Teaching a Vehicle to Autonomously Drift: A Data-based Approach Using Neural Networks

Acosta, M. & Kanarachos, S.

Teaching a Vehicle to Autonomously Drift: A Data-based Approach Using Neural Networks

Manuel Acosta[1], Stratis Kanarachos

Department of Mechanical, Aerospace and Automotive Engineering,Coventry University, Coventry, United Kingdom

Abstract

This paper presents a novel approach to teach a vehicle how to drift, in a similar manner that professional drivers do. Specifically, a hybrid structure formed by a Model Predictive Controller and feedforward Neural Networks is employed for this purpose. The novelty of this work lies in a) the adoption of a data-based approach to achieve autonomous drifting along a wide range of road radii and body slip angles, and b) in the implementation of a road terrain classifier to adjust the system actuation depending on the current friction characteristics. The presented drift control system is implemented in a multi-actuated ground vehicle equipped with active front steering and in-wheel electric motors and trained to drift by a real test driver using a driver-in-the-loop setup. Its performance is verified in the simulation environment IPG-CarMaker through different open loop and path following drifting manoeuvres.

Keywords: Neural Networks, Autonomous Drift control, Autonomous Vehicles, Multi-Actuated Ground Vehicles, Model Predictive Control

1. Introduction

Future autonomous vehicles will be required to operate safely at the limits of handling under all environmental and roadway conditions (Level 5 autonomy, [1]). Among these conditions, automated driving on limited ma-
5 noeuvrability surfaces such as deep snow or gravel is still challenging and unexplored. The friction characteristics exhibited by the tyres on these surfaces

---

[1]ac3354@coventry.ac.uk

and, in particular, the abrupt reduction of the tyre cornering stiffness [2, 3] affect drastically the vehicle stability and controllability. In these conditions, traditional Yaw Stability Control (YSC) systems based on the controllability concept presented in [4] might perform inefficiently, limiting the cornering capabilities [5, 6] and overall vehicle manoeuvrability.

Recently, vehicle dynamics researchers have shifted their attention to the rallying motorsport discipline. Specifically, it is stated that understanding how professional rally drivers exploit the full chassis potential on loose surfaces [7, 2] could help to elucidate the path towards the development of highly-skilled autonomous cars. To this point, relevant research works on expert driving skills such as active drifting [8, 9, 10, 11, 12, 13, 14, 15] or vehicle agile manoeuvring [16, 17, 18, 19, 20, 21] have been introduced. Regarding research on active drifting, Velenis et al. [10] proposed a Linear Quadratic Regulator (LQR) to achieve drift control in a rear-wheel-drive (RWD) vehicle configuration. Moreover, the authors pointed out through numerical analysis the necessity of applying active drifting techniques to maximise the vehicle lateral dynamics on loose surfaces. A similar drift controller was proposed in [15] employing onboard vision-based and inertial measurements. Finally, Acosta et al. proposed an Autonomous Drift Control (ADC) system to achieve simultaneously drift control and path following employing a cost-effective measurement setup [5, 22].

The major drawback derived from the previous works is that an accurate tyre model is required to calculate the controller references (drift equilibrium solutions) [9]. Furthermore, several tyre parameters have to be known in advance to design the controller (e.g. local cornering and longitudinal tyre stiffness at the target drift operating point [5, 22]). This methodology is not recommended from a mass production perspective, as the parameterization of a tyre model in off-road conditions can be extremely complex and costly. To start with, new dedicated facilities may be required to operate safely with a skid trailer [23] on these surfaces. In addition, state-of-the-art skid trailers have a limited lateral slip characterisation range (e.g. typically between ±15 or ± 20 degrees [23]). As lateral slip values can be as high as 40 degrees during drift stabilisation, a wider tyre characterisation range would be necessary to compute the drift equilibrium solutions employing a numerical tyre model-based approach.

These limitations have motivated the exploration of alternative strategies avoiding the use of a tyre model. In this line, Cutler and How [14] solved the autonomous drifting problem using Reinforcement Learning (RL). The

authors proposed a methodology where the control policy was approximated as a closed-loop controller using a Radial Basis Function (RBF) network and a Probabilistic Inference for Learning COntrol (PILCO) algorithm was employed to refine the initial policy and achieve the drifting motion. The ability of this approach to generalise the drifting behaviour to a wider range of reference body slip angles or along arbitrary road geometries was not explored. As indicated in [24], RL problems are often initialised using example demonstrations or hand-craft policies. The necessity of generating such initial policies and repeating the "learning" process for each envisaged task might complicate the development of a control system with a large operating envelope. Latest advances in RL techniques can be found in the following references [25, 26, 27, 28]

The main motivation of this work is to develop a novel tyre-model-less drifting system able to perform drifting manoeuvres autonomously in a wide range of scenarios (arbitrary road geometries and different road terrains). According to the authors' literature survey, it is the first time that a driverless drifting system is designed for such a large operating envelope. This is accomplished using a data-based approach. In brief, the drift stabilisation and path following tasks are achieved by means of a hierarchical structure formed by an upper-level proportional-integral-derivative (PID) controller and a lower-level Model Predictive Controller (MPC). Instead of using an analytical tyre model-based approach, the unknown drift references and tyre parameters required by the lower-level controller are provided by feedforward Neural Networks (NN). These are trained to learn the drifting operating points and tyre friction properties during real drifting manoeuvres. The drifting tests are executed by a test driver using an experimental driver-in-the-loop (DIL) setup. In addition, several NNs are trained in different road terrains and a data-based road friction classifier is designed to select the set of NNs suitable for the identified road terrain. This way, the proposed system is able to adjust its driving behaviour depending on the current road terrain, like professional drivers do.

The reason for using a hybrid structure instead of an end-to-end learning approach [29] is twofold. First, it is very difficult, if not impossible, to fully validate and verify the behaviour of an end-to-end learning system [30]. Thus, it might be not possible to meet ISO-26262 [31] requirements for mass marketing these solutions. The second reason is that it does not make sense to rely only on "black box" models and omit the knowledge and insight gained by modelling. Instead, it is proposed to use models where they can

3

describe sufficiently well the system behaviour and NN where it is difficult or meaningless to derive a model. For the latter, we refer to cases that require complex models described by a huge number of parameters, which might be also uncertain or difficult to measure (e.g. tyre model in extreme off-road conditions). The results presented in this paper demonstrate that it could be possible to develop a highly-skilled autonomous drifting vehicle from a catalogue of standardised drifting manoeuvres. The rest of the paper is organised as follows:

In Section 2 the vehicle model used in the proposed system is derived and a brief description of the actuators employed in the Multi-Actuated Ground Vehicle (MAGV) is provided. In Section 3 the intelligent ADC system is described in detail. The section is completed with a brief insight into MPC and feedforward NN. After that, the methodology followed to train the intelligent ADC system is detailed in Section 4. The performance of the proposed system is assessed in Section 5 through several simulations performed in the high-fidelity vehicle dynamics software IPG-CarMaker. Finally, conclusions and future research steps are presented in Section 6.
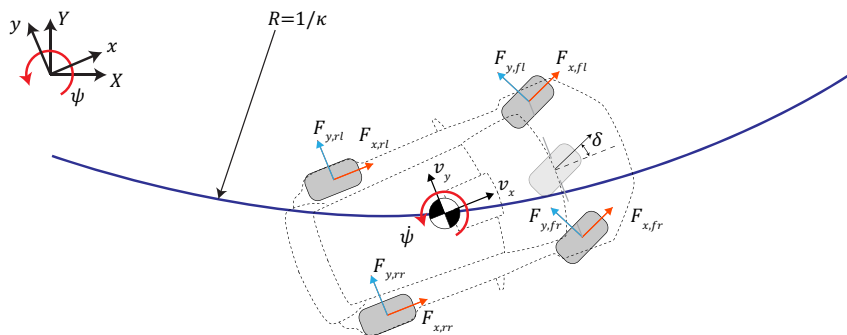
2. Vehicle Modelling

Figure 1: Two-track vehicle planar dynamics model.

In this paper, a two-track vehicle planar dynamics model is employed to approximate the vehicle responses, Figure 1. Following recommendations from previous works on drift stabilisation [10], the roll and pitch dynamics are disregarded and assumed to have little influence on the steady-state drifting

4

behaviour to reduce the model complexity. If rotating balance and force equilibrium equations are taken around the centre of gravity, the expressions (1-3) are obtained,

$$m(\dot{v}_x - v_y\dot{\psi}) = F_{x,fl}\cos\delta + F_{x,fr}\cos\delta - F_{y,fl}\sin\delta \\ - F_{y,fr}\sin\delta + F_{x,rl} + F_{x,rr} \tag{1}$$

$$m(v_x\dot{\psi} + \dot{v}_y) = F_{y,fl}\cos\delta + F_{y,fr}\cos\delta + F_{x,fl}\sin\delta + F_{x,fr}\sin\delta \\ + F_{y,rl} + F_{y,rr} \tag{2}$$

$$I_\psi\ddot{\psi} = (F_{y,fl}\cos\delta + F_{y,fr}\cos\delta + F_{x,fl}\sin\delta \\ + F_{x,fr}\sin\delta)l_f - (F_{y,rl} + F_{y,rr})l_r \\ + \frac{t_{wf}}{2}(F_{x,fr}\cos\delta - F_{x,fl}\cos\delta \\ + F_{y,fl}\sin\delta - F_{y,fr}\sin\delta) + \frac{t_{wr}}{2}(F_{x,rr} - F_{x,rl}) \tag{3}$$

where the vehicle planar velocities are denoted by v_x, v_y and the yaw rate by $\dot{\psi}$. The vehicle mass is designated by m, the yaw inertia by I_ψ, the distances from the front and rear axles to the centre of gravity by l_f, l_r, and the front and rear track widths by t_{wf}, t_{wr} respectively. δ is the angle steered by the front wheels, and $F_{y,i}, F_{x,i}$ are the tyre lateral and longitudinal forces (with $i \in \{front-left, front-right, rear-left, rear-right\}$). The latter forces are often computed using a nonlinear function $f(\alpha, \lambda)$, which depends on the tyre lateral slips (α), tyre longitudinal slips (λ), and additional variables such as the wheel inclination angle (e.g. *Magic Formula* Tyre model [32]).

2.1. Tyre forces

In order to facilitate the adoption of a linear controller, a linearised tyre force formulation was employed in this work. If a first order Taylor series expansion is performed on the nonlinear expression $f(\alpha, \lambda)$ and cross-stiffness terms are neglected [33, 5], the linearised forces (4-5) can be obtained.

$$F_{y,i} \approx F_{yss,i} + C_{\alpha,i}\Delta\alpha_i \tag{4}$$
$$F_{x,i} \approx F_{xss,i} + C_{\lambda,i}\Delta\lambda_i \tag{5}$$

Where C_α and C_λ are the cornering and longitudinal tyre stiffnesses at each equilibrium point. F_{yss}, F_{xss} are the steady-state tyre equilibrium forces,

and $\Delta\alpha, \Delta\lambda$ are perturbations around these equilibrium points. As mentioned in the introduction, the former tyre parameters (C_α, C_λ) are directly approximated by NN structures. Additional details are provided in Section 3. The steady-state equilibrium tyre forces are eliminated when the regulator problem is formulated (i.e. $\Delta F_{y,i} = F_{y,i} - F_{yss,i} \approx C_{\alpha,i}\Delta\alpha_i$). The longitudinal slips (λ) are defined in this work following the ISO slip convention [32, 34], expression (6),

$$\lambda_i = \frac{\omega_i r_e - V_{xc,i}}{V_{xc,i}} \tag{6}$$

with ω being the wheel angular velocity, r_e the wheel effective radius, and V_{xc} the longitudinal velocity at the wheel centre. The angular velocity can be easily obtained from the wheel rotating dynamics, expression (7).

$$I_\omega \dot{\omega}_i = T_i - F_{x,i} r_e \tag{7}$$

In this case, I_ω is the wheel inertia and T_i is the torque provided by the electric motors. Regarding the longitudinal velocities at the wheel centres (V_{xc}), these terms can be obtained from the vehicle planar motion states as follows,

$$V_{xc,fl} = \frac{V\cos\beta - \frac{t_{wf}}{2}\dot{\psi}}{(\cos\delta + \tan\alpha_{fl}\sin\delta)}, V_{xc,fr} = \frac{V\cos\beta + \frac{t_{wf}}{2}\dot{\psi}}{(\cos\delta + \tan\alpha_{fr}\sin\delta)} \tag{8}$$

$$V_{xc,rl} = V\cos\beta - \frac{t_{wr}}{2}\dot{\psi}, \qquad V_{xc,rr} = V\cos\beta + \frac{t_{wr}}{2}\dot{\psi} \tag{9}$$

where the velocity module is $V = \sqrt{v_x^2 + v_y^2}$ and the body slip angle $\beta = \arctan\left(\frac{v_y}{v_x}\right)$. Finally, the tyre lateral slips α_i are obtained from the following expressions,

$$\alpha_{fl} = \delta - \arctan\left(\frac{\dot{\psi}l_f + V\sin\beta}{V\cos\beta - \frac{t_{wf}}{2}\dot{\psi}}\right), \alpha_{fr} = \delta - \arctan\left(\frac{\dot{\psi}l_f + V\sin\beta}{V\cos\beta + \frac{t_{wf}}{2}\dot{\psi}}\right) \tag{10}$$

$$\alpha_{rl} = \arctan\left(\frac{\dot{\psi}l_r - V\sin\beta}{V\cos\beta - \frac{t_{wr}}{2}\dot{\psi}}\right), \qquad \alpha_{rr} = \arctan\left(\frac{\dot{\psi}l_r - V\sin\beta}{V\cos\beta + \frac{t_{wr}}{2}\dot{\psi}}\right) \tag{11}$$

6

2.2. Chassis actuators

The proposed MAGV incorporates Active Front Steering (AFS) and in-wheel electric motors (EM). Actuator limits were considered in these chassis systems. Specifically, the maximum torque provided by the electric motors was determined from the torque and wheel speed values presented in Table 1. These were extracted from field tests executed on state-of-the-art electric motors [35].

Table 1: Torque versus wheel speed values of the in-wheel EM. Values approximated from the experimental results presented in [35].

$\omega(rpm)$	0	500	750	1000	1250	1500	1750
$T(Nm)$	650	630	610	490	380	310	290

Regarding the AFS system, maximum steering amplitude and maximum slew rate constraints were considered. The steering dynamics were disregarded [10] under the assumption of a purely kinematic model [36]. In particular, the maximum angle steered by the front wheels was limited to $|\delta_{\max}| = 450$ deg $/SR$ and the maximum steering rate was limited to $|\dot{\delta}| = 1200$ deg/s $/ SR$, where SR is the steering ratio. The latter constraint was impossed based on the authors' experience with steering robots [37]. To conclude, the proposed vehicle modelisation can be synthesised by the following vectors of inputs ($\mathbf{U}$), states ($\mathbf{X}$), and parameters ($\mathbf{p}$):

$$\mathbf{U} = \{\delta, T_i\} \tag{12}$$

$$\mathbf{X} = \{v_x, v_y, \dot{\psi}, \omega_i\} \tag{13}$$

$$\mathbf{p} = \{m, I_\psi, l_f, l_r, t_{wf}, t_{wr}, r_e, I_\omega, C_{\lambda,i}, C_{\alpha,i}\} \tag{14}$$

3. Artificially Intelligent Drift Control

The structure of the intelligent drifting system proposed in this work is depicted schematically in Fig. 2. Two layers can be clearly differentiated in this structure: the high-level references and parameters layer, and the low-level reference tracking layer.

The upper-level structure generates the reference states (x_{ss}), equilibrium inputs (u_{ss}), and tyre parameters (C_{ss}) required to achieve the drift control

7

task. This is accomplished by means of the *AI Drift References and Parameters* block, which is composed of two sets of feedforward NNs trained at different road terrains. The outputs from these NNs depend on the target body slip angle β and target road curvature $\tilde{\kappa}$. The latter reference is obtained from a proportional-integral-derivative (PID) controller that corrects the open loop road curvature κ to minimise the lateral deviation error e_{lat}.
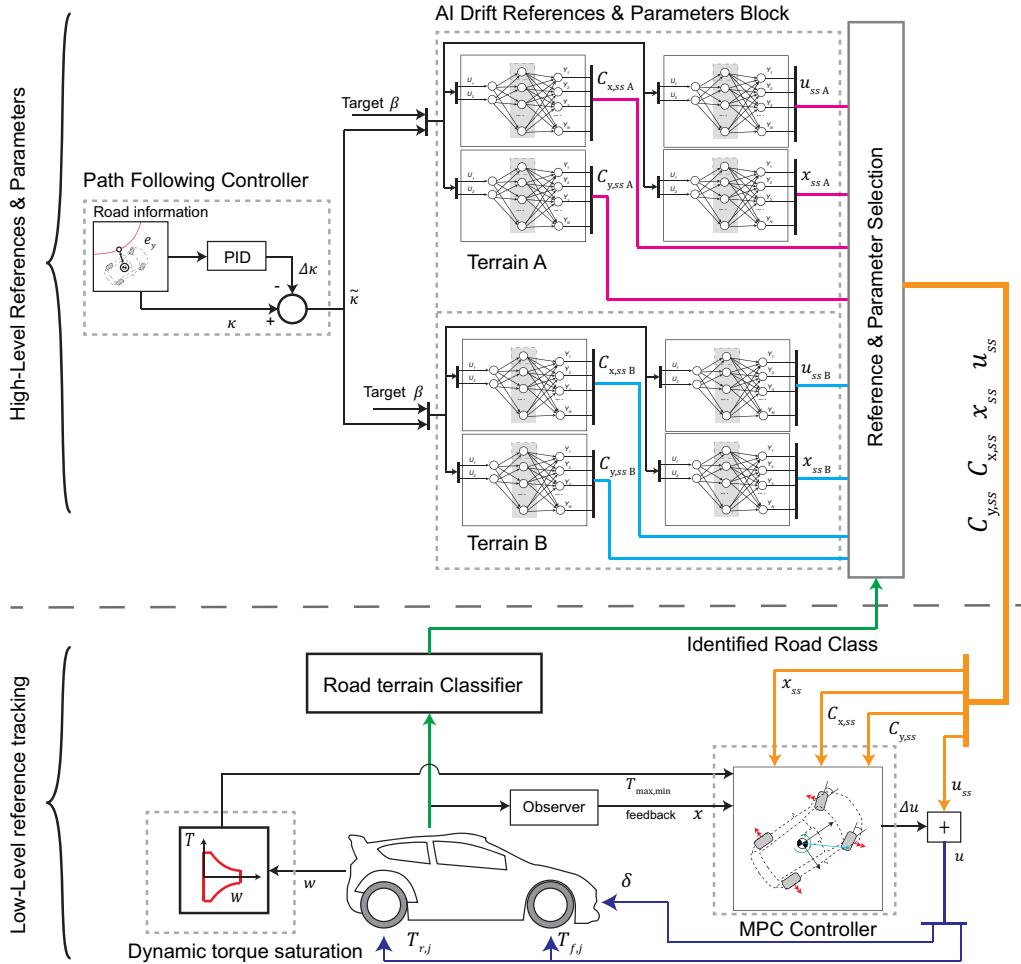

Figure 2: Scheme of the Artificially-Intelligent drift control system.

Therefore, the high-level layer provides the appropriate drift references

and tyre parameters for different combinations of target body slip angles and road geometries. These references and tyre parameters are adapted depending on the current road terrain, which is identified by an NN-based road terrain classifier. Once the suitable signals and parameters are selected by the road terrain classifier, these are used by the MPC controller situated in the low-level reference tracking layer. The MPC controller coordinates optimally the five constrained chassis actuators (AFS and four individual EM). For simplicity, full-state feedback of the tracked states $(v_x, v_y, \dot{\psi}, \omega_{fl}, \omega_{fr}, \omega_{rl}, \omega_{rr})$ is assumed in this work. As the lateral velocity cannot be measured cost-effectively and its estimation is not trivial [38, 39, 33], virtual sensing strategies are currently being investigated by the authors to eliminate the previous hypothesis using an observer block. The integration of these virtual sensing solutions [22] into the proposed hybrid controller will be presented in future works.

3.1. Why this system design?

According to the authors' literature survey, the drift control task has been studied only in a reduced number of works. Among these, analytical model-based controllers [10, 15, 8] have been used most. Other solutions based on RL [14] have been also explored. The proposed design introduces a new hybrid approach based on supervised machine learning.

With respect to analytical model-based controllers, these require a set of tyre parameters and steady-state drifting references to operate correctly. Thus, if the tyre information embedded in the controller and used to compute the previous drifting references is inadequate, the controller will exhibit a poor performance, and will not be able to reproduce accurately the drifting motion exhibited by real drivers. Moreover, if the friction characteristics change drastically and the controller does not adapt accordingly, its operating envelope will be very limited.

The major aim of the proposed system is to avoid the previous shortcomings adopting a data-based approach. In this case, the system internal parameters and drifting steady-state references are directly learned from a real driver using a supervised machine learning approach. Therefore, the proposed system tries to mimic the "real" drifting motion, instead of tracking a set of analytical references that might be computed with an uncertain tyre model and may not approximate accurately the desired drifting state. Furthermore, the proposed scheme incorporates a data-based terrain classifier which adapts the system actuation to the current surface characteristics, like

9

professional drivers do. According to the authors' survey, it is the first time that a drifting system exhibiting this friction-based adaptation is presented.

In what concerns RL, the existing solutions have been verified in a reduced number of open-loop (path following not incorporated) test cases. RL-based systems require interactions with the real environment during several iterations to learn a specific task (e.g. drifting with a specific body slip angle and a specific open-loop radius). If the task to be performed by the agent changes (e.g. drifting with a different body slip angle), a new policy needs to be learned, and the training process has to be repeated. Therefore, it is expected that training an agent to drift along a wide continuous range of road radii and body slip angles (even without considering different road terrains) will be a cumbersome and costly task. Instead, the proposed data-based system relies on the interpolation capabilities of feedforward NNs. These are trained with data extracted from a coarse grid of operating conditions (different combinations of body slip angles and road radii). Once trained, the NNs are able to interpolate accurately between these points. This way, the proposed design incorporates a large continuous operating envelope requiring a reduced number of experimental drifting manoeuvres.

3.2. Feedforward Neural Networks

Feedforward or "static" Neural Networks are a powerful tool to characterise time-independent relationships between the inputs and outputs of a system ($Y = f(U)$). Following the definition presented in [40], the simplest element of a Neural Network structure is the Neuron, Fig. 3.
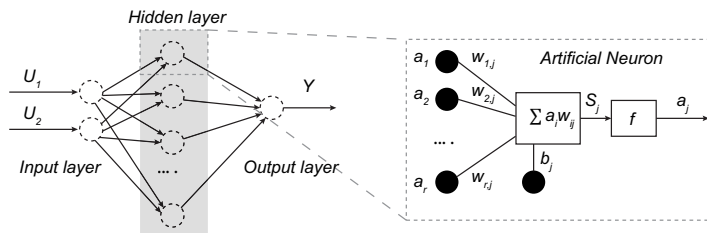

Figure 3: Scheme representing an Artificial NN structure and artificial neuron.

Neurons, or individual NN cells, are grouped into different layers, giving as a result a multi-layered structure formed by input, hidden, and output

10

layers. With such structure, any nonlinear function can be approximated by means of simple multiplications and summations,

$$S_j = \sum_i w_{ij} a_i + b_j \qquad (15)$$

$$a_j = f(S_j) \qquad (16)$$

where the output from the jth neuron is represented by S_j and is formed by the sum of the weighted outputs $(w_{ij}a_i)$ from the previous layer i. A constant bias b_j is added to the previous expression. Finally, f is the activation function of the jth layer and a_j represents the activation of the node at hand. Regarding the former, sigmoid activation functions are often chosen for the hidden layers whereas linear activation functions are selected for the output layers [41].

3.2.1. Neural Networks for Drift Control

As has been demonstrated in previous investigations [10, 5, 15, 22], the steady-state drift solutions corresponding to a given combination of body slip and road curvature (β, κ) can be entirely represented by a unique vector of vehicle feedforward inputs ($\mathbf{u_{ss}}$), vehicle equilibrium states ($\mathbf{x_{ss}}$), and vehicle parameters ($\mathbf{p_{ss}}$). Assuming that the vehicle mass and chassis geometric parameters remain unaltered during the system actuation, only the tyre parameters ($C_{\lambda,i}, C_{\alpha,i}$, expression (14)) depend on the pair (β, κ). Therefore, the dependence of the feedforward inputs, reference states, and tyre parameters on the given body slip and road curvature can be defined by the following nonlinear functions:

$$\mathbf{x_{ss}} = f_{x_{ss}}(\beta, \kappa) \qquad (17)$$

$$\mathbf{u_{ss}} = f_{u_{ss}}(\beta, \kappa) \qquad (18)$$

$$\mathbf{C_{\alpha,ss}} = f_{C_{\alpha,ss}}(\beta, \kappa) \qquad (19)$$

$$\mathbf{C_{\lambda,ss}} = f_{C_{\lambda,ss}}(\beta, \kappa) \qquad (20)$$

In this paper NNs are employed to "learn" these nonlinear expressions directly from real manoeuvres performed by a test driver, thus avoiding the necessity of performing numerical offline calculations with analytical tyre models. The proposed approach essentially overcomes the problem of deriving and identifying a tyre-road model (which can be particularly costly and

11

complex in off-road conditions [42, 3]) by following a data-based approach that can be directly implemented using a production vehicle.

As the training process is carried out on a given road terrain, the NNs are expected to produce inaccurate results if the drifting manoeuvres are performed on a different surface. A possible way to overcome this issue would be to extend the input vector of the previous expressions, incorporating additional parameters representative of the current road friction characteristics. Due to the large number of parameters required to approximate the friction characteristics of off-road surfaces, the complexity of the NN structure would increase drastically. Instead, authors propose a modular approach in which different sets of NNs are trained on each road terrain. The selection of the appropriate NNs is carried out by a road terrain classifier. Other modular approaches using NN have been employed by the authors in previous works on road friction monitoring [33].

3.3. Model Predictive Control

The role of the MPC controller is to stabilise the vehicle around the equilibrium states $\mathbf{x_{ss}}$ dictated by the upper-level drift reference structure (formed by the NNs defined previously). In other words, the MPC controller must provide the vector of input corrections $\mathbf{\Delta u}$ to drive the state error vector $\mathbf{\Delta x}$ to zero.

$$\mathbf{\Delta x} = \mathbf{x} - \mathbf{x_{ss}}, \quad \mathbf{\Delta u} = \mathbf{u} - \mathbf{u_{ss}} \tag{21}$$

Following the procedure described in [5], the vehicle dynamics expressions (1-11) were discretised using a first order approximation ($e^{AT_s} \approx 1 + AT_s$). The following system of linearised equations was obtained,

$$\mathbf{\Delta x}(k+1) = \mathbf{A_{ss}}\mathbf{\Delta x}(k) + \mathbf{B_{ss}}\mathbf{\Delta u}(k) \tag{22}$$

where the steady-state matrices $\mathbf{A_{ss}}$ and $\mathbf{B_{ss}}$ are particularised at each operating point (target body slip β and corrected road curvature $\tilde{\kappa}$). This is accomplished using the vector of equilibrium states $\mathbf{x_{ss}}$, feedforward inputs $\mathbf{u_{ss}}$, and tyre longitudinal and lateral stiffnesses ($\mathbf{C}_\alpha, \mathbf{C}_\lambda$) provided by the upper-level NN-based structure. The rate of change of the equilibrium points is assumed to be negligible in comparison to the system dynamics. The rest of variables that form the vector of chassis parameters ($m, I_\psi, l_f, l_r, t_{wf}, t_{wr}, r_e, I_\omega$)

12

are assumed to be known. Regarding the vehicle mass, yaw inertia, and centre of gravity location, different online parameter identification algorithms have been proposed in the literature [43]. Finally, full-state feedback is assumed for simplicity. Virtual sensing strategies using signals from inexpensive sensors have been already developed by the authors and promising results have been obtained [22]. The integration of these solutions into the system described in this work will be pursued in future stages of this research.

Model Predictive Control is chosen in this work to track the drift references due to its superior ability to handle optimally amplitude and slew rate actuator constraints. Specifically, in comparison to other linear-feedback-law controllers (e.g. Linear Quadratic Regulator), MPC calculates optimally nonlinear feedback laws through online numerical optimisation [44] when constrained systems are faced. Following the linear MPC derivation presented in [45], the evolution of the state-space model (22) during the future N_p steps under the sequence of future N_c control inputs $(\boldsymbol{\Delta}\mathbf{u}(k), ..., \boldsymbol{\Delta}\mathbf{u}(k + N_c - 1))$, can be expressed in compact form by the augmented system,

$$\boldsymbol{\Delta}\mathbf{X} = \mathbf{F_{ss}}\boldsymbol{\Delta}\mathbf{x}(k) + \boldsymbol{\Phi_{ss}}\boldsymbol{\Delta}\mathbf{U} \tag{23}$$

where the augmented state $\boldsymbol{\Delta}\mathbf{X}$ and input $\boldsymbol{\Delta}\mathbf{U}$ vectors are expressed by the equations (24-25).

$$\boldsymbol{\Delta}\mathbf{X} = [\boldsymbol{\Delta}\mathbf{x}(k + 1|k)^T, \boldsymbol{\Delta}\mathbf{x}(k + 2|k)^T, ..., \boldsymbol{\Delta}\mathbf{x}(k + N_p|k)^T]^T \tag{24}$$

$$\boldsymbol{\Delta}\mathbf{U} = [\boldsymbol{\Delta}\mathbf{u}(k)^T, \boldsymbol{\Delta}\mathbf{u}(k + 1)^T, ..., \boldsymbol{\Delta}\mathbf{u}(k + N_c - 1)^T]^T \tag{25}$$

The augmented matrices $\mathbf{F_{ss}}$ and $\boldsymbol{\Phi_{ss}}$ were extracted from [45] and are written below for clarity.

$$\mathbf{F_{ss}} = \begin{bmatrix} \mathbf{A_{ss}} \\ \mathbf{A_{ss}}^2 \\ \vdots \\ \mathbf{A_{ss}}^{N_p} \end{bmatrix}, \boldsymbol{\Phi_{ss}} = \begin{bmatrix} \mathbf{B_{ss}} & 0 & \cdots & 0 \\ \mathbf{A_{ss}B_{ss}} & \mathbf{B_{ss}} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{A_{ss}}^{N_p-1}\mathbf{B_{ss}} & \mathbf{A_{ss}}^{N_p-2}\mathbf{B_{ss}} & \cdots & \mathbf{A_{ss}}^{N_p-N_c}\mathbf{B_{ss}} \end{bmatrix} \tag{26}$$

The MPC control problem consists of calculating the N_c future control inputs that minimise the cost function $J(\boldsymbol{\Delta}\mathbf{U})$,

13

$$\begin{aligned} \underset{\mathbf{\Delta U}}{\text{minimize}} \quad & J(\mathbf{\Delta U}) \\ \text{subject to} \quad & \mathbf{A_c \Delta U} \leq \mathbf{b}(k) \end{aligned}$$

subjected to the linear amplitude and slew rate input constraints given by the matrices $\mathbf{A_c}$, $\mathbf{b}$. The derivation of these matrices is omitted here due to space limitations, additional details can be consulted in [45]. Finally, this optimisation problem can be solved online with reduced computational resources using Quadratic-Programming (QP) optimisation [44]. In order to employ the previous routine, it is necessary to rearrange the cost function $J(\mathbf{\Delta U})$ in a suitable form,

$$J(\mathbf{\Delta U}) = \mathbf{\Delta U}^T \mathbf{H} \mathbf{\Delta U} + 2\mathbf{\Delta x}(k)^T \mathbf{M}^T \mathbf{\Delta U} \tag{27}$$

$$\mathbf{H} = \mathbf{\Phi_{ss}}^T \mathbf{\hat{Q}} \mathbf{\Phi_{ss}} + \mathbf{\hat{R}} \tag{28}$$

$$\mathbf{M} = \mathbf{\Phi_{ss}}^T \mathbf{\hat{Q}} \mathbf{F_{ss}} \tag{29}$$

where $\mathbf{\hat{Q}}, \mathbf{\hat{R}}$ are weighting matrices used to fix the relative importance of the tracking error and input energy expenditure.

3.4. Path Following PID

In all the previous works on drift control consulted by the authors, the road geometry is not considered, and the attention is placed on stabilising the vehicle around the drift equilibrium states [10, 15, 14]. Following a similar approach to [5], the PID control law

$$\tilde{\kappa} = \kappa - \Delta\kappa = \kappa - \left(k_p e_{lat} + k_d \dot{e}_{lat} + k_i \int e_{lat}\right) \tag{30}$$

is proposed to achieve simultaneously the drift control and path following tasks. The concept behind the proposed control law is illustrated in Fig. 4, and can be summarised in the following manner: when the lateral deviation error is positive, the reference curvature is decreased to straighten the current vehicle trajectory; if the lateral deviation error is negative, the reference road curvature is increased to tighten the vehicle trajectory. The sign criteria for the lateral deviation error is adopted from the road model presented in [2]. If the PID gains are carefully chosen and the response of the lower-level MPC controller is assumed to be fast enough to track the changes in the corrected

14

road curvature $\tilde{\kappa}$, the proposed control law will drive the lateral deviation error to zero. Therefore, the vehicle will eventually converge to the reference path under the proposed PID law. As an analytical Lyapunov-based stability proof for the proposed system is not trivial, the closed loop stability under different initial errors is studied by means of a phase-space coverage analysis. These results are presented in Section 5.2.1.
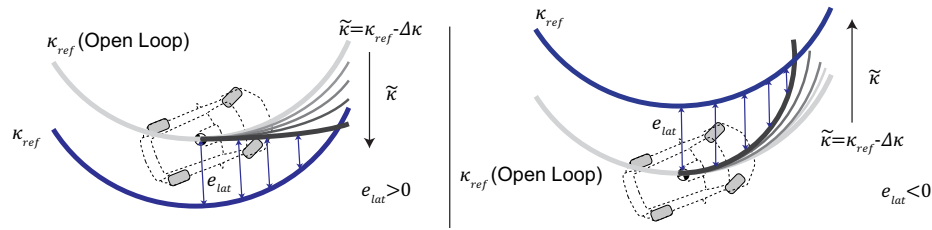

Figure 4: Scheme of the drift path-following PID actuation for a left-handed turn. (a) Positive lateral deviation error, (b) negative lateral deviation error.
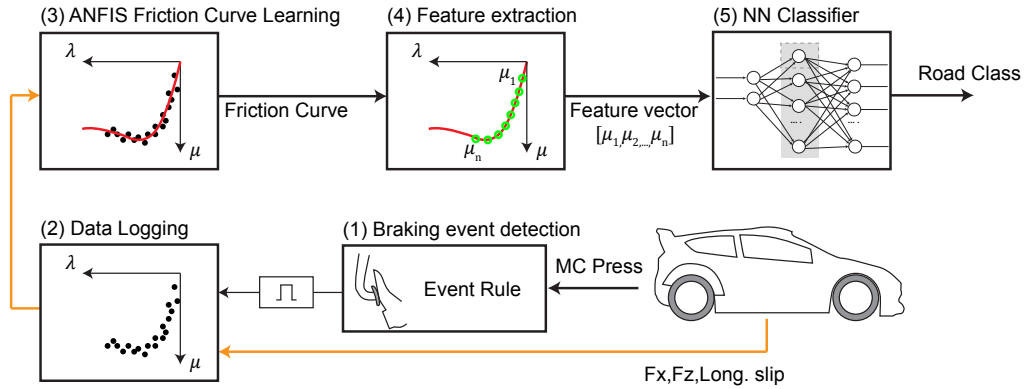
3.5. Road Terrain Classifier

Figure 5: Scheme of the road terrain classifier developed in this work. Once the road terrain is identified, the high-level drift references and parameters are adjusted accordingly.

The road terrain classifier proposed in this work is depicted schematically in Fig. 5. The operation of this structure can be summarised in the following manner. The first block (Braking event detection) monitors the master cylinder pressure signal and identifies whether a braking intervention is taking place. A rule-based approach is employed for this task. During the braking situation, the tyre vertical and longitudinal forces, as well as the tyre longitudinal slips are logged (block 2, Data Logging). Once the braking event has concluded, an Adaptive Neuro-Fuzzy Inference System (ANFIS) is trained to approximate the friction curve formed by the cloud of normalised longitudinal force (defined as $\mu = F_x/F_z$) versus slip data (block 3, ANFIS Friction Curve Learning). In order to acquire a cloud of points representative of the terrain friction characteristics, only braking interventions in which a certain longitudinal slip value (λ_{thres}) is developed (i.e. enough longitudinal excitation is present) are considered. The feature vector required by the terrain classifier is obtained directly from the trained ANFIS structure and consists of a set of uniformly-spaced friction values $\mu = \{\mu_1, \mu_2, ..., \mu_n\}$. This vector is passed through a Neural Network classifier (block 5) trained with braking data from different terrains to infer the road class that best matches the current friction characteristics.

ANFIS structures have been employed by the authors in previous works to "learn" the road friction characteristics during sinusoidal steering inputs [22]. In brief, the use of ANFIS as an intermediate step permits a straightforward extraction of a noise-free feature vector. Otherwise, e.g. computing the previous vector from raw data, would require the execution of non-trivial post-processing steps (such as data averaging in a predefined region) to reduce the influence of outlier points. Finally, the use of a data-based classifier is justified by the necessity of identifying drastically-different terrains (e.g. asphalt roads and loose surfaces). As has been demonstrated in previous works [22], the use of grip scaling-based methods to estimate the road friction potential taking as a reference an analytical tyre model (e.g. the Magic Formula [46]) can lead to significant errors when loose surfaces are considered.

3.5.1. Training of the road terrain classifier

The classifier was trained in IPG-CarMaker with data obtained from five different road surfaces, Fig. 6. These surfaces were implemented in the previous software employing an isotropic *Magic Formula* tyre model. The tyre parameters of the terrains 1 to 4 were obtained from Tavernini et al. [2],

16

and correspond to dry asphalt, wet asphalt, dirty off-road, and gravel-like surfaces respectively. A fifth terrain, low-mu loose surface, was added to the training dataset by decreasing the maximum friction factor ($D = 0.2$) of the fourth surface. For simplicity, the classifier was implemented in the front-left wheel of a sports-class vehicle. A two-wheel configuration will be employed in the future to detect non-uniform road friction characteristics (e.g. mu-split conditions).
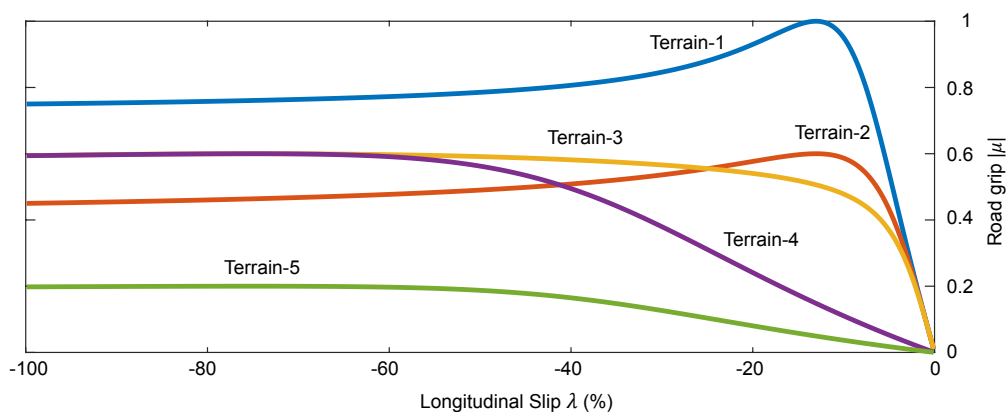

Figure 6: Road terrains implemented in IPG-CarMaker to train the classifier.

In total, 250 braking manoeuvres were executed with the static simulator presented in Fig. 8 (50 at each road surface). An additive white Gaussian noise model was employed to simulate noise levels representative of state-of-the-art automotive instrumentation, Table 5. Virtual Sensing techniques will be evaluated in the future to reconstruct the classifier inputs (F_x, F_z, λ) from inexpensive onboard measurements.

Table 2: Parameters employed in the road terrain classifier.

$\lambda_{thres}(\%)$	ANFIS MF	ANFIS $epochs$	NN $neurons$
-20	4	20	2

The parameters employed in the classifier are presented in Table 2. The longitudinal slip level required to consider a braking event as valid was set to -20%. The influence of the Anti-lock Braking System (ABS) actuation

17

on this threshold will be evaluated in the future. An ANFIS structure composed of 4 generalised bell-shaped Membership Functions (MF) was selected
395 after an iterative trial-and-error procedure. Overall, a reduced-size structure was pursued to limit the ANFIS training time (0.27s for a training dataset of 254 samples and 20 maximum epochs). The ANFIS was trained using a hybrid algorithm composed of least-squares and backpropagation gradient descent methods [47]. Additional details regarding ANFIS can be found in
400 [47]. The classifier feature vector is formed by the friction values associated with the longitudinal slip vector $\lambda = [-0.02 : -0.02 : -0.2]$. A pattern recognition shallow Neural Network was built in Matlab using the *pattern-net* function. A reduced two-neuron hidden-layer size was selected with the aim to avoid potential overfitting issues. The structure was trained using the
405 Scaled Conjugate Gradient algorithm and a 70/15/15 dataset division (training/validation/test). During the execution of this process, the classifier was able to identify correctly 231 samples out of the 250 samples that composed the training dataset. After that, several braking events were executed with the trained classifier and similar performance levels were obtained. An ex-
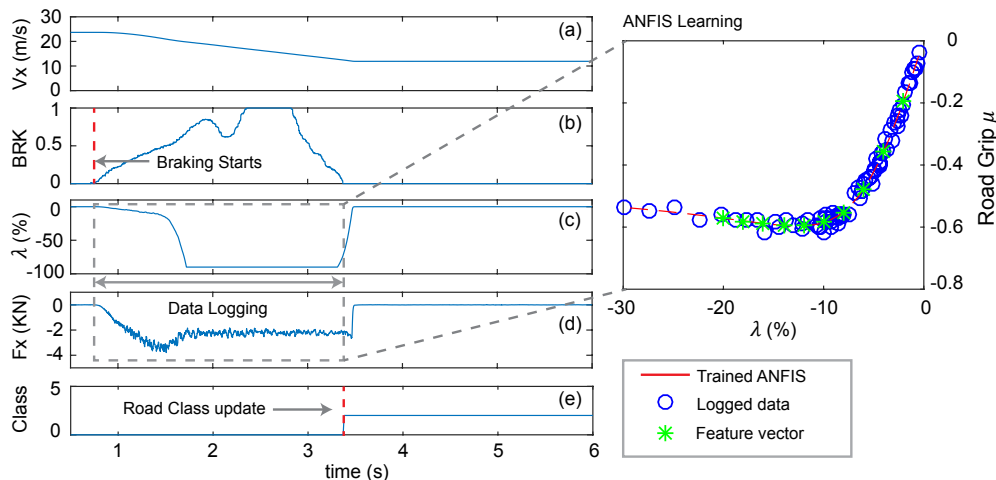410 ample braking test on a type 2 road terrain is depicted in Fig. 7.

Figure 7: Braking test on a type 2 road terrain. (a) Vehicle velocity, (b) Brake pedal position, (c) Longitudinal slip, (d) Longitudinal force, (e) identified road class.

During the first seconds ($t < 3.3$ s) the tyre forces and longitudinal slip are logged. Once the braking event has concluded, the ANFIS network is

18

trained and the classifier feature vector extracted. As the ANFIS structure is trained in a short time (0.27s), the proposed approach can be considered to run on "real-time". The experimental validation of the classifier with data from field tests will be pursued in the future. Authors believe that the proposed structure can significantly facilitate the recognition of the road terrain under challenging conditions (e.g. driving on loose surfaces), where grip scaling-based methods relying on an analytical tyre model are expected to fail [22].

4. Teaching a vehicle to drift

The strategy employed to train the intelligent drift control system is depicted schematically in Fig. 8.
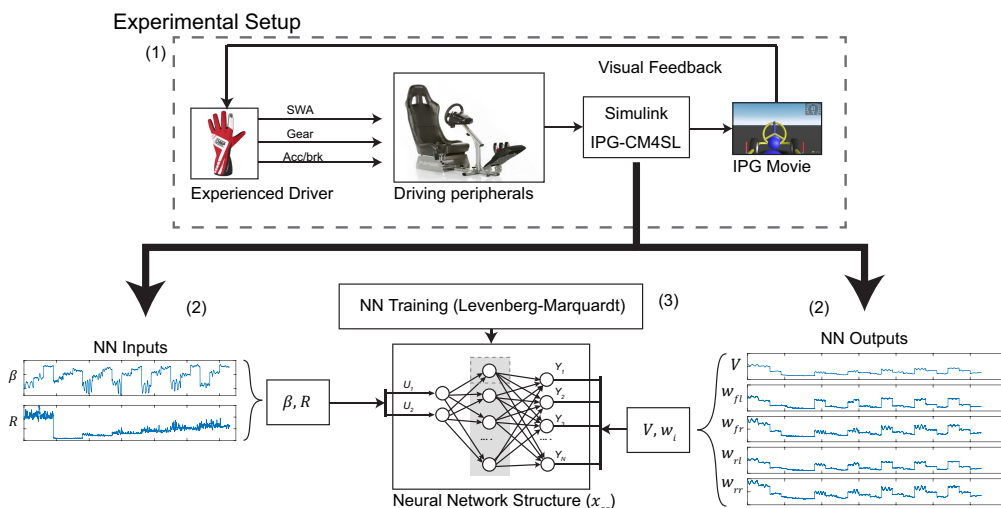
Figure 8: Scheme of the intelligent drift control training process. (1) Execution of drifting manoeuvres in IPG-CarMaker - (2) Preparation of the training datasets - (3) Training of the Neural Networks.

In brief, the training process was carried out in the following manner. Drifting manoeuvres on circular roads of different radii and covering different steady-state body slip angles were performed by a test engineer in a static DIL setup. The runs generated in IPG-CarMaker during the execution of these manoeuvres were logged using the *CarMaker for Simulink Library*

(CM4SL) and employed to construct the training datasets. These were post-
processed and used to train the NNs contained in the *AI drift references and parameters* block. The Levenberg-Marquardt optimisation routine was used for this purpose. This procedure was repeated in two different road terrains: class 4 (gravel) and class 5 (low-mu loose surface). A more detailed description regarding the latter steps is provided in Section 4.2.1.

4.1. High-fidelity simulation environment

A high-fidelity virtual environment was constructed in the vehicle dynamics simulation software *IPG-CarMaker* to train and evaluate the proposed intelligent drift control system. The static DIL setup introduced in Fig. 8 was employed to perform agile manoeuvres. A driving simulator was chosen to avoid compromising the driver's safety. The Logitech G-27 driving peripherals [48] were used to feed the driver's steering, gear, clutch, brake, and gas pedal inputs into IPG-CarMaker. In addition, road random profiles were used to generate a virtual rough road representative of off-road surfaces. The road random profiles were created with the Sayers model [49] and incorporated into the IPG virtual road by means of a .crg file. This file was constructed with the open source "opencrg" [50] software. For additional details [51] can be consulted.

4.1.1. Vehicle Model

A sports-class vehicle model from the IPG-CarMaker library was employed to train and verify the proposed intelligent drift control system. A full experimental validation of the proposed simulation model is out of the scope of this work due to the costs and complexity derived from the proposed MAGV architecture. The ability of IPG-CarMaker to provide realistic vehicle responses has been verified by the authors in previous works using a compact-class experimental vehicle [33]. A *custom powertrain* model was developed in IPG-CarMaker to incorporate the torque versus wheel speed curves presented in Section 2. For simplicity, uneven torque distributions were not considered in the virtual test vehicle. Thus, the electric motors provide a torque response proportional to the driver pedal position, given by the expression

$$T_{output,i} = T_{\max,i} P_{pos} \tag{31}$$

20

where $T_{\max,i}$ is the maximum torque that can be generated at the current wheel speed, and P_{pos} is the gas pedal position (ranging between 0 and 1). The tyre forces are modelled in the simulation environment using an Isotropic Magic Formula parameterisation [2], equations (32-33).

$$F_x = F_z \frac{\sigma_x}{\sigma} D \sin[C \arctan\{\sigma B - E(\sigma B - \arctan \sigma B)\}] \tag{32}$$

$$F_y = F_z \frac{\sigma_y}{\sigma} D \sin[C \arctan\{\sigma B - E(\sigma B - \arctan \sigma B)\}] \tag{33}$$

In this model, the theoretical slip quantities are obtained from the longitudinal and lateral wheel slips through the computation of the following expressions,

$$\sigma_x = \frac{\lambda}{1+\lambda}, \quad \sigma_y = \frac{\tan \alpha}{1+\lambda}, \quad \sigma = \sqrt{\sigma_x^2 + \sigma_y^2} \tag{34}$$

This formulation has been employed in previous works to evaluate the influence of the road terrain on the vehicle cornering performance [2] or to approach the drift stabilisation problem [10]. Finally, the chassis and tyre parameters used in the simulation model are presented in Table 3.

Table 3: Vehicle and tyre parameters used in the simulation model. The sub-indexes 4 and 5 denote the terrains 4 and 5 introduced in Section 3.5.

$l_f(m)$	$l_r(m)$	$I_\psi(Kgm^2)$	$m(kg)$
1.20	1.45	2325	1580
$r_e(m)$	$I_w(Kgm^2)$	$t_{wf}(m)$	$t_{wr}(m)$
0.32	2	1.5	1.5
B_4	C_4	D_4	E_4
1.5289	1.0901	0.6	-0.95084
B_5	C_5	D_5	E_5
1.5289	1.0901	0.2	-0.95084

4.2. Neural Networks Training

As mentioned previously, two sets of NNs were trained in this work: the first set in a gravel-like surface (terrain 4), and the second set in a low-mu loose surface (terrain 5). The graphs portrayed in this section were generated during the training procedure carried out on the class 4 terrain. The same

21

procedure was followed to train the second set of NNs on the class 5 terrain. The training datasets were generated by direct concatenation of the test runs executed in the virtual testing environment described previously, Fig. 9.
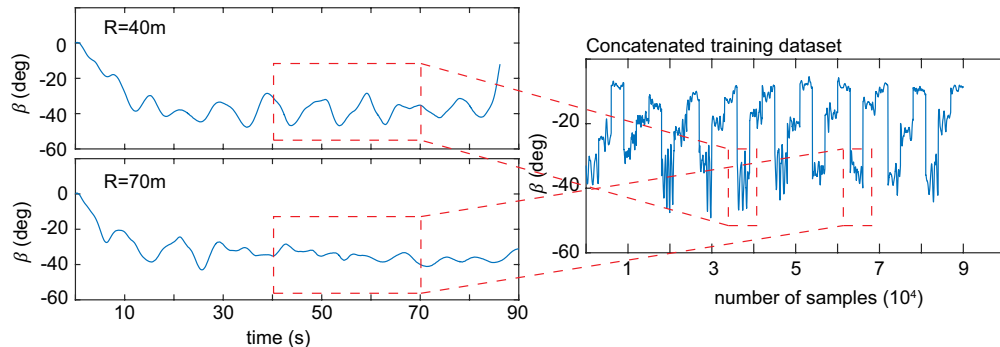

Figure 9: Direct concatenation of individual test runs once steady-state conditions have been achieved.

The drifting state ($|\beta| \approx$ 30-40 degrees) is achieved after some seconds and maintained during the rest of the test. In order to employ only data corresponding to steady-state conditions, the data contained in the time interval $t \in [40 \text{ s}, 70 \text{ s}]$ were extracted from each individual test run. Additional experiments were carried out to complete the test matrix presented in Table 4. Firstly, drifting manoeuvres were performed in circles of different radii, $R = [10 : 10 : 100]$. After that, in order to facilitate the generalisation of the proposed system to a wide range of reference body slip angles, these tests were repeated for two additional body slip angle ranges: Medium body slip $|\beta| \approx 15 - 25$ degrees and Low body slip $|\beta| \approx 10 - 15$ degrees. In total, 30 runs were generated and concatenated at each road terrain (terrains 4 and 5, Fig. 6).

Table 4: Test Matrix executed to form the training data sets. T-4: Terrain 4 / T-5: Terrain 5.

Body slip	$R = 10$	$R = 20$	$R = 30$	...	$R = 100$
High	T-4/T-5	T-4/T-5	T-4/T-5	...	T-4/T-5
Medium	T-4/T-5	T-4/T-5	T-4/T-5	...	T-4/T-5
Low	T-4/T-5	T-4/T-5	T-4/T-5	...	T-4/T-5

These runs were acquired at a sampling frequency of 100 Hz. Moreover,

22

in order to reproduce realistically the noise associated with state-of-the-art automotive testing equipment, an additive noise model of the form $\bar{y} = y + y_\sigma$ was employed. The standard deviations of these noise terms (y_σ) were extracted from sensor datasets [52, 53] and from field tests executed with Wheel Force Transducers (WFT). These values are given in Table 5. The standard deviation of the wheel speed noise was obtained from experimental onboard CAN measurements.

Table 5: Standard deviation error of the measured tyre force, inertial, and wheel speed signals.

σ_V [m/s]	σ_β [rad]	σ_ω [rad/s]
0.0278	0.0035	0.062
σ_{F_x} [N]	σ_{F_y} [N]	$\sigma_{\dot{\psi}}$[rad/s]
70	70	0.0019

The rest of signals required to train the NNs (e.g. tyre lateral slips) were computed from the noise-corrupted tyre force, wheel speed, and inertial simulation signals. Finally, noise-free torque signals were assumed to be provided by the electric motors. The acquisition and generation of the training datasets from cost-effective CAN measurements (avoiding the use of WFT) will be evaluated in the future using virtual sensing approaches [22].

4.2.1. Reference states and feedforward inputs

The nonlinear functions $f_{x_{ss}}$ and $f_{u_{ss}}$, defined in Section 3.2.1, were approximated by NNs trained following the scheme portrayed in Fig. 8. The vector of inputs (containing the road radius calculated as $R = \frac{V}{\dot{\psi}}$ and the body slip angle β) and the vectors of outputs were constructed by direct concatenation. The latter were formed by the steering wheel angle and individual wheel torques ($f_{u_{ss}}$) and the velocity module and individual wheel speeds ($f_{x_{ss}}$). The Levenberg-Marquardt algorithm was employed with a data partition of 70/15/15 (training/validation/test). The NNs were constructed in Matlab using the *feedforwardnet* function and trained with the *trainlm* routine. The training process was repeated several times varying the number of neurons of the hidden layer and good results were obtained with a six-neuron hidden-layer structure (2-6-5 NN). As acceptable results were already obtained using the preliminary training method and dataset partition, additional simulations were not performed.

23

The stability of the NNs was studied following the methodology detailed in [40] and adopted in previous works [33]. A total number of 50 NN structures were trained using different initial weights (randomly selected by the *trainlm* function), and the average output from these structures was taken as a reference. After that, the NN exhibiting the lowest Normalised Root Mean Square Error (NRMSE) with respect to the average value was chosen as the most accurate and representative structure of the real field data, and implemented in the final intelligent drift controller.

4.2.2. Cornering stiffness and longitudinal stiffness

The generation of the nonlinear functions $f_{C_\alpha,ss}$ and $f_{C_\lambda,ss}$ presented an added complexity, as it was not possible to obtain these directly from raw data. Due to the fact that the measured data present some inherent noise (from the rough road excitation and the measuring equipment), differentiating the tyre forces with respect to the tyre slips would lead to poor tyre stiffness estimates. Instead, a two-step approach is adopted in this work, Fig. 10. If the problem is particularised on the lateral dynamics case (function $f_{C_\alpha,ss}$), the following procedure is followed: First, a two-neuron hidden-layer NN (1-2-1) is trained to fit a cloud of $F_y - \alpha$ values. These are extracted from the steady-state interval of each test run ($t \in [40 \text{ s}, 70 \text{ s}]$). The average tyre lateral slip ($\bar{\alpha}$) is calculated on this time interval and the cornering stiffness (C_α) for the given steady-state drift equilibrium is obtained employing a finite differences approach [33],

$$C_\alpha \approx \frac{F_{y,up} - F_{y,down}}{\Delta\alpha_{up} + \Delta\alpha_{down}} \tag{35}$$

where the forces $F_{y,up}$ and $F_{y,down}$ are the outputs from the 1-2-1 NN evaluated on the lateral slip limits $\bar{\alpha} + \Delta\alpha_{up}$ and $\bar{\alpha} - \Delta\alpha_{down}$. The increments $\Delta\alpha_{up}$ and $\Delta\alpha_{down}$ were obtained from expressions

$$\Delta\alpha_{up} = \min(0.4(\max(\alpha) - \min(\alpha)), \max(\alpha) - \bar{\alpha}) \tag{36}$$
$$\Delta\alpha_{down} = \min(0.4(\max(\alpha) - \min(\alpha)), -\min(\alpha) + \bar{\alpha}) \tag{37}$$

These expressions guarantee that the lateral slip limits will remain within the interval defined by the experimental slip data used to train the 1-2-1 NN. The factor 0.4 was determined empirically with the aim to capture the small

24

oscillations derived from the continuous steering and throttle corrections required to stabilise the vehicle around the drift steady-state equilibrium. These small oscillations around the steady-state slip equilibrium facilitate the extraction of the correct tyre stiffness and avoid misleading results derived from linearisations in too short slip intervals. After that, the average values of the body slip $\bar{\beta}$ and radius $\bar{R}$ are associated to each cornering stiffness estimate $C_{\alpha,i}$. This process is repeated with each test run until the complete cloud of points (R, β, C_α) is formed, Figure 10-(1).
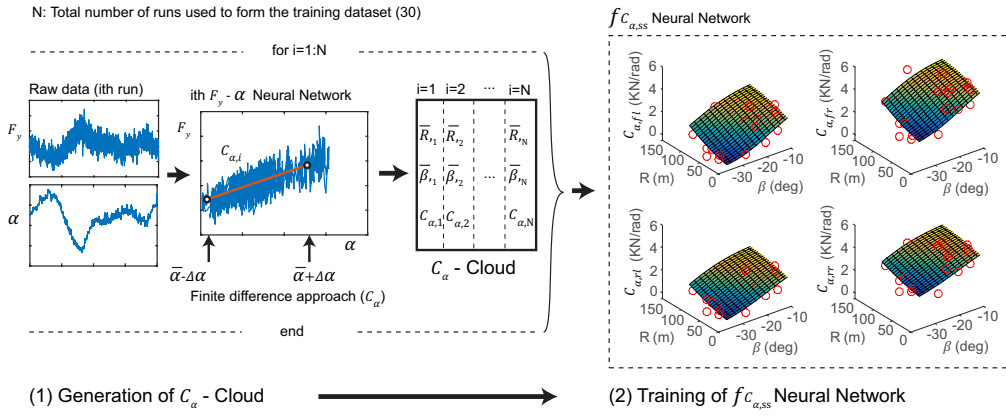

Figure 10: Approximation of the $f_{C_{\alpha,ss}}$ function. (1) A 1-2-1 NN structure is trained to approximate the pair $(F_y - \alpha)$ at each test run and the matrix of C_α points is generated. (2) A second NN structure is trained to approximate the cloud of points generated in step 1.

In the second step, a new NN structure is trained to fit the cloud of tyre stiffness points calculated previously. The same two-step procedure was followed to compute the nonlinear function $f_{C_\lambda,ss}$, this time using the tyre longitudinal slips λ and tyre longitudinal forces F_x. Regarding the selection of the NN structure used in the second step, NNs with distinct hidden-layer sizes (with the number of hidden neurons ranging from 2 to 8) were trained using the Levenberg-Marquardt algorithm and a dataset division 70/15/15 (training/validation/test). For consistency, these parameters were maintained during the training of each NN structure. The proposed hybrid controller was fitted with these NN structures and remarkable performance differences were not noticed. In order to avoid overfitting problems and guarantee a smooth

surface shape, the NN structure with the smallest number of hidden neurons was selected (2-2-4), Figure 11-(a).
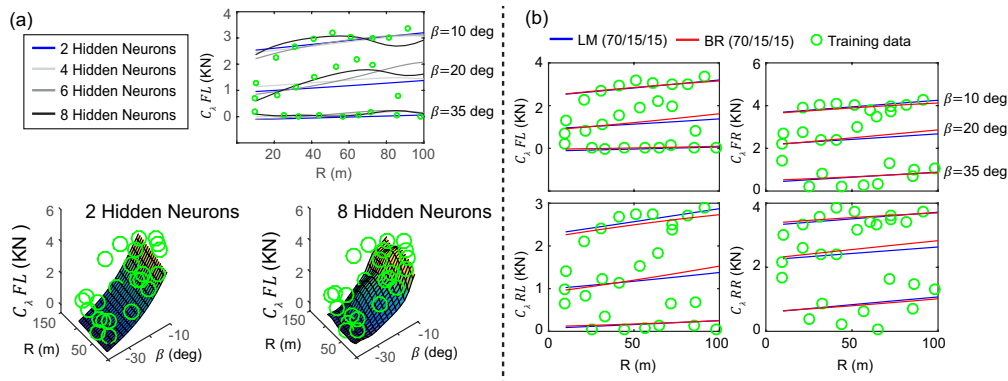

Figure 11: (a) Outputs of NNs with different number of hidden neurons. (b) Outputs of the selected 2-2-4 NNs trained with the Levenberg-Marquardt (LM) and Bayesian Regularisation (BR) routines.

After fixing the NN structure, different training methods (Levenberg-Marquardt (LM) and Bayesian Regularisation (BR)) were compared and significant differences were not observed, Figure 11-(b). The BR training method was executed in Matlab with the function *trainbr*. Finally, similarly to the process described in Section 4.2.1, a total number of 50 NNs were trained with different initial weights (randomly generated by the *trainlm* function) and the average output from these structures was taken as a reference [40]. The NN exhibiting the lowest NRMSE with respect to the average value was implemented in the final intelligent drift controller.

4.3. Controller tuning

The structure presented in Figure 2 was implemented in *Simulink* and *IPG-CarMaker* using the CM4SL library. The simulation time was set to 1 ms and white Gaussian noise (Table 5) was added to the virtual signals in order to assess the controller performance to track noise-corrupted references (body slip, wheel speed, yaw rate and vehicle velocity). For simplicity, noise-free road curvature and lateral deviation error signals were taken from the simulation environment.

The PID and MPC controllers were carefully tuned following a systematic trial-and-error process. Specifically, the MPC was tuned first in open loop,

removing the reference curvature corrections provided by the PID path following controller. The terminal cost $\overline{\mathbf{Q}}$ necessary to incorporate an infinite preview horizon was determined in first place. A Linear Quadratic Regulator (LQR) constrained with saturation functions was constructed first, and a preliminary linear feedback gain $\mathbf{K_{lqr}}$ was obtained after solving the steady-state Ricatti equation (See [54, 5] for further details). After that, the derivation presented in [44] was followed, and the terminal cost ($\overline{\mathbf{Q}}$) was calculated solving the Lyapunov equation (38),

$$\overline{\mathbf{Q}} - (\mathbf{A} + \mathbf{B}\mathbf{K_{lqr}})^T \overline{\mathbf{Q}} (\mathbf{A} + \mathbf{B}\mathbf{K_{lqr}}) = \mathbf{Q} + \mathbf{K_{lqr}}^T \mathbf{R} \mathbf{K_{lqr}} \qquad (38)$$

The values of the matrix K_{lqr} are presented below:

$$\mathbf{K_{lqr}} = \begin{bmatrix} 2.6102 & 1.1559 & -13.6908 & 9.6976 & -0.0042 & 0.0137 & -0.0156 \\ 16.3130 & -3.2610 & 60.8824 & -0.0042 & 7.8221 & -0.0607 & 0.1466 \\ 9.5240 & 15.0613 & -199.9869 & 0.0137 & -0.0610 & 5.7538 & -0.2912 \\ 34.0090 & -23.3092 & 308.1216 & -0.0157 & 0.1470 & -0.2913 & 3.8358 \\ -0.3358 & 0.1225 & 0.2890 & -0.0005 & -0.0018 & -0.0029 & -0.0025 \end{bmatrix} \tag{39}$$

The MPC preview time N_p and preview control sequence N_c were set to 50 and 1 respectively, and the MPC discretisation time was set to $T_{MPC} = 0.02$ s in order to keep an acceptable computational load. Such discretisation time remains within the minimum limits defined in [55], as the vehicle yaw resonance frequency lies around $1Hz$. The intelligent drift system was run in real time on a computer: Intel(R) Core(TM) i7-3632QM CPU at 2.20Ghz, Simulink 2016 64 bits version, and IPG-CarMaker 5.1.3 version. The MPC was fine-tuned to reduce the body slip overshoot and ensure a smooth convergence to the drift references. The values presented in Table 6 were finally obtained.

Table 6: Diagonal terms of the MPC $\mathbf{Q}$ and $\mathbf{R}$ matrices.

Q_1	Q_2	Q_3	Q_4
1	100	1	1
Q_5	Q_6	Q_7	R_1
1	1	1	0.01
R_2	R_3	R_4	R_5
0.01	0.01	0.01	$1e4$

Where $\mathbf{Q}$ and $\mathbf{R}$ are the MPC tuning matrices. These are concatenated to form the total $\hat{\mathbf{Q}}$ and $\hat{\mathbf{R}}$ weighting matrices presented in expressions (27-29) in the following way [44],

$$\hat{\mathbf{Q}} = \begin{bmatrix} \mathbf{Q} & 0 & \cdots & 0 \\ 0 & \mathbf{Q} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \overline{\mathbf{Q}} \end{bmatrix}, \hat{\mathbf{R}} = \begin{bmatrix} \mathbf{R} & 0 & \cdots & 0 \\ 0 & \mathbf{R} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{R} \end{bmatrix} \tag{40}$$

Finally, the upper-level PID controller was incorporated and tuned in several path-following test cases. The values presented in Table 7 were found to provide accurate path tracking capabilities for the proposed intelligent drift controller.

Table 7: PID Tuning parameters

K_p	K_d	K_i
$4e-3$	$2e-2$	$1e-4$

The tuning procedure described in this subsection was performed with the driverless drift control system trained on the type 4 road terrain. For simplicity, the same tuning was tested on the drift control system trained on the type 5 road terrain and good results were obtained. A gain-scheduling approach will be evaluated in the future to adapt the controller parameters depending on the identified friction characteristics.

5. Results

The performance of the driverless drift controller was assessed in IPG-CarMaker under a wide range of manoeuvres performed in two different loose surfaces (terrains 4 and 5, Figure 6). The catalogue of manoeuvres presented in Table 8 was executed on the terrain 4 in the first place. In these tests, the performance of the system was evaluated assuming a known road terrain. In total, #4 manoeuvres covering different radii and reference body slip angles were selected. These were grouped into open loop (no path tracking) and closed loop (path tracking and drift control) tests. This preliminary evaluation on terrain 4 was completed with a phase-space coverage analysis, Section 5.2.1.

After that, the ability of the proposed system to cope with drastically-different road terrains (using the terrain classifier presented in Section 3.5) was assessed. The proposed terrain-based adaptation strategy was compared to a non-adaptive one in a clothoid test case executed on a type 5 terrain. Several tests were also performed to study the system robustness to small variations in the peak surface friction. Moreover, as parameters like the vehicle mass might vary during real operating conditions, additional simulations were executed to evaluate the system performance under these variations.

Table 8: Catalogue of manoeuvres executed on a type 4 terrain.

Test	Path Following	$|\beta|$ (deg)	R (m)	Description
#1	OL	35	$10 - 100$	Radius ramp at constant body slip.
#2	OL	15	$10 - 100$	Radius ramp at constant body slip.
#3	CL	35	$100 - 20$	Decreasing radius clothoid at constant body slip.
#4	CL	$35 - 15 - 35$	30	Sinusoidal body slip at constant radius.

5.1. Open Loop Drifting Manoeuvres

The proposed system was evaluated first in open loop conditions (i.e. without a predefined reference path). The main aim of this evaluation was to assess the open-loop ability of the proposed structure to track different radii and body slip references. The results corresponding to the test #1 (open loop radius ramp reference) are presented in Fig. 12. The system stabilises the car around the reference body slip angle during an extensive continuous radius range ($R \in [10, 100$ m]). This demonstrates the suitability of the proposed *AI drift references block* to capture and generalise the drifting equilibria in the studied ranges from a reduced number of drifting manoeuvres. Similar results were obtained on the second open-loop test (graphs are omitted here due to space limitations). The NRMSE of the tracked states obtained in these tests can be consulted in Table 9.

5.2. Closed Loop drifting manoeuvres

The closed loop evaluation of the proposed system (path-following and drift control) is presented in the following. The results corresponding to the
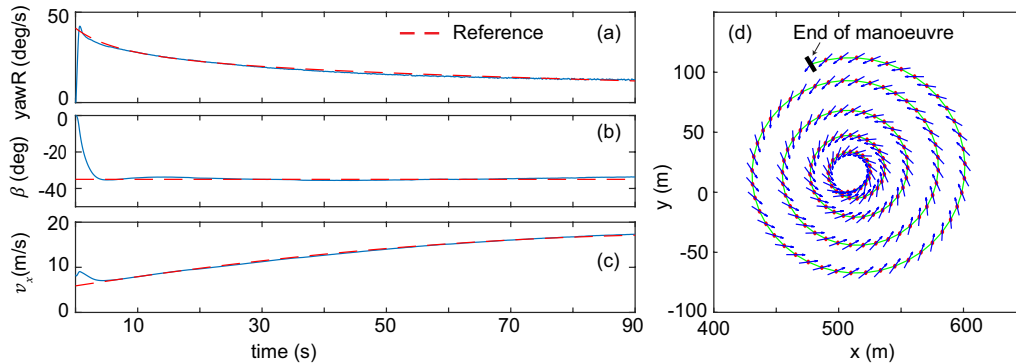
29

Figure 12: Test #1, Open Loop ramp radius test with high body slip reference $|\beta| = 35$ degrees. (a) Yaw rate, (b) body slip angle, (c) longitudinal velocity, (d) vehicle trajectory.

test number #3 (radius reduction clothoid at high body slip reference) are depicted in Fig. 13. Overall, the performance of the proposed system is remarkable. The vehicle is stabilised around the target body slip reference and the lateral deviation error e_{lat} is kept within 1 metre during the rest of the manoeuvre. As can be seen in Fig. 15-left, the vehicle is able to complete the clothoid segment and maintain a drifting motion simultaneously.

Most importantly, this intelligent system is able to perform a complicated driving task (clothoid segment) that was not included in the NNs training dataset (formed exclusively by constant radius drifting manoeuvres). Thus, the proposed hybrid system presents a larger operating envelope than other agents trained with methodologies such as RL, which would be able to execute exclusively the tasks learned during the training stage. Authors envisage that the proposed methodology can help to develop highly-skilled autonomous vehicles from a reduced number of standardised manoeuvres. This can contribute to (a) the reduction of the development costs of these systems and (b) facilitate the training procedure of these systems (using average test drivers that might be unable to perform drift control and path following tasks along complex road segments of arbitrary geometries).

The results regarding the test number #4, constant radius path following with sinusoidal body slip reference, are depicted in Figure 14.
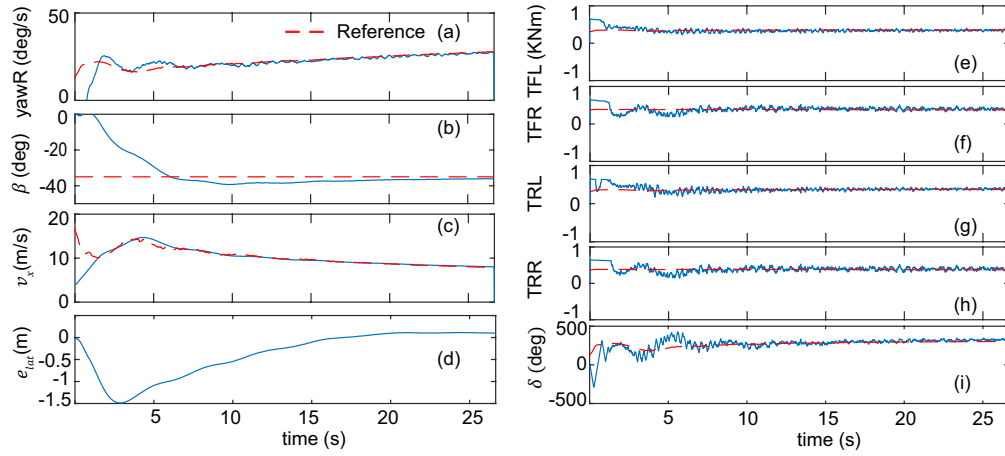
30

Figure 13: Test #3, Closed Loop decreasing-radius clothoid test. High body slip reference $|\beta|$ =35 degrees. (a) Yaw rate, (b) body slip angle, (c) longitudinal velocity, (d) lateral deviation error, (e) Front-left wheel torque, (f) Front-right wheel torque, (g) Rear-left wheel torque,(h) Rear-right wheel torque and (i) Steering wheel angle.
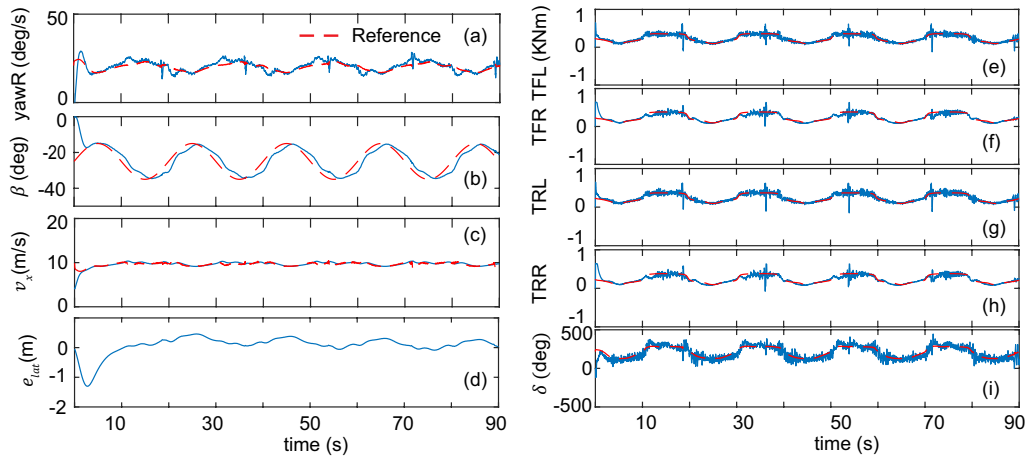

Figure 14: Test #4, Closed Loop constant radius test. Sinusoidal body slip reference $|\beta|$ =35-15 degrees at 0.05 Hz. (a) Yaw rate, (b) body slip angle, (c) longitudinal velocity, (d) lateral deviation error, (e) Front-left wheel torque, (f) Front-right wheel torque, (g) Rear-left wheel torque,(h) Rear-right wheel torque and (i) Steering wheel angle.

Once again, the upper-level PID controller maintains the lateral deviation error within 1 metre in spite of the changing body slip reference. Regarding

31

the latter signal, a slight delay between the reference and the vehicle body slip is noticed during the execution of this test. Additional investigations regarding the dynamic response of this system to changes in the references will be covered in the future. The trajectories of the closed-loop tests are depicted in Figure 15. As can be observed in Figure 15-left, the vehicle is initialised in a straight line and the proposed intelligent drift system drives the vehicle autonomously to the drift equilibrium. This is a significant advantage with respect to previous works on drift control [10], where expert open-loop inputs are required to build up a certain body slip angle before triggering the controller action.
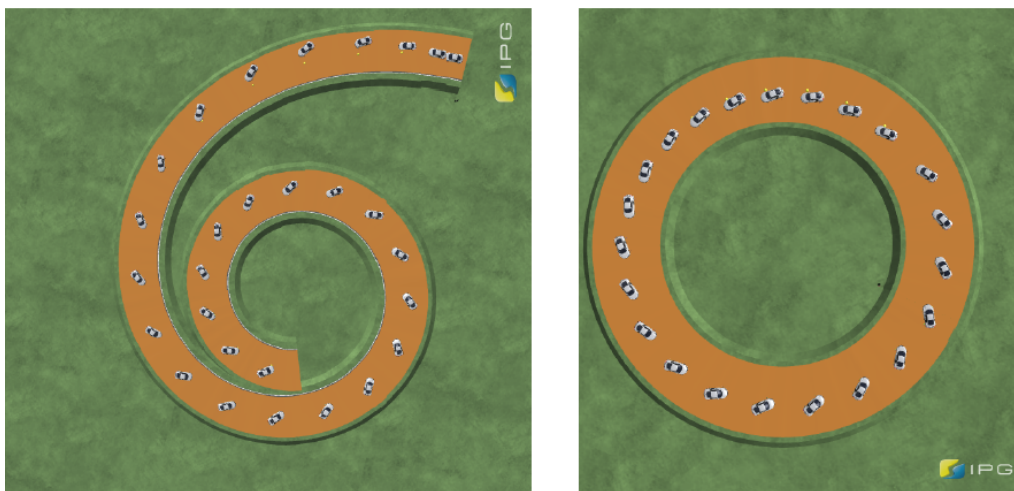

Figure 15: Left: Test #3, Decreasing-radius clothoid test with high body slip reference. Right: Test #4, Constant-radius test with sinusoidal body slip reference.

Finally, the NRMSE of the tracked vehicle planar motion states are presented in Table 9.

Table 9: Normalised root mean square error (%) of the tracked vehicle planar motion states.

Test	$e_{\dot\psi}$	e_{vx}	e_β
#1	1.76	0.79	1.80
#2	1.71	0.57	3.01
#3	4.41	3.45	6.37
#4	7.66	2.35	8.45

For consistency, and with the aim to focus on the body slip tracking capabilities of the proposed system, the initial stabilisation period was not considered. Therefore, the state errors were calculated between $t > 5$ s and t_{end} in all the tests presented in this paper. Expectedly, the largest body slip error is observed on the test (#4), where a time-varying body slip angle is tracked. Nevertheless, this error is kept below the 10% error band and can be considered acceptable for the proposed application. Reduced state errors were obtained in the rest of tests. As similar results have not been found in the literature (previous drift control works considered time-invariant body slip and radius references), these errors will be taken as a reference for further refinements of the proposed system.

5.2.1. Phase-space coverage

As was remarked in Section 3.4, studying the closed-loop stability of the complete driverless model is not trivial. A phase-space coverage analysis was performed to evaluate the system convergence to the desired path and drift equilibria under initial errors on the lateral deviation and vehicle states. Specifically, the vehicle is expected to follow a thirty-metre circular trajectory maintaining a high body slip angle $|\beta| = 35$ degrees. The road width was set to 10 metres and the trajectories leaving the road boundaries were considered unacceptable. The vehicle was initiated in straight line conditions (null body slip angle and null yaw rate) at different initial speeds and lateral deviation errors, Figure 16-a.

Specifically, the grid formed by the vector of initial longitudinal velocity errors (vehicle speed at the start of the simulation minus target speed at steady-state drift equilibrium) [-20:5:20] (km/h) and vector of lateral deviation errors [-8:2:8] (m) was simulated in the proposed scenario. The three-dimensional trajectories of the body slip, lateral deviation, and longitudinal velocity state errors are depicted in Figure 16-b. The distortion of the longitudinal velocity error Δv_x is caused by the action of the upper-level PID controller, which modifies the tracked references to reduce the lateral deviation error. As can be noticed, the proposed system converged to the reference trajectory for the majority of initial errors, Figures 16-c and 16-d. Additional simulations were repeated in a wide range of road radii and similar results were obtained, evidencing the ability of the proposed intelligent drift controller to drive the vehicle to the reference path in a finite time.

Remark: The proposed system converged to the desired equilibrium conditions for the majority of initial errors considered. Nevertheless, for high
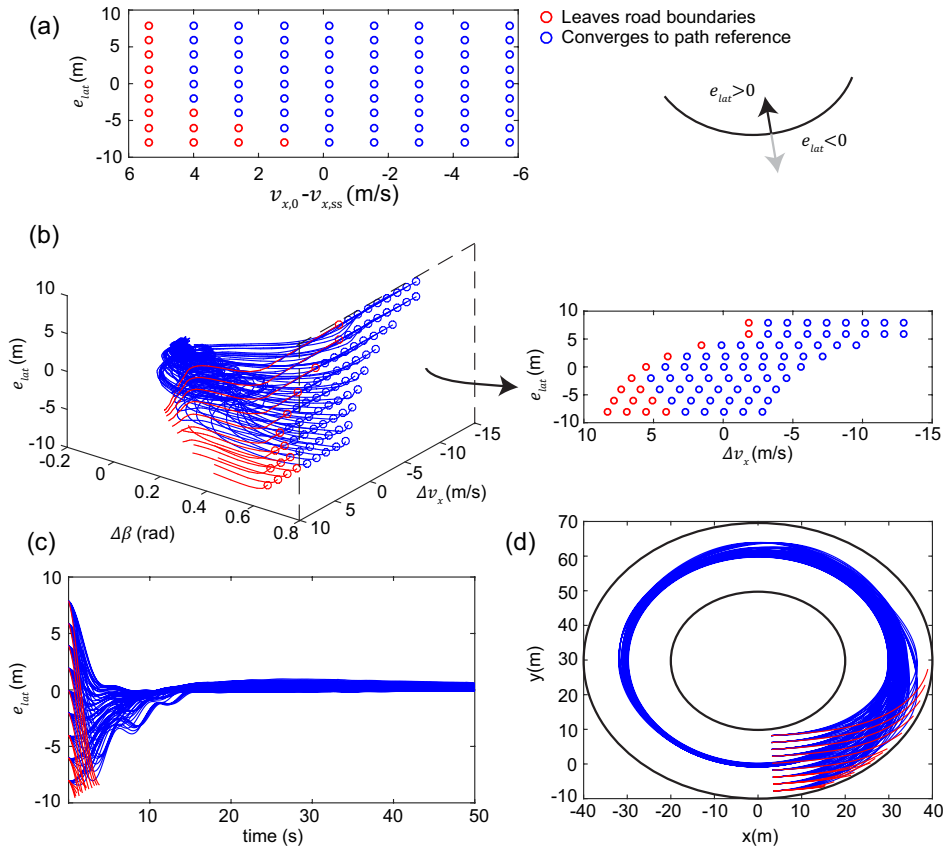
Figure 16: Phase-space coverage analysis. (a) Initial longitudinal velocity and lateral deviation errors, (b) Three-dimensional state error trajectories, (c) time histories of the lateral deviation error, (d) drift controller trajectories.

positive speed errors, the system was unable to converge to the reference trajectory and left the road. Metaheuristic optimisation routines will be employed in the future to optimise the controller performance and elucidate whether the phase-space limits observed in this section are due to the physical limitations of the system or caused by the preliminary non-optimal controller tuning.

The previous analysis was executed on a type 4 terrain with the corresponding set of NNs. Additional simulations were performed on a type 5 terrain to assess the convergence ability of the system with the second set of NNs and satisfactory results were obtained.

34

5.3. Robustness assessment

The ability of the driverless drift control system to adapt to drastically-different terrains or cope with uncertainties in the vehicle parameters was assessed in this subsection. The closed-loop path following clothoid test case (#3, Table 8) was selected for this purpose.
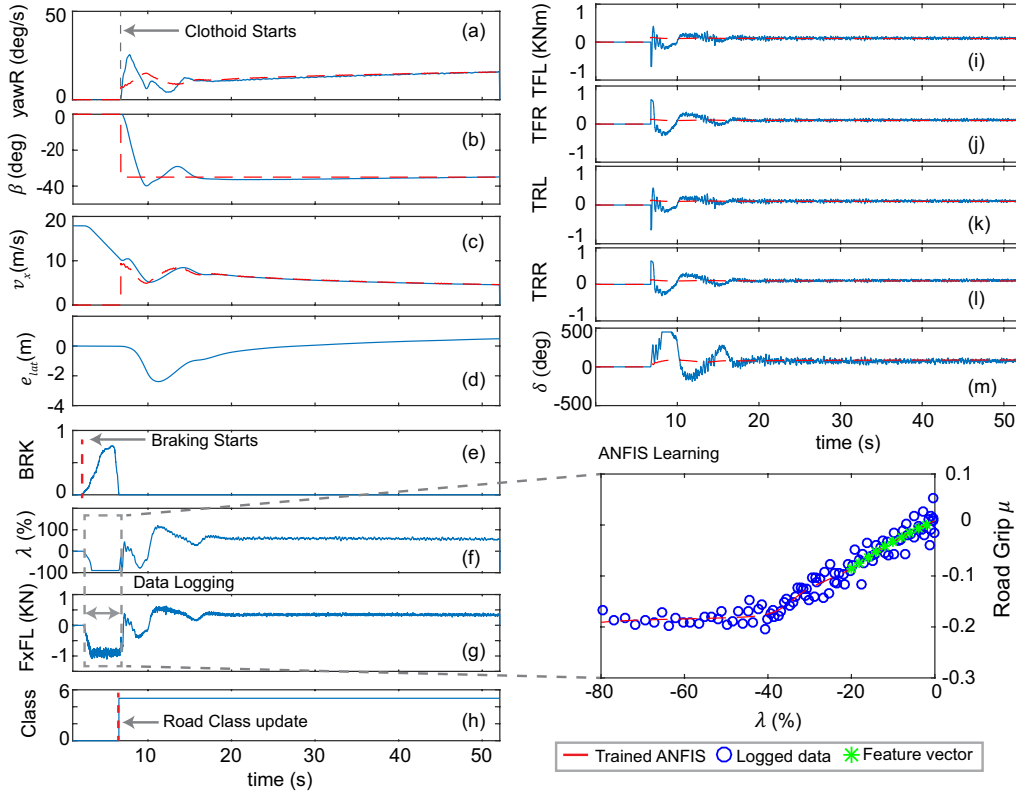
5.3.1. Terrain-based adaptation

Figure 17: Time histories of the clothoid test case simulated in low-mu loose surface (terrain 5) with the proposed friction-based adaptive drift control system. (a) Yaw rate, (b) body slip angle, (c) longitudinal velocity, (d) lateral deviation error, (e) Braking pedal position, (f) Front-left longitudinal slip, (g) Front-left wheel force, (h) Identified road class, (i) Front-left wheel torque, (j) Front-right wheel torque, (k) Rear-left wheel torque,(l) Rear-right wheel torque and (m) Steering wheel angle

In the first place, the proposed system was simulated in a low-mu loose surface (class 5 terrain, Figure 6). The vehicle is initialised at a high speed

35

(18 m/s), and a hard braking intervention is performed while the clothoid segment is being approached. During the braking event (t $\approx$ 2 s to t $\approx$ 8 s) the classifier described in Section 3.5 recognises the current road terrain and the references and tyre parameters of the MPC are adjusted accordingly. After that, the vehicle achieves the desired drifting motion and follows the clothoid road segment with minimum lateral deviation, Fig. 17. In order to evaluate the importance of an adaptive terrain-based control strategy, a second simulation eliminating the terrain classifier action was executed. In this case, the system is initialised with the set of NNs trained in the gravel-like terrain (class 4). As can be observed in Figure 18, the non-adaptive system is unable to cover the clothoid segment and leaves the road in an uncontrolled spinning motion.
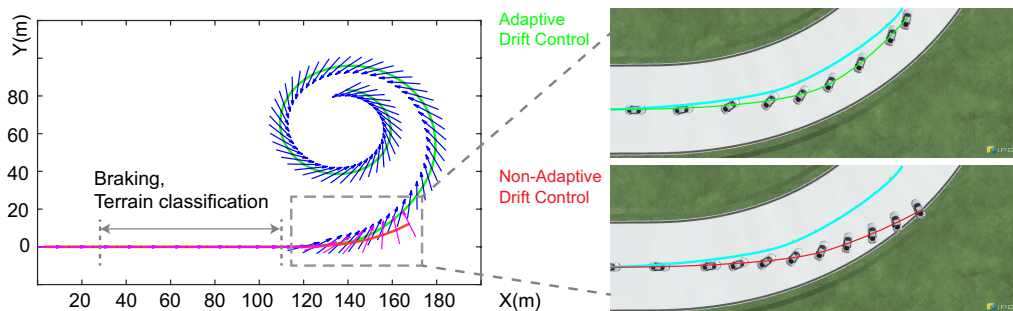

Figure 18: Trajectories of the adaptive (green) and non-adaptive (red) drift control systems.

These results evidence the necessity of incorporating terrain-based adaptive strategies to future highly-skilled autonomous vehicles. Such skills are often exhibited by professional Rally drivers, who adapt their driving style to the current terrain. According to the authors' survey, such strategies have not been presented before in related works, where the drift control systems have been evaluated in an unique terrain.

5.3.2. Slight friction and mass variations

In addition to the previous terrain changes, the proposed system should be robust to slight variations in the friction characteristics (which might not be identified as a different terrain by the classifier). According to recent results on off-road tyre data analysis [56], peak friction values (D factor, expression (32)) can vary within a 0.1 band if different tyres (e.g. summer

tyres, studded tyres) are tested in the same off-road surface (e.g. gravel).
Additional simulations were performed in the clothoid test case increasing
and decreasing by a factor of 0.05 the maximum friction parameter corre-
sponding to the type 4 road terrain. The set of NNs trained with data from
the previous terrain class was employed in the driverless controller for con-
sistency. These simulations were aimed at identifying potential performance
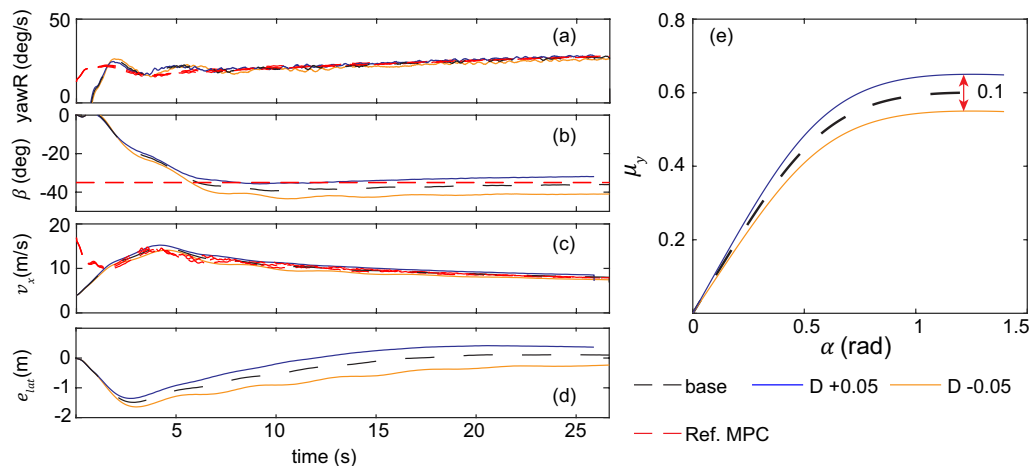problems that might arise if different tyres are used in the proposed system.

Figure 19: Results obtained after varying the maximum friction coefficient (D parameter)
by a factor of 0.05. (a) Yaw rate, (b) body slip angle, (c) longitudinal velocity, (d) lateral
deviation error, (e) lateral friction versus lateral wheel slip curve.

As can be seen in Figure 19, the proposed system is able to stabilise the
vehicle around the target drifting motion and follows the desired path within
reasonable error limits, Table 10.

Table 10: NRMSE of the tracked vehicle states. front*: $\Delta m \approx 150$ kg on front position /
rear*: $\Delta m \approx 150$ kg on rear position.

Test	Configuration	e_ψ	e_{vx}	e_β
1	D+0.05	4.20	3.29	5.58
2	D-0.05	7.02	4.02	18.36
3	$+\Delta m$ front*	3.33	2.02	4.40
4	$+\Delta m$ rear*	4.08	2.73	4.31
5	$+\Delta m$ front* $+ \Delta m$ rear*	3.76	2.55	5.23

Finally, the previous test case was repeated introducing several variations in the vehicle mass and weight distribution ($+\Delta m \approx 150$ kg on front and rear positions). These results are omitted here due to space limitations and the NRMSE of the tracked states can be consulted in Table 10. Overall, the proposed system performed well to changes in the vehicle mass and slight variations in the road friction characteristics (e.g. due to the use of different tyres [56]). In addition, the ability of the proposed system to cope with more significant terrain changes was demonstrated in the previous section.

6. Conclusions

In this paper, a novel data-based approach to teach an autonomous car to drift without requiring any tyre friction information has been presented. Specifically, autonomous drift control has been achieved by means of an artificially intelligent hybrid structure composed of Feedforward Neural Networks and a Model Predictive Controller. The proposed system incorporates a longitudinal dynamics-based road terrain classifier, which adapts the system behaviour depending on the identified terrain, as expert drivers do.

The intelligent structure has been trained using experimental data generated by a real test driver in IPG-CarMaker using a static driver-in-the-loop setup. The suitability of the Neural Networks to capture the drifting equilibrium from noise-corrupted data and provide the drifting references and tyre parameters to the lower-level Model Predictive Controller has been evidenced in a relevant number of test cases. Overall, the proposed intelligent drift controller has exhibited a remarkable performance, being able to perform advanced driving tasks not included in the training dataset even in the presence of significant initial state errors or different road terrains.

6.1. Current limitations and future research steps

In the opinion of the authors, the main drawbacks of the proposed system could be related to vehicle state estimation requirements. The implementation of virtual sensing techniques emphasizing tyre-model-less approaches will be pursued in future research steps. Apart from this, the refinement of the proposed terrain-based adaptive system to cope with a wider range of road terrains will be studied in detail. Furthermore, friction fusion approaches combining longitudinal and lateral dynamics signals will be investigated to extend the operating limits of the current road terrain classifier.

To conclude, authors envisage that the proposed system, and in particular the methodology described to develop and train a self-adaptive intelligent autonomous car, will contribute to accelerating the development of future highly-skilled autonomous vehicles.

Acknowledgment

References

[1] SAE, SAE J3016 Standard: Taxonomy and Definitions for Terms Related to On-Road Motor Vehicle Automated Driving Systems, SAE, 2014.

[2] D. Tavernini, M. Massaro, E. Velenis, D. Katzourakis, R. Lot, Minimum time cornering: the effect of road surface and car transmission layout, Vehicle System Dynamics: International Journal of Vehicle Mechanics and Mobility 51 (10) (2013) 1533–1547.

[3] C. Liang, R. Allen, J. Rosenthal, J. Chrstos, Tire modeling for off-road vehicle simulation, SAE Technical Papers (2004-01-2058).

[4] A. Van Zanten, Evolution of electronic control systems for improving the vehicle dynamic behavior, in: International Symposium on Advanced Vehicle Control (AVEC), 2002.

[5] M. Acosta, S. Kanarachos, M. Fitzpatrick, A hybrid hierarchical rally driver model for autonomous vehicle agile maneuvering on loose surfaces., in: International Conference on Informatics in Control, Automation and Robotics (ICINCO), 2017.

[6] M. Acosta, Research on multi-actuated agile electric vehicles: A drift-based approach to last-moment accident avoidance manoeuvres on loose surfaces, phd progress report, Coventry University, DOI: 10.13140/RG.2.2.14132.96640.

[7] E. Velenis, P. Tsiotras, J. Lu, Modeling aggressive maneuvers on loose surfaces: The cases of trail-braking and pendulum-turn, in: IEEE European Control Conference (ECC), 2007.

[8] R. Hindiyeh, Dynamics and Control of Drifting in Automobiles, Ph.D Thesis, Stanford University, 2013.

[9] J. Edelmann, M. Plochl, Handling characteristics and stability of the steady-state powerslide motion of an automobile, Regular and Chaotic Dynamics 14 (2009) 682–692.

[10] E. Velenis, D. Katzourakis, E. Frazzoli, P. Tsiotras, R. Happee, Steady-state drifting stabilization of rwd vehicles, Control Engineering Practice 19 (11) (2011) 1363–1376.

[11] E. Velenis, Fwd vehicle drifting control: The handbrake-cornering technique, in: IEEE Conference on Decision and Control and European Control Conference, 2011.

[12] E. Velenis, E. Frazzoli, P. Tsiotras, Steady-state cornering equilibria and stabilisation for a vehicle during extreme operating conditions, International Journal of Vehicle Autonomous Systems 8 (2010) 217–241.

[13] A. Gray, Y. Gao, T. Lin, K. Hedrick, H. Tseng, F. Borrelli, Predictive control for agile semi-autonomous ground vehicles using motion primitives, in: American Control Conference (ACC), 2012.

[14] M. Cutler, J.P.How, Autonomous drifting using simulation-aided reinforcement learning, in: IEEE International Conference on Robotics and Automation, 2016.

[15] J. Gonzales, F. Zhang, K. Li, F. Borrelli, Autonomous drifting with onboard sensors, in: Advanced Vehicle Control (AVEC), 2016.

[16] M. Acosta, S. Kanarachos, M. Blundell, Agile maneuvering: From rally drivers to a finite state machine approach, in: IEEE Symposium Series on Computational Intelligence (SSCI), 2016.

[17] I. Chakraborty, P.Tsiotras, J. Lu, Vehicle posture control through aggressive maneuvering for mitigation of t-bone collision, in: IEEE Conference on Decision and Control, 2011.

40

[18] J. Yi, J. Li, J. Lu, Z. Liu, On the stability and agility of aggressive vehicle maneuvers: A pendulum-turn maneuver example, IEEE Transactions on Control Systems Technology 20 (2012) 663–676.

[19] J. Li, J. Yi, Y. Zhang, Z. Liu, Understanding agile-maneuver driving strategies using coupled longitudinal / lateral vehicle dynamics, in: ASME Dynamics and Control Conference, 2011.

[20] K. Berntorp, B. Olofsson, B. Bernhardsson, K. Lundahl, L. Nielsen, Models and methodology for optimal vehicle maneuvers applied to a hairpin turn, in: American Control Conference (ACC), 2013.

[21] K. Berntorp, B. Olofsson, K. Lundahl, L. Nielsen, Models and methodology for optimal trajectory generation in safety-critical road-vehicle manoeuvres, Vehicle System Dynamics: International Journal of Vehicle Mechanics and Mobility 52 (2014) 1304–1332.

[22] M. Acosta, S. Kanarachos, M. Fitzpatrick, Robust virtual sensing for vehicle agile manoeuvring: A tyre-model-less approach, IEEE Transactions on Vehicular Technology. (In press). DOI:10.1109/TVT.2017.2767942, 2017.

[23] Dufournier, Skid Trailer: For tire Characterization and Labelling, Dufournier Technologies, 2012.

[24] J. Kober, J. A. Bagnell, J. Peters, Reinforcement learning in robotics: A survey, The international Journal of Robotics Research 32 (2017) 1238–1274.

[25] P. Escandell-Montero, D. Lorente, J. M. Martínez-Martínez, E. Soria-Olivas, J. Vila-Francés, J. Martín-Guerrero, Online fitted policy iteration based on extreme learning machines, Knowledge-Based Systems 100 (2016) 200–211.

[26] H. Zhang, L. Cui, X. Zhang, Y. Luo, Data-driven robust approximate optimal tracking control for unkown general nonlinear systems using adaptive dynamic programming method, IEEE Transactions on Neural Networks 22 (2011) 2226 – 2236.

[27] X. Zhang, Y. Tao, B. Yang, L. Cheng, Accelerating bio-inspired optimizer with transfer reinforcement learning for reactive power optimization, Knowledge-Based Systems 116 (2017) 26 – 38.

[28] H. Zhang, D. Liu, Y. Luo, D. Wang, Adaptive Dynamic Programming for Control, Springer-Verlag, 2010.

[29] M. Bojarski, D. Del Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang, X. Zhang, J. Zhao, K. Zieba, End to end learning for self-driving cars, Cornell University Library.
URL https://arxiv.org/abs/1604.07316v1

[30] S. Patterson, Google ai expert explains the challenge of debugging machine-learning systems, online: www.networkworld.com (May 2016).
URL https://www.networkworld.com/article/3075413/software/google-ai-expert-explains-the-challenge-of-debugging-machine-learning-systems.html

[31] I. O. for Standardization, BS ISO 26262-7:2011 Road vehicles. Functional safety. Production and operation, BSI Standards publication, 2011.

[32] H. Pacejka, Tire and Vehicle Dynamics, Butterworth-Heinemann, 2012.

[33] M. Acosta, S. Kanarachos, Tire force estimation and road grip recognition using extended kalman filter, neural networks and recursive least squares, Neural Computing and Applications, Springer 2017 (2017) 1–21.

[34] M. Acosta, S. Kanarachos, M. Blundell, Virtual tyre force sensors: A review on tyre model-based and tyre model-less state estimation techniques. (online first), doi: doi.org/10.1177/0954407017728198, 2017, Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering.

[35] C. Ifedi, A High Torque Density, Direct Drive In-wheel Motor for Electric Vehicles, Ph.D Thesis, Newcastle University, 2013.

[36] I. CarMaker, IPG CarMaker, Reference Manual. 14 Steering system. Reference manual version 5.1, IPG Automotive, 2016.

[37] A. B. Dynamics, SP6020 Steering robots. Issue 6. Outline specification, Anthony Best Dynamics Ltd, 2009.

[38] M. Doumiati, A. Charara, A. Victorino, D. Lechner, Vehicle Dynamics Estimation using Kalman Filtering, Wiley-ISTE, 2012.

[39] M. Hrgetic, J. Deur, V. Ivanovic, E. Tseng, Vehicle sideslip angle ekf estimator based on nonlinear vehicle dynamics model and stochastic tire forces modeling, SAE International Journal of Passenger Cars - Mechanical Systems 7 (2014) 86–95.

[40] I. Belic, Neural Networks and Static Modeling, Recurrent Neural Networks and Soft Computing, Dr. Mahmoud ElHefnawi, 2012.

[41] K. Hornik, M. Stinchcombe, H. White, Multilayer feedforward networks are universal approximators, Neural Networks. 2 (1989) 359–366.

[42] J. Lee, Q. Liu, T. Zhang, Predictive semi-analytical model for tire-snow interaction, SAE Technical Paper (2005-01-0932).

[43] K. Jiang, A. Victorino, A. Charara, Adaptive estimation of vehicle dynamics through rls and kalman filter approaches, in: Proceedings of the 18th IEEE International Conference on Intelligent Transport Systems, 2015.

[44] M. Cannon, C21 Model Predictive Control, 4 Lectures Hilary Term, mark.cannon@eng.ox.ac.uk, Oxford University, 2016.

[45] L. Wang, Model Predictive Control System Design and Implementation Using Matlab, Springer, 2009.

[46] S. Antonov, A. Fehn, A. Kugi, Unscented Kalman filter for vehicle state estimation, Vehicle System Dynamics 49 (9) (2011) 1497–1520.

[47] J. Jang, Anfis: Adaptive-network-based fuzzy inference system, IEEE Transactions on Systems, Man, and Cybernetics 23 (1993) 665–685.

[48] T. Lousberg, Building a MATLAB/Simulink-based truck simulator, SEP Report, DC2016-012, Eindhoven University of Technology, 2016.

[49] M. Sayers, Dynamic Terrain Inputs to Predict Structural Integrity of Ground Vehicles, The University of Michigan Transportation Research Institute, 1998.

[50] V. S. GmbH, OpenCRG User Manual, VIRES Simulationstechnologie GmbH, 2015.

[51] M. Acosta, S. Kanarachos, M. E. Fitzpatrick, Accurate virtual sensing of vertical tire forces for enhanced handling dynamics., in: 43rd Annual Conference of the IEEE Industrial Electronics Society, IECON, 2017.

[52] RaceLogic, Vbox 3i GPS System Technical datasheet, Racelogic, 2015.

[53] RaceLogic, RLVBIMU04 Inertial Motion Unit Technical datasheet, Racelogic, 2015.

[54] K. Ogata, Modern Control Engineering, Prentice Hall, 2010.

[55] M. Cannon, 2A2 Discrete Systems, 4 lectures Hilary Term, mark.cannon@eng.ox.ac.uk, Oxford University, 2014.

[56] A. Albinsson, F. Bruzelius, B. Jacobson, J. Fredriksson, Design of tyre force excitation for tyre road friction estimation, Vehicle System Dynamics 55 (2017) 208–230.