guidance of differential-drive mobile robots with kinematic and precision constraints. *Robotica*, 34(4):876–899.
- [Buck et al., 2016] Buck, S., Hanten, R., Pech, C. R., and Zell, A. (2016). Synchronous dataflow and visual programming for prototyping robotic algorithms. In *Intelligent Autonomous Systems (IAS-14)*. Springer.
- [Caracciolo et al., 1999] Caracciolo, L., De Luca, A., and Iannitti, S. (1999). Trajectory tracking control of a four-wheel differentially driven mobile robot. In *Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference on*, volume 4, pages 2632–2638. IEEE.
- [De Luca et al., 1998] De Luca, A., Oriolo, G., and Samson, C. (1998). Feedback control of a nonholonomic car-like robot. In *Robot motion planning and control*, pages 171–253. Springer.
- [Deremetz et al., 2017] Deremetz, M., Lenain, R., Thuilot, B., and Rousseau, V. (2017). Adaptive trajectory control of off-road mobile robots: A multi-model observer approach. In *Robotics and Automation (ICRA), 2017 IEEE International Conference on*, pages 4407–4413. IEEE.
- [Elbanhawi et al., 2016] Elbanhawi, M., Simic, M., and Jazar, R. (2016). Receding horizon lateral vehicle control for pure pursuit path tracking. *Journal of Vibration and Control*, page 1077546316646906.
- [Huskić et al., 2017a] Huskić, G., Buck, S., Ibragüen González, L. A., and Zell, A. (2017a). Outdoor person following at higher speeds using a skid-steered mobile robot. In *Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on*. IEEE.
- [Huskić et al., 2017b] Huskić, G., Buck, S., and Zell, A. (2017b). Gerona: Generic robot navigation; a modular framework for robot navigation and control. *Journal of Intelligent & Robotic Systems*. (submitted).
- [Huskić et al., 2017c] Huskić, G., Buck, S., and Zell, A. (2017c). Path following control of skid-steered wheeled mobile robots at higher speeds on different terrain types. In *Robotics and Automation (ICRA), 2017 IEEE International Conference on*, pages 3734–3739. IEEE.
- [Iagnemma and Dubowsky, 2004] Iagnemma, K. and Dubowsky, S. (2004). *Mobile robots in rough terrain: Estimation, motion planning, and control with application to planetary rovers*, volume 12. Springer Science & Business Media.
- [Indiveri et al., 2007] Indiveri, G., Nüchter, A., and Lingemann, K. (2007). High speed differential drive mobile robot path following control with

- bounded wheel speed commands. In *Robotics and Automation, 2007 IEEE International Conference on*, pages 2202–2207. IEEE.
- [Kanayama et al., 1990] Kanayama, Y., Kimura, Y., Miyazaki, F., and Noguchi, T. (1990). A stable tracking control method for an autonomous mobile robot. In *Robotics and Automation, 1990. Proceedings., 1990 IEEE International Conference on*, pages 384–389. IEEE.
- [Khalil and Grizzle, 1996] Khalil, H. K. and Grizzle, J. (1996). *Nonlinear systems*, volume 3. Prentice hall New Jersey.
- [Kim and Minor, 2007] Kim, Y. and Minor, M. A. (2007). Path manifold-based kinematic control of wheeled mobile robots considering physical constraints. *The International Journal of Robotics Research*, 26(9):955–975.
- [Kozłowski and Pazderski, 2004] Kozłowski, K. and Pazderski, D. (2004). Modeling and control of a 4-wheel skid-steering mobile robot. *Int. J. Appl. Math. Comput. Sci.*, 14(4):477–496.
- [Kozłowski and Pazderski, 2006] Kozłowski, K. and Pazderski, D. (2006). Practical stabilization of a skid-steering mobile robot—a kinematic-based approach. In *Proc. IEEE 3rd Int. Conf. on Mechatronics*, pages 519–524.
- [Kronfeld et al., 2010] Kronfeld, M., Planatscher, H., and Zell, A. (2010). The eva2 optimization framework. In *International Conference on Learning and Intelligent Optimization*, pages 247–250. Springer.
- [Lenain et al., 2010] Lenain, R., Lucet, E., Grand, C., Thuilot, B., and Amar, F. B. (2010). Accurate and stable mobile robot path tracking: An integrated solution for off-road and high speed context. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 196–201. IEEE.
- [Lenain et al., 2011] Lenain, R., Thuilot, B., Hach, O., and Martinet, P. (2011). High-speed mobile robot control in off-road conditions: a multi-model based adaptive approach. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 6143–6149. IEEE.
- [Lucet et al., 2009] Lucet, E., Grand, C., Sallé, D., and Bidaud, P. (2009). Dynamic velocity and yaw-rate control of the 6wd skid-steering mobile robot roburoc6 using sliding mode technique. In *IEEE/RSJ International Conference on Intelligent Robots and Systems, St. Louis, Missouri, USA*.
- [Mandow et al., 2007] Mandow, A., Martínez, J. L., Morales, J., Blanco, J. L., García-Cerezo, A., and Gonzalez, J. (2007). Experimental kinematics for wheeled skid-steer mobile robots. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pages 1222–1227. IEEE.
- [Martínez et al., 2005] Martínez, J. L., Mandow, A., Morales, J., Pedraza, S., and García-Cerezo, A. (2005). Approximating kinematics for tracked mobile robots. *The International Journal of Robotics Research*, 24(10):867–878.
- [Martins et al., 2017] Martins, F. N., Sarcinelli-Filho, M., and Carelli, R. (2017). A velocity-based dynamic model and its properties for differential drive mobile robots. *Journal of Intelligent & Robotic Systems*, 85(2):277–292.
- [Micaelli and Samson, 1993] Micaelli, A. and Samson, C. (1993). Trajectory tracking for unicycle-type and two-steering-wheels mobile robots. In *IFAC Proceedings Volumes*, volume 27.
- [Miller and Murphey, 2012] Miller, L. M. and Murphey, T. D. (2012). Simultaneous optimal parameter and mode transition time estimation. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 719–724. IEEE.
- [Ostafew et al., 2016] Ostafew, C. J., Schoellig, A. P., and Barfoot, T. D. (2016). Robust constrained learning-based nmpc enabling reliable mobile robot path tracking. *The International Journal of Robotics Research*, 35(13):1547–1563.
- [Pentzer et al., 2014a] Pentzer, J., Brennan, S., and Reichard, K. (2014a). Model-based prediction of skid-steer robot kinematics using online estimation of track instantaneous centers of rotation. *Journal of Field Robotics*, 31(3):455–476.
- [Pentzer et al., 2014b] Pentzer, J., Brennan, S., and Reichard, K. (2014b). The use of unicycle robot control strategies for skid-steer robots through the icr kinematic mapping. In *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, pages 3201–3206. IEEE.
- [Quigley et al., 2009] Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., Wheeler, R., and Ng, A. Y. (2009). Ros: an open-source robot operating system. In *ICRA workshop on open source software*.
- [Rajagopalan et al., 2016] Rajagopalan, V., Meriçli, Ç., and Kelly, A. (2016). Slip-aware model predictive optimal control for path following. In *Robotics and Automation (ICRA), 2016 IEEE International Conference on*, pages 4585–4590. IEEE.
- [Samson and Ait-Abderrahim, 1991] Samson, C. and Ait-Abderrahim, K. (1991). Mobile robot control part 1: feedback control of a non-holonomic mobile robots. *INRIA report*, 1281.
- [Shigley and Uicker, 1991] Shigley, J. and Uicker, J. (1991). *Theory of machines and mechanisms*.
- [Shimoda et al., 2007] Shimoda, S., Kuroda, Y., and Iagnemma, K. (2007). High-speed navigation of unmanned ground vehicles on uneven terrain using potential fields. *Robotica*, 25(4):409–424.
- [Slotine et al., 1991] Slotine, J.-J. E., Li, W., et al. (1991). *Applied nonlinear control*, volume 199. Prentice hall Englewood Cliffs, NJ.
- [Soetanto et al., 2003] Soetanto, D., Lapierre, L., and Pascoal, A. (2003). Adaptive, non-singular path-following control of dynamic wheeled robots. In *Decision and Control, 2003. Proceedings. 42nd IEEE Conference on*, volume 2, pages 1765–1770. IEEE.
- [Spenko et al., 2004] Spenko, M., Iagnemma, K., and Dubowsky, S. (2004). High speed hazard avoidance for mobile robots in rough terrain. In *Proceedings of the SPIE Conference on Unmanned Ground Vehicles*, pages 439–450.
- [Williams et al., 2016] Williams, G., Drews, P., Goldfain, B., Rehg, J. M., and Theodorou, E. A. (2016). Aggressive driving with model predictive path integral control. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1433–1440. IEEE.
- [Williams et al., 2017] Williams, G., Wagener, N., Goldfain, B., Drews, P., Rehg, J. M., Boots, B., and Theodorou, E. A. (2017). Information theoretic mpc for model-based reinforcement learning. In *International Conference on Robotics and Automation (ICRA)*.
- [Wong, 2001] Wong, J. Y. (2001). *Theory of ground vehicles*. John Wiley & Sons.
- [Xin and Minor, 2012] Xin, M. and Minor, M. (2012). Backstepping vehicle steering controller using integral and robust control based on dynamic state estimation. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 3132–3137. IEEE.
- [Yi et al., 2007] Yi, J., Song, D., Zhang, J., and Goodwin, Z. (2007). Adaptive trajectory tracking control of skid-steered mobile robots. In *Robotics and Automation, 2007 IEEE International Conference on*, pages 2605–2610. IEEE.
- [Yi et al., 2009] Yi, J., Wang, H., Zhang, J., Song, D., Jayasuriya, S., and Liu, J. (2009). Kinematic modeling and analysis of skid-steered mobile robots with applications to low-cost inertial-measurement-unit-based motion estimation. *Robotics, IEEE Transactions on*, 25(5):1087–1097.
- [Yu et al., 2010] Yu, W., Chuy Jr, O. Y., Collins Jr, E. G., and Hollis, P. (2010). Analysis and experimental verification for dynamic modeling of a skid-steered wheeled vehicle. *IEEE transactions on robotics*, 26(2):340–353.