Journal Title
XX(X):1–12

DOI: 10.1177/ToBeAssigned
www.sagepub.com/
SAGE

An integrated ANN-UKF vehicle sideslip angle estimation based on IMU measurements

Tommaso Novi[1], Renzo Capitani[1] and Claudio Annicchiarico[2]

Abstract

Vehicle dynamics stability control systems rely on the entity of the so called sideslip angle and yaw rate. However, the sideslip angle can be measured directly only with very expensive sensors, hence, its estimation has been widely studied in the literature. Because of the large non-linearities and uncertainties in the dynamics, model-based methods are not a good solution to estimate the sideslip angle. On the other hand, machine learning techniques require large datasets which cover the entire working range for a correct estimation. In this paper we propose an integrated Artificial Neural Network (ANN) and Unscented Kalman Filter (UKF) observer which uses only IMU measurements and can work as a standalone sensor. The ANN is trained solely with numerical data obtained with a Vi-Grade model and outputs a pseudo-sideslip angle which is used as an input for the UKF. This is based on a kinematic model which makes the filter completely transparent to model uncertainty. A direct integration with integral damping and integral reset value allows to estimate the longitudinal velocity of the kinematic model. A modification strategy of the pseudo-sideslip angle is then proposed to improve the convergence of the filter's output. The algorithm is tested on numerical data aswell as experimental data. The results show the effectiveness of the solution.

Keywords

Sideslip Angle, Unscented Kalman Filter, Neural Networks, IMU

Introduction

Background

Over recent years, safety has become a predominant feature in the design of road vehicles. As a result, there has been an increase in the research and development into Advanced Driver Assistance Systems (ADAS) and Autonomous Vehicles (AV). These can guarantee a high level of security by eliminating human error, particularly when the vehicle is near the limit of adherence. In such conditions, the vehicle is operating in the non-linear range of the tire curve, i.e. large slip angles and, therefore, large vehicle sideslip angle. However, these are highly unstable conditions since even a small variation in the contact patch between tires and road can induce in a rapid variation of available grip[1]. These unstable conditions can be analysed by tracing phase portraits and studying the stability of the equilibrium points. Some researchers such as Edelmann et al.[2] and Voser et al.[3] showed the existence of a bifurcation and demonstrated that above certain values of steering angle and velocity, no state trajectories converge to the equilibrium point since a saddle-node bifurcation is present, making the system highly unstable for any value of the sideslip angle. However, there are many other conditions for which the vehicle is stable within a certain range of sideslip angle and yaw rate. The knowledge of the sideslip angle is, therefore, particularly important for stability control systems such as ESP[4,5] which acts on independent wheel braking and, occasionally, torque to limit the sideslip angle amplitude. This concept and be extended to path following autonomous vehicles at limit handling conditions for which it is important to maintain stability but also guarantee that the vehicle is operating in the tire saturation region. However, the sideslip angle is also important for tracking error purposes. In fact, as for Hu et al.[6], the velocity vector of the centre of gravity should be along the tangent direction on the desired path to prevent a conflict between the sideslip angle control and the yaw rate control.

Previous work

Many types of model-based observers have been developed over the years to estimate the sideslip angle. Wang et al.[7] use a closed-loop state feedback observer based on IMU measurements, longitudinal velocity and an inverted Dugoff tire model[8]. Grip et al.[9] use a non-linear observer based on asymptotic stabilization of the estimation errors guaranteed by means of Lyapunov functions. Other approaches are the ones of Shraim et al.[10] who use sliding mode observers and Zhao et al.[11] who use moving horizon strategies. With all these methods, the estimation is strongly influenced by the vehicle and tire models and system uncertainties. Although modelling errors cannot be completely eliminated with model-based observers, the estimation can still be improved with adaptive methods. Zhang et al.[12] use a gain-scheduling observer based on a linear-parameter-varying system whilst

[1]Dept. of Industrial Engineering, University of Florence (Italy)
[2]Meccanica 42 S.r.l, Sesto Fiorentino (Florence, Italy)

Corresponding author:
Tommaso Novi, Dept. of Industrial Engineering, University of Florence
Email: tommaso.novi@unifi.it

You et al.[13] use online adaptive laws based on yaw rate dynamics and lateral acceleration measurements.
The most common method based on vehicle dynamics models used to estimate the sideslip angle is the Kalman Filter (KF) and its derivations. Many different ways of applying this filter can be found in the literature. Ryu et al.[14] apply linear KFs to a lateral dynamics model-based on measurements of both GPS and IMU. Doumiati et al.[15] applied an Extended Kalman Filter (EKF) directly to a simplified vehicle model in which the road friction coefficient is considered to be known and a Dugoff tire model is used. A better description of the tire dynamics is given by Huang et al.[16] and Li et al.[17] who use the standard Pacejka tire model[18]. To reduce the error given by model uncertainty, the latter use a sideslip angle rate feedback and a sideslip damping for error accumulation. In this case, a separate estimator is used to evaluate the road friction coefficient. Other approaches do not use already existing tire model but use a separate observer to estimate the tire force. Baffet et al.[19] use a Sliding Mode Observer (SMO) whilst Lian et al.[20] calculate the cornering stiffness by means of a Recursive Least Squares (RLS) regression model. These forces are then used in a simplified vehicle model and a EKF is applied to estimate the sideslip angle. As for Julier et al.[21] the EKF is only reliable for systems which are almost linear within the operating frequency range. When the tires are in the saturation region, it would be necessary to operate at a very high frequency to assure linearity, however, this is infeasible with state of the art ECUs. The EKF can also become unstable due to the need to calculate the Jacobians at every time step. To solve these problems the Unscented Transform (UT) and Unscented Kalman Filter (UKF) can be used. A graphical outline of this method and its differences with the EKF can be seen in figure 1.

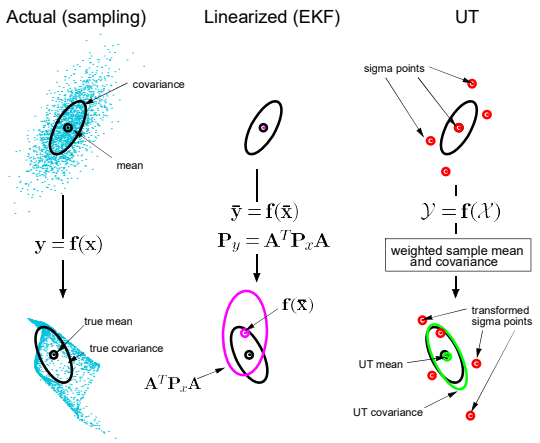

Figure 1. Unscented transform[22]

Morrison et al.[23] and Antonov et al.[24] use this to estimate the sideslip angle on heavy duty trucks and passenger car vehicles using high fidelity vehicle models. Chen et al.[25] add an integral correction with damping and integral value reset in to the model-based UKF to improve model uncertainty.
A completely different approach is the one based on machine learning techniques and, specifically, supervised learning. The main algorithm which has been used in the literature are Artificial Neural Networks (ANN), which have been demostrated to be capable of approximating any function[26]. An application of ANNs for sideslip angle estimation has been done by Du et al[27] who use as inputs to the network IMU measurements as well as wheel velocity and steering wheel angle. In this case only pure lateral behaviour is tested and only on a small amount of manoeuvres on numerical data. Melzi et al.[28] test the algorithm also on combined slip behaviour and on a larger number of manoeuvre types but still limited to ISO standard manoeuvres. They use a hidden layer with hyperbolic tangent transfer functions and a seven degrees of freedom vehicle model to train the network. In their work the network is trained on the same experimental data on which the performance of the estimation is evaluated. The problem about using wheel velocity is that at limit handling conditions these require preconditioning for many situations, e.g. locking of a LSD differential. Also, since these sensors are generally attached to a CAN line of the vehicle, it is not possible to have a standalone sensor. Additionally, using multiple sensors with different frequency acquisition requires some sort of synchronization. Sasaki et al.[29] and Wei et al.[30] use only lateral acceleration and yaw rate which can be measured directly with an IMU. The former train and test the ANN on experimental data only, the latter apply a General Regression Neural Network (GNRR) trained on numerical data and tested on experimental data. These works show that it is possible to estimate the sideslip angle by means of ANN by using only IMU measurements. The results obtained by these researchers seem to be promising, however, they lack of generality and show good results only on very specific situations since the effects of vehicle speed variation and combined slip (no longitudinal acceleration is considered) are not addressed. Additionally, a heuristic approach is used to define the ANN structure in these works. Another common machine learning technique used is Adaptive Neuro-Fuzzy Inference Systems (ANFIS), an example of which is given by Boada et al.[31] who use IMU measurements, steering wheel angle and longitudinal velocity. By means of numerical results the authors show that ANFIS outperforms both ANNs and model-based estimators such as Kalman Filters.
Finally, there are many examples in the literature which integrate the two approaches. Acosta et al.[32] apply a stochastic EKF based on a single track model and use ANFIS for road friction estimation. Boada et al.[33] use ANFIS[31] to estimate "pseudo-sideslip angle" which is fed as an input to a two degrees of freedom dynamic model UKF which filters the signal. GPS, IMU and steering wheel angle measurements are used and the virtual sensor is tested by means of numerical data. The two main issues of the approaches analysed are the following:

1. Model-based observers are subject to model uncertainty and do not work properly when external conditions are varied or unknown
2. Machine learning techniques depend on the training dataset and consequently cannot generalise the problem. The works in the literature show that it is feasible to estimate the sideslip angle by using only IMU measurements but the estimation is correct only when the tested dataset is very similar to the training dataset.

Proposed observer

This paper presents a novel sideslip angle estimator based on an integrated ANN and UKF. The first minor contribution is that only IMU measurements are used for the estimation. The scheme of this observer can be seen in figure 2.

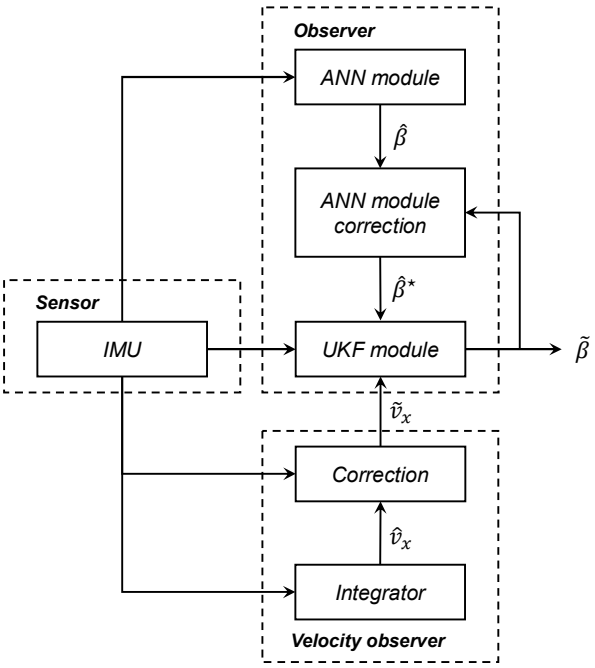

Figure 2. Proposed observer structure

The ANN is trained by means only of numerical data. The second contribution of this paper is an algorithm used to find the best network architecture in a numerical environment. Like in the work of Boada et al.[33], the ANN outputs a pseudo-sideslip angle which is fed as an input to a UKF which corrects the estimation. It will be shown how the sideslip angle time response given by the ANN is accurate, unlike the magnitude which is however corrected by the UKF. The third and main contribution of this paper is that the UKF is based on a kinematic model of the sensor rather than a vehicle dynamics model. The use of a kinematic model allows to be completely free from vehicle model uncertainty (namely the tires) and still correct the estimation given by the ANN. The longitudinal velocity of the kinematic model is estimated by direct integration with integral damping[34] and integral reset value correction.

Due to model uncertainty and different closed-loop behaviour between numerical and experimental environment (driver) the pseudo-sidelslip angle output is saturated to the values seen during training. To improve the convergence of the algorithm a correction strategy for the pseudo-sideslip angle is adopted, this represents a final contribution of this work. The proposed estimator shows very good results in both a numerical and experimental environment also in conditions never seen by ANN. The structure of this paper is as follows. In section 2 the ANN and the algorithm used to define structure and training dataset will be shown. In Section 3 the integrated ANN-UKF algorithm and pseudo-sideslip angle correction will be shown. Finally, in Section 4, the results of the observer on experimental data will be shown.

ANN module

In this section the ANN module will be described. This module outputs a pseudo-sideslip angle which is used as a measurement for a UKF. The neural network performance is strongly influenced by its architecture which is normally heuristically defined. In this work, an algorithm to define the structure of the ANN is developed.

Architecture definition procedure

When designing a neural network, there are three main issues[35]:

1. the network is not "powerful" enough to fit the data
2. the network overfits the training results
3. the networks extrapolates when deployed.

To avoid incurring into these problems an algorithm was developed for the network's structure definition. This requires first dividing the dataset into three sets, a training set over which the network is trained, a validation set used to evaluate overfitting and a test set used to evaluate the network. Then, some general architecture features which do not want to be varied have to be defined. Finally, a set of m parameters of the network's architecture which want to be varied have to be selected (e.g. number of neurons) together with their initial value p^i and percentage variation $k^i \in (0,1)$ for $i = 1, ..., m$. The matrix of architecture parameters $\mathbf{P}$ and variation vector $\mathbf{K} = [k^1, k^2, ..., k^m]'$ can at this point be defined in the following way:

$$\mathbf{P} = \begin{bmatrix} p^1 - k^1 p^1 & p^1 & p^1 + k^1 p^1 \\ p^2 - k^2 p^2 & p^2 & p^2 + k^2 p^2 \\ \vdots & \vdots & \vdots \\ p^m - k^m p^m & p^m & p^m + k^m p^m \end{bmatrix} = \begin{bmatrix} p_l^1 & p_n^1 & p_p^1 \\ p_l^2 & p_n^2 & p_p^2 \\ \vdots & \vdots & \vdots \\ p_l^m & p_n^m & p_p^m \end{bmatrix}$$

Given a loss function $R(\rho, \hat{\rho})$ on any estimated $\hat{\rho}$ and real ρ data, the estimated values of the training $\hat{\gamma}$, validation $\hat{\eta}$, test $\hat{\zeta}$ and overall $\hat{\theta}$ datasets, the correspondent real values γ, η, ζ and θ, and assigning a maximum value for the training dataset loss function l_γ, validation dataset loss function l_η and test dataset loss function l_ζ, algorithm 1 can be defined.

Sideslip angle ANN - general features

Before running the algorithm, some general architecture features have to be defined. To avoid reaching a local minima during the optimization process, five different initial conditions were considered for each network architecture. These conditions were chosen so that the scaled input data applied to the network would have a Gaussian distribution within a range of the transfer function which would guarantee a high initial gradient descent, i.e. speed up the training. To train the network early stopping was used and due to the large amount of data given to the network the scaled conjugate gradient method was used as an optimization algorithm. The loss function $R(\rho, \hat{\rho})$ used to evaluate and train the network is the root mean square error (1).

$$R(\rho, \hat{\rho}) = \sqrt{\frac{\sum_{t=1}^{n} (\hat{\rho} - \rho)^2}{n}} \tag{1}$$

Data: Datasets, General architecture features
Result: ANN estimator
$i = 1$;
while (1) **do**
  for $j \leftarrow 1$ **to** 3 **do**
    $q = [p_{\{i,j\}}, p_{\{i+1,2\}}, ..., p_{\{i+m-1,2\}}]$
    train ANN with parameters $\rightarrow q$
    if $R(\gamma, \hat{\gamma}) < l_\gamma$ **then**
      if $R(\eta, \hat{\eta}) < l_\eta$ **then**
        if $R(\zeta, \hat{\zeta}) < l_\zeta$ **then**
          print Deploy network;
          return *ANN*
        else
          print Network extrapolates;
        end
      else
        print Network overfits;
      end
    else
      print Network failed;
    end
    $\delta\{j\} = R(\theta, \hat{\theta})$
  end
  $s = index[min(\delta)]$
  $p_{\{i+m,2\}} = p_{\{i,s\}}$
  $p_{\{i+m,1\}} = p_{\{i,s\}} - k_{\{i\}} p_{\{i,s\}}$
  $p_{\{i+m,3\}} = p_{\{i,s\}} + k_{\{i\}} p_{\{i,s\}}$
  $i = i + 1$
end

Algorithm 1: Architecture definition

A validated Vi-Grade vehicle model was used to create the datasets which are composed only of numerical data to reduce cost and time in the development. To keep the computational cost low and be able to run the network on standalone embedded automotive platforms with low computational cost and memory, only one hidden fully connected layer was considered, keeping the number of neurons below one hundred. For this same reason, more advanced architectures such as Long Short Term Memory (LSTM)[36] or Gated Recurrent Unit (GRU)[37] were not taken into consideration.
The ANN developed in this work is based on the work of[28]. The hidden layer uses hyperbolic tangent transfer functions while the output layer uses linear transfer functions. The hyperbolic tangent was chosen since its shape is very similar to typical tire force curves. It is a global transfer function, thus, it activates everywhere except close to the origin meaning that the output will be the sum of all activation functions.
The inputs of the network are those coming solely from an IMU, hence, the accelerations along the three axes and rotational velocities about the three axes. The dataset used is composed of different manoeuvre types, for each manoeuvre type different manoeuvre configurations were used (e.g. for a step steer manoeuvre type, step duration and amplitude define the configuration). The dataset manoeuvre types and parameter range of the different configurations can be seen in table 1.
The manoeuvres were chosen so that the network would be trained over the entire working range of the vehicle. For each input manoeuvre type, a multiple of three different values for each manoeuvre configuration parameter was chosen (within the predefined range). This allowed to divide the data between training and validation so that the latter was always a subset of the former in terms of signal amplitude and frequency. With this procedure the validation set properly evaluated overfitting rather than extrapolation. Finally, a test set was appropriately chosen to evaluate the network performance. The various configurations of the manoeuvres in the test set were not present in the training and validation sets. Concerning the circuit, a different circuit was used to the ones used for training and validation. The test set manoeuvre types and configuration parameter range can be seen in table 2.

Table 1. Input manoeuvres for training and validation sets.

Manoeuvre	Parameter	Unit	Range
Sine steer	Steer amplitude	[°]	5,0 - 75
(Freq. response)	Steer frequency	[Hz]	0,1 - 5,0
Step steer	Final steer angle	[s]	5,0 - 75
(Step response)	Step duration	[s]	0,1 - 1,0
Circuit	Max lat. acc.	[g]	0,1 - 1,0
(Combined slip)	Max long. acc.	[g]	0,1 - 1,0

Table 2. Input manoeuvres for test set.

Manoeuvre	Parameter	Unit	Range
Sine steer	Steer amplitude	[°]	10 - 55
(Low freq. response)	Steer frequency	[Hz]	1,0
Sine steer	Steer amplitude	[°]	10 - 55
(High freq. response)	Steer frequency	[Hz]	3,0
Step steer	Final steer angle	[°]	10 - 55
(Step response)	Step duration	[s]	0,2
Circuit	Max lat. acc.	[g]	0,1 - 1,0
(Combined slip)	Max long. acc.	[g]	0,1 - 1,0

Sideslip angle ANN - structure definition

The general structure of the network is shown in figure 3. After having defined the general architecture, constraints and datasets of the network, algorithm 1 was run to obtain the exact architecture by building matrix **P** with the following parameters in the following order:

- **Par. 1** $\rightarrow$ *Number of input manoeuvres (open-loop)*
- **Par. 2** $\rightarrow$ *Ratio of sine steer to step steer manoeuvres*
- **Par. 3** $\rightarrow$ *Number of hidden layer neurons*
- **Par. 4** $\rightarrow$ *Number of delays*

The first parameter varied was the number of total manoeuvres. Only the open-loop manoeuvre number were varied since these can be reproduced also on a real vehicle unlike the closed-loop manoeuvres which depend on the driver, thus were kept constant. Generally the training improves when increasing the number of manoeuvres, however, with many manoeuvres and not many neurons, the network could underfit the data as it is not powerful

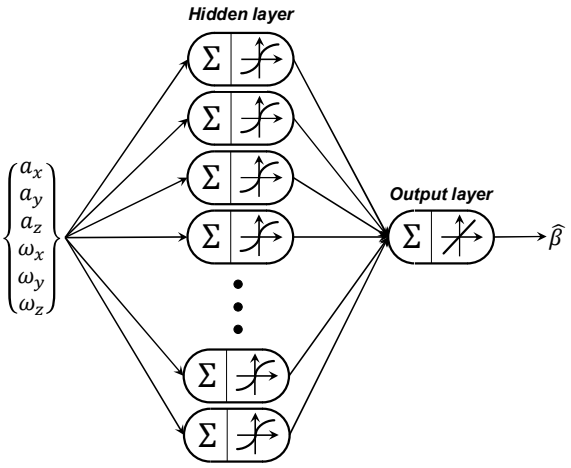

Figure 3. General structure of ANN module

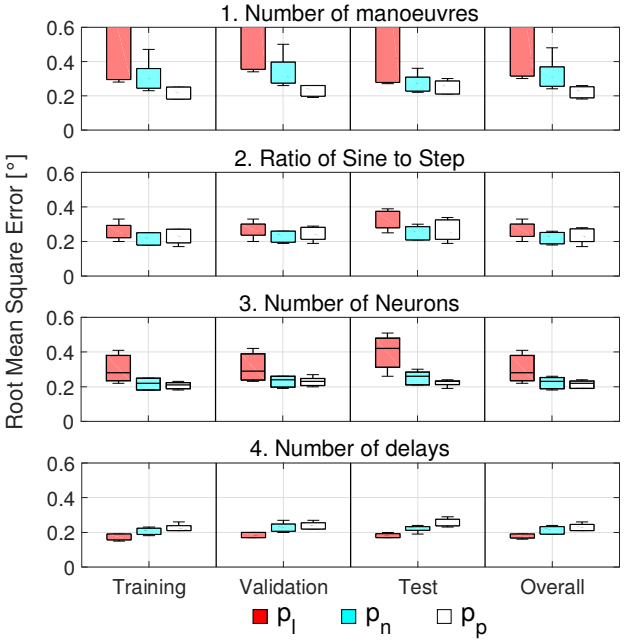

Figure 4. Algorithm results

enough to approximate the function. This also depends on the number of tapped delays as these increase the number of data, hence the dimension of the input space. The next parameter varied was the ratio between number of sine steer configurations and step steer configurations within the predefined range (higher discretization) whilst keeping the total number of manoeuvres approximately the same. It was not possible to maintain the exact same number of manoeuvres since multiples of three for each parameter variation was necessary. The reason why this ratio was chosen as a parameter is that frequency response of a non-linear system is harder to identify respect to the step response[38]. The next parameter which was varied is the number of neurons. The higher the number the more powerful the network, meaning it can approximate more complicated functions, however it is also easier for the system to overfit. The lower the number of neurons, the harder it is to overfit but it is also more probable that the network will underfit and not be powerful enough. The main factor which causes a ANN to overfit, is the ratio between input data and neurons, for this reason, the number of neurons were varied after the total number of manoeuvres. Finally, the number of tapped delays, which give some sort of time memory to the ANN, was varied, considering a frequency acquisition of 100Hz, i.e. 10 delays correspond to 0,1s.

Sideslip angle ANN - results

Running the algorithm, the system converged after one sweep of every parameter with $l_\gamma = l_\eta = l_\zeta = 0.2°$. The final structure of the ANN obtained with the proposed algorithm is the following:

- **Par. 1** $\rightarrow$ 450 (1 hour and 25 minutes driving)
- **Par. 2** $\rightarrow$ 0,5 (225 sine steers and 225 step steers)
- **Par. 3** $\rightarrow$ 90 (hidden layer)
- **Par. 4** $\rightarrow$ 10 (equal to 0,1s in time)

The results of the algorithm are shown in figure 4.
For each step, the loss function of the training set, validation set, test set and overall set are plotted. Every colour of boxplot corresponds to a different network architecture. Boxplots are used to consider the five networks with different initial conditions but same architecture so that it is possible to see the error distribution. Using five different initial conditions for each network configuration allows to evaluate if a network with a certain architecture and initial conditions reaches a local minimum instead of a global minimum, thus guaranteeing great performance for some manoeuvres types but terrible performance for others. For this reason the average of the loss function of the five initial conditions was used as an evaluation criteria in the algorithm. As expected, the performance increased with an increase in the number of input manoeuvres. The variation of the ratio of sine steer configurations to step steer configurations at step two didnt influence much the performance, however, the lowest error was given by an equal number of sine steer and step steer configurations. At step three it can be seen how the larger the number of neurons, the better the performance. No overfitting occured as can be seen by the validation boxplot. This was also due to the fact that a greater number of manoeuvres was chosen. Finally, the number of tapped delays which gave best performance was the lowest one. The results of the deployed ANN on the test set can be seen in figure 5. Specifically, for each manoeuvre type of table 2 (low frequency sine steer, high frequency sine steer, step steer and circuit), only the manoeuvre configuration with the highest values of I/O signals are shown. In these condition, the vehicle is at limit handling and is operating in the non-linear region. For confidentiality reasons, the sideslip angle values have been normalised to the maximum value obtained in all datasets. The ANN obtained with the proposed method shows very good results for all types of manoeuvres in a numerical environment. Due to different external conditions (road friction), model uncertainty (tire model) and closed-loop behaviour (driver) when applying this ANN to the experimental data, the sideslip angle estimated by the ANN saturates to the maximum value seen in the numerical

environment as will be shown in the following section. Thus, the value estimated by this ANN will serve as correction measurement for the UKF.

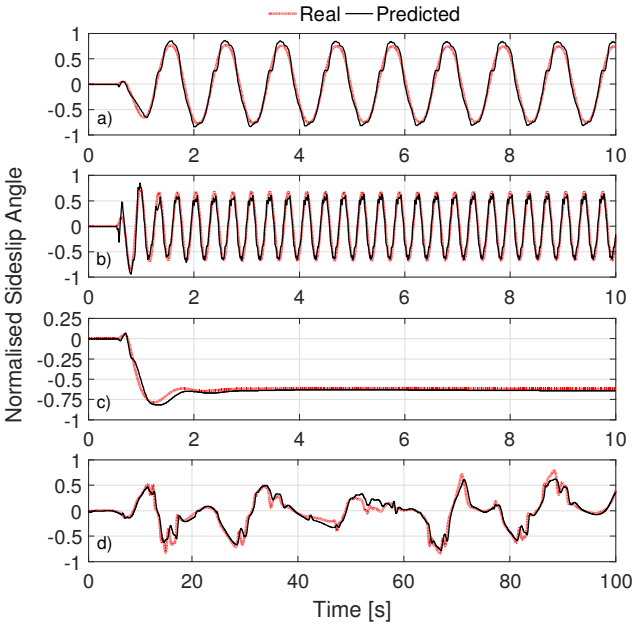

Figure 5. Sideslip angle ANN estimation on test set. a) Low frequency sine steer. b) High frequency sine steer. c) Step steer. d) Circuit.

UKF module

In this section the remaining part of the proposed observer will be discussed. This is a UKF based on a kinematic model which has as inputs the IMU measurements, the pseudo-sideslip angle estimated by the ANN and the longitudinal velocity given by direct integration with integral damping and integral value reset. For a better convergence of the observer, the output of the UKF is used to correct the pseudo-sideslip angle so that the measurement entering the UKF and its output asymptotically tend to the same value.

UKF algorithm

The UKF was first proposed by Julier et al.[39]. Unlike the EKF where the time update is conducted by the linearised system by means of the Jacobi matrix, the UKF uses directly the non-linear function model. The main idea is to use the Unscented Transform to propagate the so-called "sigma points" which are a group of symmetrically distributed points around the previous estimated system states which contain the information of the expected mean value and variance of the system. Let us consider a non-linear system in discrete time form with additive noise (2):

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k, t_k) + \mathbf{v}_k \tag{2a}$$
$$\mathbf{y}_k = \mathbf{g}(\mathbf{x}_k, t_k) + \mathbf{w}_k \tag{2b}$$

where $\mathbf{x}_k \in \mathbb{R}^n$ represents the state vector, $\mathbf{u}_k \in \mathbb{R}^m$ is the input vector, $\mathbf{y}_k \in \mathbb{R}^q$ the measurement vector. The process noise $\mathbf{v}_k$ and measurement noise $\mathbf{w}_k$ are white Gaussian noise, i.e. zero mean and uncorrelated (3):

$$\mathbf{v}_k \sim \mathcal{N}(0, \mathbf{R}_k^v) \tag{3a}$$
$$\mathbf{w}_k \sim \mathcal{N}(0, \mathbf{R}_k^w) \tag{3b}$$

where $\mathbf{R}_k^v$ and $\mathbf{R}_k^w$ are respectively the process noise and measurement noise covariance matrices. The UKF algorithm can be presented considering the state vector $\mathbf{x}_k$ (random variable) having mean value $\tilde{\mathbf{x}}_k$ and covariance $\mathbf{Q}_k$.

1. Initialise mean value and variance matrix (4):

$$\tilde{\mathbf{x}}_0 = \mathbb{E}[\mathbf{x}_0] \tag{4a}$$
$$\mathbf{Q}_0 = \mathbb{E}[(\mathbf{x}_0 - \tilde{\mathbf{x}}_0)(\mathbf{x}_0 - \tilde{\mathbf{x}}_0)^T] \tag{4b}$$

where $\tilde{\mathbf{x}}_0$ is the posteriori estimation of the expected mean value for $k = 0$ and $\mathbf{Q}_0$ is the posteriori estimation of the variance matrix for $k = 0$.

2. Calculate sigma points $\boldsymbol{\mathcal{X}}_{k-1} \in \mathbb{R}^{n \times (2n+1)}$ (5) for $k \in \{1, ..., \infty\}$ according to the following:

$$\boldsymbol{\mathcal{X}}_{k-1} = \begin{bmatrix} \tilde{\mathbf{x}}_{k-1} & \tilde{\mathbf{x}}_{k-1} + \mathbf{A}_{k-1} & \tilde{\mathbf{x}}_{k-1} - \mathbf{A}_{k-1} \end{bmatrix} \tag{5a}$$
$$\mathbf{A}_{k-1} = \sqrt{(n+\lambda)\mathbf{Q}_{k-1}} \tag{5b}$$
$$\lambda = \alpha^2(n + \kappa) - n \tag{5c}$$

where λ is a scaling parameter, the constant α determines the spread of the sigma points around $\mathbf{x}_{k-1}$ and is usually set to a small positive value. The constant κ is a secondary scaling parameter. Many studies have been carried out to calculate the square root of covariance matrix $\mathbf{Q}_{k-1}$. In this paper the Cholesky factorization was used for which a positive Hermitian positive-definite matrix B can be decomposed as $B = LL^+$ with L being a lower triangular matrix with real and positive diagonal terms.

3. Time update by transforming the sigma points with the non-linear functions (6):

$$\boldsymbol{\mathcal{X}}_{k|k-1} = \mathbf{f}(\boldsymbol{\mathcal{X}}_{k-1}, \mathbf{u}_{k-1}, t_k) \tag{6a}$$
$$\boldsymbol{\mathcal{Y}}_{k|k-1} = \mathbf{h}(\boldsymbol{\mathcal{X}}_{k-1}, t_k) \tag{6b}$$

and computing the priori estimation of the expected mean value $\tilde{\mathbf{x}}_k^-$ (7a), variance matrix $\mathbf{Q}_k^-$ (7b) and measurement estimation $\tilde{\mathbf{y}}_k^-$ (7c):

$$\tilde{\mathbf{x}}_k^- = \sum_{i=0}^{2n} W_i^{(m)} \mathcal{X}_{i,k|k-1} \tag{7a}$$
$$\mathbf{Q}_k^- = \sum_{i=0}^{2n} W_i^{(c)} [\mathcal{X}_{i,k|k-1} - \tilde{\mathbf{x}}_k^-][\mathcal{X}_{i,k|k-1} - \tilde{\mathbf{x}}_k^-]^T + \mathbf{R}_k^v \tag{7b}$$
$$\tilde{\mathbf{y}}_k^- = \sum_{i=0}^{2n} W_i^{(m)} \mathcal{Y}_{i,k|k-1} \tag{7c}$$

where the W_i weights are (8):

$$W_0^{(m)} = \lambda/(n+\lambda) \tag{8a}$$
$$W_0^{(c)} = \lambda/(n+\lambda) + (1 - \alpha^2 + \gamma) \tag{8b}$$
$$W_i^{(m)} = W_i^{(c)} = 1/\{2(n+\lambda)\} \qquad i = 1, ..., 2n \tag{8c}$$

where γ is used to incorporate prior knowledge of the distribution of the state vector (for Gaussian distributions,

$\gamma = 2$ is optimal).

4. Measurement update by computing the measurement estimation variance $\mathbf{Q}_{y_k y_k}$ (9a) and covariance matrix $\mathbf{Q}_{x_k y_k}$ between $\tilde{\mathbf{x}}_k^-$ and $\tilde{\mathbf{y}}_k^-$ (9b):

$$\mathbf{Q}_{y_k y_k} = \sum_{i=0}^{2n} W_i^{(c)} [\mathcal{Y}_{i,k|k-1} - \hat{\mathbf{y}}_k^-][\mathcal{Y}_{i,k|k-1} - \hat{\mathbf{y}}_k^-]^T + \mathbf{R}_k^w \tag{9a}$$

$$\mathbf{Q}_{x_k y_k} = \sum_{i=0}^{2n} W_i^{(c)} [\mathcal{X}_{i,k|k-1} - \hat{\mathbf{x}}_k^-][\mathcal{Y}_{i,k|k-1} - \hat{\mathbf{y}}_k^-]^T \tag{9b}$$

and finally calculating the Kalman gain $\mathcal{K}_k$ (10a), posteriori estimation of expected mean value $\tilde{\mathbf{x}}_k$ (10b) and variance matrix $\mathbf{Q}_k$ (10c):

$$\mathcal{K}_k = \mathbf{Q}_{x_k y_k} \mathbf{Q}_{y_k y_k}^{-1} \tag{10a}$$

$$\tilde{\mathbf{x}}_k = \tilde{\mathbf{x}}_k^- + \mathcal{K}_k(\mathbf{y}_k - \tilde{\mathbf{y}}_k^-) \tag{10b}$$

$$\mathbf{Q}_k = \mathbf{Q}_k^- - \mathcal{K}_k \mathbf{Q}_{y_k y_k} \mathcal{K}_k^T \tag{10c}$$

Kinematic model

The model used for the UKF (2) is a pure kinematic model, thus, it does not depend on any tire model. The model describes the kinematics of the vehicle's planar motion as in figure 6.

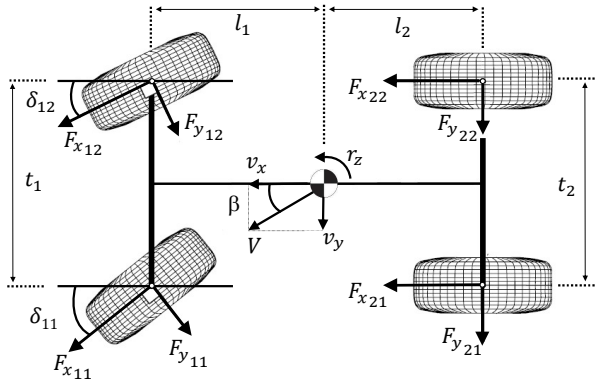

Figure 6. Vehicle model

Given the longitudinal acceleration a_x, the lateral acceleration a_y, yaw rate r_z, longitudinal velocity v_x, lateral velocity v_y and sideslip angle β a set of differential equations (11) can be written in the vehicle's reference system with origin in its centre of gravity.

$$\dot{v}_x = a_x + v_y r_z \tag{11a}$$

$$\dot{v}_y = a_y - v_x r_z \tag{11b}$$

From the definition of β (neglecting roll and pitch motions), v_y can be written as (12):

$$v_y = v_x \tan\beta \tag{12}$$

and consequently its derivative respect to time (13) is:

$$\dot{v}_y = \dot{v}_x \tan\beta + v_x(1 + \tan^2\beta)\dot{\beta} \tag{13}$$

Substituting (12) and (13) in (11) a new set of differential equations is obtained (14):

$$\dot{v}_x = a_x + v_x r_z \tan\beta \tag{14a}$$

$$\dot{\beta} = \frac{1}{v_x(1+\tan^2\beta)}(a_y - v_x r_z - v_x r_z \tan^2\beta - a_x \tan\beta) \tag{14b}$$

applying some trigonometric identities the sideslip angle kinematics can be expressed by (15):

$$\dot{\beta} = \frac{a_y}{v_x} cos^2\beta - \frac{a_x}{2v_x} sin2\beta - r_z \tag{15}$$

thus the state vector becomes $\mathbf{x} = [\beta]$, the input vector $\mathbf{u} = [a_x, a_y, r_z, v_x]$, measurement vector $\mathbf{y} = [\hat{\beta}]$ with $\hat{\beta}$ being the pseudo-sideslip angle, v_x is found by direct integration while a_x, a_y and $r_z$are measured with an IMU. Applying the UKF equations to this system, the estimated sideslip angle $\tilde{\beta}$ is obtained.

Longitudinal velocity observer

The longitudinal velocity is kept external to the UKF to not influence the covariance matrices and because it can be found with good precision by direct integration with integral damping and integral reset value (DI-IDIRV) to eliminate integration drift. First, moving average filters with moving windows of five tenths of a second are applied to the signals coming from the IMU. Then, integral damping is added (16):

$$\dot{v}_x = \int \left[a_x + v_x r_z \tan\beta - \frac{v_x}{\tau} sgn(\dot{v}_x)\right] dt \tag{16}$$

where τ ($\tau \geq 1$) represents a damping coefficient. A fourth order explicit Runge-Kutta numerical integration is used to solve the equation. Finally, the integral value is reset with the steady state value. The condition for steady state[1] is that yaw moment must be zero, i.e. $\dot{r}_z = 0$. When this occurs, the steady state longitudinal velocity v_x^s can be found as (17):

$$v_x^s = \frac{a_y}{r_z} \tag{17}$$

A quasi steady state condition is considered and "activated" at time step k in the following way. Given the time discrete system and a vector $\mathbf{r}_z = [r_z^{(k-l)}, r_z^{(k-l+1)}, ..., r_z^{(k)}]$ composed by the yaw rate at step k and the l previous time steps, the following conditions must be fulfilled (18):

$$|r_z^i| \geq \epsilon \quad i = k-l, k-l+1, ...k \tag{18a}$$

$$\sum_{i=k-l}^{k} |r_z^i| \geq d \tag{18b}$$

with $\epsilon, d \in \mathbb{R}$ being small positive numbers. When this happens, the integral value is reset to the steady state value (17). This integral reset value method is valid under the assumption that:

- the vehicle is not driving in a straight line on a flat road, in which case the sidelip angle is anyhow zero
- the vehicle is a non-neutral steering vehicle, in which case even with external disturbance or banked road the sideslip angle is different to zero but no yaw rate is generated.

Experimental results of this method will be shown in the next section and show the effectiveness of the algorithm.

Pseudo-sideslip angle correction

The pseudo-sideslip angle is based on supervised learning, the hyperbolic tangent function used as transfer function in the ANN has two horizontal asymptotes. Thus, the output of the network is saturated to the maximum and minimum value seen during the training. To increase the performance of the UKF, at time step k the pseudo-sideslip angle $\hat{\beta}_k$ is corrected and a corrected pseudo-sideslip angle $\hat{\beta}_k^\star$ is obtained. This is done with a feedback proportional correction as in the following equation:

$$\hat{\beta}_k^\star = C_k \hat{\beta}_k \tag{19}$$

where the proportional gain C_k at time step k is found with the following system of equations (20):

$$C_k = \begin{cases} 1, & \text{if } |\tilde{\beta}_{k-1} - \hat{\beta}_{k-1}| \geq L_k \\ U_k, & \text{if } C_k \geq U_k \\ \frac{\tilde{\beta}_{k-1}}{\hat{\beta}_{k-1}}, & \text{otherwise} \end{cases} \tag{20}$$

Since the correction is recursive, the lower saturation L_k and upper saturation U_k are used to make the algorithm stable.

Results

In this final section, the experimental setup used to obtain the real data will first be described. After that, the results of the algorithm will be shown for both the longitudinal velocity and sideslip angle estimation. It will be shown how the structure of the estimator shows its benefits against the standalone solutions.

Experimental setup

The experimental data was obtained on a sport saloon which can be seen in figure 7. The vehicle was equipped with a two axis non-contact optical sensor with halogen lamp for sideslip angle measurement. Specifically the Kistler Correvit S-Motion which has a measurement accuracy angle of $\pm 0.2°$ and angle resolution of $\pm 0.1°$ was used. The measurement is guaranteed with a minimum speed of ± 0.1km/h and a maximum speed of ± 250km/h, accelerations up to ± 18g and angle speeds up to $\pm 300°$/s, with a frequency of 500Hz. The sensor was mounted in front of the vehicle's front axis, hence, the velocity measurements were translated in the centre of gravity by means of the rigid body laws. The vehicle was also equipped with Kistler HF Sensors on both sides of the vehicle. These are Optical Laser Height-Sensors which were used to validate the Vi-Grade model used to train the ANN. Also a Kistler MSW Sensor was mounted behind the steering wheel. This is a non-contact optical steering angle sensor capabale of measuring both steering wheel torque and angle. This sensor was also used for numerical model validation.

To measure accelerations and rotational velocities the MPU-6050 three axis IMU of InvenSense was used. The noise characteristics of the IMU are shown in table 3.

This IMU was installed in the M42A2C10 ECU of Meccanica 42 which is composed of a AVR 32 bit CPU, 512Kb flash memory, 64Kb EEPROM and 64Kb RAM. The maximum clock signal of the the ECU is 66MHz. This ECU is also provided with a 4.1 bluetooth, a H-bridge with

Figure 7. Experimental setup

Table 3. InvenSense MPU-6050 datasheet.

Parameter	Unit	Value	Sensor
Power spectral density	400	$\mu g/\sqrt{Hz}$	Acceler.
Total RMS noise	0.05	°/s-rms	Gyrosc.
Low-frequency RMS noise	0.033	°/s-rms	Gyrosc.
Rate noise spectral density	0.005	$°/s/\sqrt{Hz}$	Gyrosc.

30A and 40V of maximum electric capacity and 2 CAN interfaces. The acquisition system used to convert the signals is shown in figure 8. It is composed a Vector VN1640A CAN/LIN interface. The device has a time stamp accuracy of 1μs and can acquire CAN signals with a rate up to 2Mbit/s. The testing was done at Marzaglia with professional test drivers.

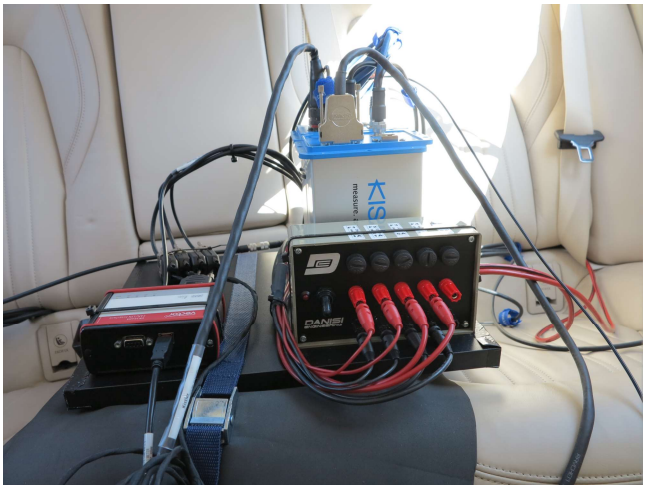

Figure 8. Acquisition system

IMU measurements

The acquisitions obtained during the test with the described IMU are represented in figure 9. Specifically, the yaw rate, longitudinal acceleration and lateral acceleration are shown. A moving average filter with a moving window of five tenths of a second was used to smoothen the inputs fed to the estimator. This filter was chosen since it requires low memory and it is capable of smoothening the high

frequency oscillations introducing very little phase lag given the small time window used. The results obtained from the IMU were compared with the IMU installed in the Correvit sensor to assure no drift in the gyroscope and accelerometer measurements. The output of the Correvit sensor are the longitudinal and lateral velocity. Thus, the sideslip angle was easily calculated (12) while the longitudinal velocity was used as a comparison value for the DI-IDIRV estimator.

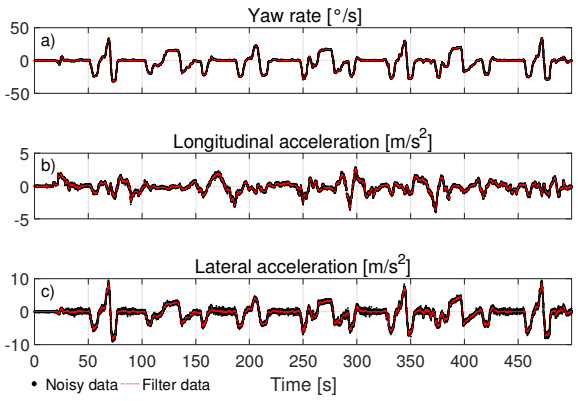

Figure 9. IMU measurements. a) Yaw rate. b) Longitudinal acceleration. c) Lateral acceleration

The other IMU measurements, i.e. vertical acceleration, roll rate and pitch rate were also filtered with the same moving average filter. These signals are used only in the ANN while for the UKF planar motion was considered, therefore, these motions were neglected. The experimental data of these signals can be seen in figure 10.

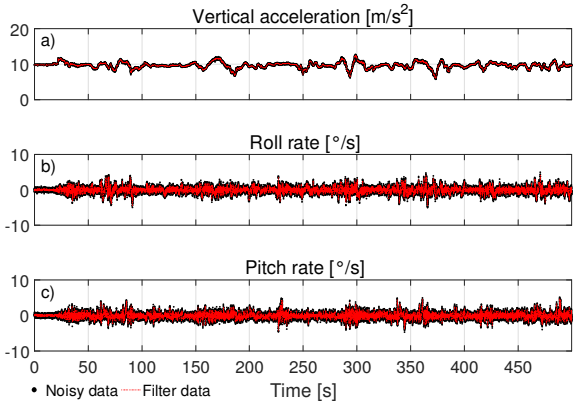

Figure 10. IMU measurements. a) Vertical acceleration. b) Roll rate. c) Pitch rate

Longitudinal velocity estimation

Based on the DI-IDIRV observer, the longitudinal velocity of the vehicle was estimated. The same moving average filter used to filter the IMU measurements was used to filter the estimator's feedback. Note that for the UKF dynamics (15), the longitudinal velocity appears in two terms in the denominator. Hence, it is important for the estimated longitudinal velocity to not diverge into very small values for correct sideslip angle estimation and numerical integration stability. If the estimated longitudinal velocity is larger than the real value, the sideslip angle derivative will assume a lower value respect to the real one. The term with the longitudinal acceleration generally gives a small contribution to the dynamics for normal values of sideslip angle due to the sine multiplication. The results of the estimation can be seen in figure 11 where the integral resets are also illustrated. With proper tuning, the integral resets correspond to the real longitudinal velocity. Without the integral reset, the observer is highly unstable and the estimated longitudinal velocity quickly diverges, making the error in the sideslip angle estimation very large. The moving average filter applied to the inputs and the integral damping allow to improve the numerical integration stability.

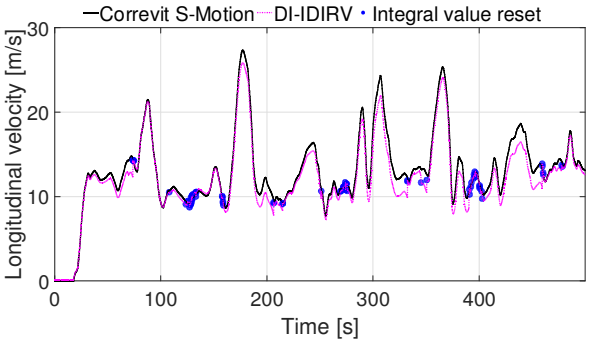

Figure 11. Longitudinal velocity estimation

The estimated velocity profile is very similar to the measured one. Both the velocity variations and quasi steady-state behaviour are well observed. To better see the efficiency of the estimation, a quantitative analysis by means of different error estimates is shown in table 4. Specifically, the mean absolute error (MAE), the root mean square error (RMSE), and the correlation factor R have been calculated.

Table 4. Longitudinal velocity estimation error.

Observer	MAE [m/s]	RMSE [m/s]	R
DI-IDIRV	0.88	1.13	0.98

Due to the small relative magnitude of the error estimates, it can be concluded that the proposed observer efficiently estimates the longitudinal velocity by using only IMU measurements.

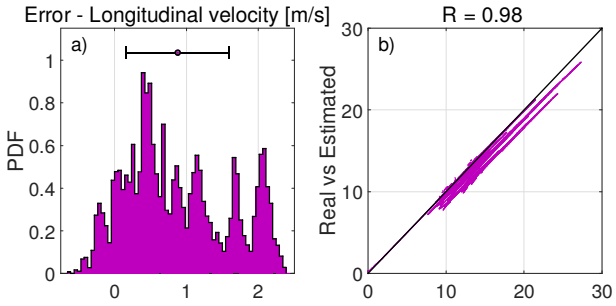

Figure 12. Longitudinal velocity estimation error. a) Probability density function of the estimation error. b) Real value vs estimated value

This is also confirmed by the results shown in figure 12. Here, the probability density function (PDF) of the velocity

estimation error is represented. It can be noted how the error does not resemble a normal distribution and is random. However, the errors are bounded between -0.5m/s and 2.5m/s approximately. The lower error equal to -0.5m/s is very important since, as already mentioned, the longitudinal velocity appears only as a denominator in the sideslip angle dynamics of the UKF. Finally, it is important to mention that the estimated velocity is independent of the velocity magnitude.

Sideslip angle estimation

After having discussed the longitudinal velocity estimation and the IMU data filtering, the sideslip angle estimation can now be discussed. The results of the pseudo-sideslip angle, the corrected pseudo-sideslip angle and UKF estimated sideslip angle will be shown and compared. The results of the sideslip angle estimation is shown in figure 13, for confidentiality reasons, the normalized sideslip angle is plotted. The normalization has been done respect to the maximum value seen during the network's training shown in figure 5. Because of external disturbances, model uncertainty and, mostly, different closed-loop behaviour, the magnitude of the sideslip angle is often greater values than those seen in the numerical environment. For this reason, despite the great effort to avoid ANN extrapolation, the ANN is not capable of estimating correctly the experimental data in terms of magnitude. However, the sideslip angle time response given by the ANN is very accurate. In fact, the sideslip angle variations are correctly estimated. As expected, the pseudo-sideslip angle is saturated to the value seen during the network training phase while the time response is very good. Thus, the proposed strategy for the neural network structure definition can be considered efficient despite needing a magnitude correction.

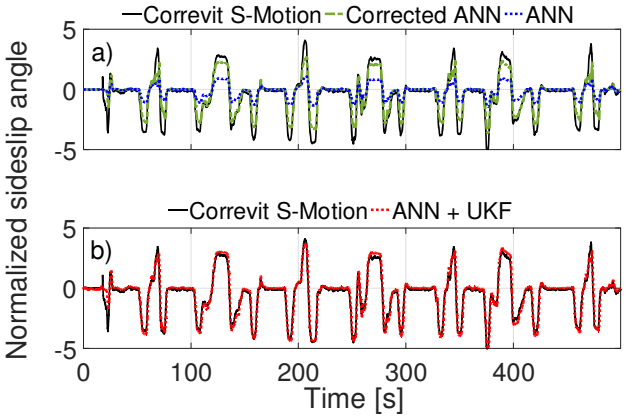

Figure 13. Sideslip angle estimation. a) ANN and Corrected ANN estimation. b) ANN + UKF estimation

The result of the integrated ANN and UKF seen in figure 13 show how the sideslip angle estimation is properly corrected with the proposed method. Both the time response and the magnitude of the estimated sideslip angle are similar to the measured one. Note that in the first sideslip angle variation, the sideslip angle measurement given by the Correvit sensor is not coherent with the IMU measurements (i.e. physics of the problem). The big discontinuity in the measured sideslip angle plot leads to conclude that the Correvit measurement is wrong for time window between 20s and 25s. Excluding this, the proposed observer is very efficient in the estimation. This is confirmed by an absolute error time analysis of the observer which is shown in figure 14 and the quantitative analysis shown in table 5. In the figure, the real value versus the estimated value is shown on the left while on the right the sideslip angle error PDF is shown. In the table, the MAE, RMSE and R values are reported. Additionally to the quantitative analysis of the longitudinal velocity observer, the mean value of the error μ and standard deviation σ were calculated. These were calculated since the error distribution resembles a normal distribution. The corrected ANN, which depends on the output of the UKF (19-20), also improves the estimation given by the ANN. The saturation in the pseudo-sideslip angle correction assures that the observer's output remains stable. Since $\hat{\beta}^{\star}$ and $\tilde{\beta}$ recursively depend on each other, the correction helps the convergence of the algorithm in terms of speed and accuracy.

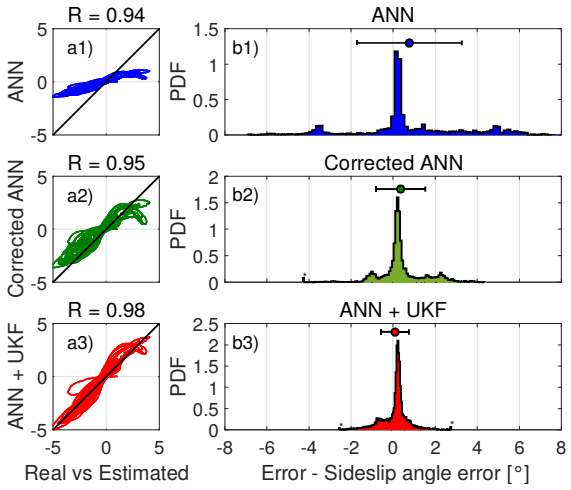

Figure 14. Sideslip angle estimation error. a1-3) Real value vs estimated value. b1-3) Probability density function of the estimation error

Table 5. Sideslip angle estimation error.

Observer	MAE [°]	RMSE [°]	R	μ[°]	σ[°]
ANN	0.77	2.63	0.94	0.77	2.50
Corrected ANN	0.36	1.34	0.95	0.36	1.17
ANN + UKF	0.09	0.70	0.98	0.09	0.67

For small values of the sideslip angle, the ANN estimates very well, however, for larger values the estimation is incorrect. The proposed algorithm properly corrects the ANN estimate. Since for many seconds the vehicle is in a straight line, the sideslip angle is close to zero, thus, the correlation factor and mean absolute error have very good values for all observers. However, the root mean square error of the ANN is very high. The proposed ANN + UKF shows instead very good values for all metrics with an improvement of 73.3% on the RMSE respect to the ANN and of 47.7% respect to the corrected ANN. Also the low mean error and standard deviation show how the proposed observer estimates the sideslip angle with very high precision.

Conclusion

In this paper an integrated ANN and UKF method to estimate the sideslip angle is proposed and tested on real data. The estimation is given by using only measurements coming from an IMU. The ANN is trained by means of numerical data only. An algorithm has been proposed to select the best neural network structure. Experimental results show that the algorithm is effective and that the deployed ANN can correctly predict the sideslip angle time response of a real vehicle despite not being able to correctly estimate its magnitude. This is mainly due to different closed-loop behaviour and model uncertainty. By means of an novel integrated ANN and UKF observer based on a kinematic model, the sideslip angle estimation is corrected. The kinematic model uses a longitudinal velocity observer based on direct integration with integral damping and integral value reset. The proposed method allows the sideslip angle estimation to be transparent to vehicle and tire models due to the use of a kinematic vehicle model rather than a dynamic one. An additional strategy for the correction of the pseudo-sideslip angle improves the accuracy and the convergence of the observer. The estimated value given by the proposed method shows very good results on experimental data.

Acknowledgements

The authors wish to thank Meccanica 42 and Danisi Engineering, industrial partners of this research activity who have provided a fundamental technical and financial support.

Declaration of conflicting interests

The authors declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

References

1. Milliken WF, Milliken DL. Race car vehicle dynamics. Warrendale, PA, U.S.A: SAE International; 1995.
2. Edelmann J, Plchl M. Handling characteristics and stability of the steady-state powerslide motion of an automobile. Regular and Chaotic Dynamics. 2009 Dec;14(6):682–692. Available from: http://link.springer.com/10.1134/S1560354709060069.
3. Voser C, Hindiyeh RY, Gerdes JC. Analysis and control of high sideslip manoeuvres. Vehicle System Dynamics. 2010 Dec;48(sup1):317–336. Available from: http://www.tandfonline.com/doi/abs/10.1080/00423111003746140.
4. van Zanten AT, Erhardt R, Pfaff G. VDC, The Vehicle Dynamics Control System of Bosch; 1995. Available from: http://papers.sae.org/950759/.
5. Van Zanten AT. Bosch ESP Systems: 5 Years of Experience; 2000. Available from: http://papers.sae.org/2000-01-1633/.
6. Hu C, Wang R, Yan F, Chen N. Should the Desired Heading in Path Following of Autonomous Vehicles be the Tangent Direction of the Desired Path? IEEE Transactions on Intelligent Transportation Systems. 2015;p. 1–11. Available from: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=7120090.
7. Wang W, Yuan L, Tao S, Zhang W, Su T. Estimation of vehicle side slip angle in nonlinear condition based on the state feedback observer. In: Automation and Logistics (ICAL), 2010 IEEE International Conference on. IEEE; 2010. p. 632–636. Available from: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5585359.
8. Ding N, Taheri S. A Modified Dugoff Tire Model for Combined-slip Forces. Tire Science and Technology. 2010 Sep;38(3):228–244. Available from: http://tiresciencetechnology.org/doi/abs/10.2346/1.3481696.
9. Grip HaFe, Imsland L, Johansen TA, Kalkkuhl JC, Suissa A. Vehicle sideslip estimation. Control Systems, IEEE. 2009;29(5):36–52. Available from: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5256356.
10. Shraim H, Ouladsine M, Fridman L, Romero M. Vehicle parameter estimation and stability enhancement using sliding modes techniques. International journal of vehicle design. 2008;48(3-4):230–254. Available from: http://www.inderscienceonline.com/doi/abs/10.1504/IJVD.2008.022578.
11. Haiyan Zhao, Hong Chen. Estimation of Vehicle Yaw Rate and Side Slip Angle using Moving Horizon Strategy. IEEE; 2006. p. 1828–1832. Available from: http://ieeexplore.ieee.org/document/1712670/.
12. Zhang H, Huang X, Wang J, Karimi HR. Robust energy-to-peak sideslip angle estimation with applications to ground vehicles. Mechatronics. 2015 Sep;30:338–347. Available from: http://linkinghub.elsevier.com/retrieve/pii/S095741581400124X.
13. You SH, Hahn JO, Lee H. New adaptive approaches to real-time estimation of vehicle sideslip angle. Control Engineering Practice. 2009 Dec;17(12):1367–1379. Available from: http://linkinghub.elsevier.com/retrieve/pii/S0967066109001361.
14. Ryu J, Rossetter EJ, Gerdes JC. Vehicle sideslip and roll parameter estimation using GPS. In: Proceedings of the AVEC International Symposium on Advanced Vehicle Control; 2002. .
15. Doumiati M, Victorino A, Charara A, Lechner D. A method to estimate the lateral tire force and the sideslip angle of a vehicle: Experimental validation. In: American Control Conference (ACC), 2010. IEEE; 2010. p. 6936–6942. Available from: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5531319.
16. Huang Y, Bao C, Wu J, Ma Y. Estimation of Sideslip Angle Based on Extended Kalman Filter. Journal of Electrical and Computer Engineering. 2017;2017:1–9. Available from: https://www.hindawi.com/journals/jece/2017/5301602/.
17. Li L, Jia G, Ran X, Song J, Wu K. A variable structure extended Kalman filter for vehicle sideslip angle estimation on a low friction road. Vehicle System Dynamics. 2014 Feb;52(2):280–308. Available from: http://www.tandfonline.com/doi/abs/10.1080/00423114.2013.877148.
18. Pacejka HB. Tire and vehicle dynamics. No. 342 in SAE-R. Warrendale, Pa: Society of Automotive Engineers; 2002. OCLC: 249480435.
19. Baffet G, Charara A, Lechner D. Estimation of vehicle sideslip, tire force and wheel cornering stiffness.

Control Engineering Practice. 2009 Nov;17(11):1255–1264. Available from: http://linkinghub.elsevier.com/retrieve/pii/S0967066109001129.

20. Lian YF, Zhao Y, Hu LL, Tian YT. Cornering stiffness and sideslip angle estimation based on simplified lateral dynamic models for four-in-wheel-motor-driven electric vehicles with lateral tire force information. International Journal of Automotive Technology. 2015 Aug;16(4):669–683. Available from: http://link.springer.com/10.1007/s12239-015-0068-4.
21. Julier SJ, Uhlmann JK. A new extension of the Kalman filter to nonlinear systems. In: Int. symp. aerospace/defense sensing, simul. and controls. vol. 3. Orlando, FL; 1997. p. 182–193.
22. Wan EA, Van Der Merwe R. The unscented Kalman filter for nonlinear estimation. In: Adaptive Systems for Signal Processing, Communications, and Control Symposium 2000. AS-SPCC. The IEEE 2000. Ieee; 2000. p. 153–158.
23. Morrison G, Cebon D. Sideslip estimation for articulated heavy vehicles at the limits of adhesion *. Vehicle System Dynamics. 2016 Sep;p. 1–28. Available from: https://www.tandfonline.com/doi/full/10.1080/00423114.2016.1223326.
24. Antonov S, Fehn A, Kugi A. Unscented Kalman filter for vehicle state estimation. Vehicle System Dynamics. 2011 Sep;49(9):1497–1520. Available from: http://www.tandfonline.com/doi/abs/10.1080/00423114.2010.527994.
25. Jia G, Yang C, Li L, Ran X, Song J, Chen J. UKF-based adaptive variable structure observer for vehicle sideslip with dynamic correction. IET Control Theory & Applications. 2016 Sep;10(14):1641–1652. Available from: http://digital-library.theiet.org/content/journals/10.1049/iet-cta.2015.1030.
26. Cybenko G. Approximation by superpositions of a sigmoidal function. Mathematics of Control, Signals, and Systems (MCSS). 1989;2(4):303–314.
27. Du X, Sun H, Qian K, Li Y, Lu L. A prediction model for vehicle sideslip angle based on neural network. In: Information and Financial Engineering (ICIFE), 2010 2nd IEEE International Conference on. IEEE; 2010. p. 451–455. Available from: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5609398.
28. Melzi S, Sabbioni E. On the vehicle sideslip angle estimation through neural networks: Numerical and experimental results. Mechanical Systems and Signal Processing. 2011 Aug;25(6):2005–2019. Available from: http://linkinghub.elsevier.com/retrieve/pii/S0888327010003341.
29. Sasaki H, Nishimaki T. A side-slip angle estimation using neural network for a wheeled vehicle. SAE Technical Paper; 2000.
30. Wei W, Shaoyi B, Lanchun Z, Kai Z, Yongzhi W, Weixing H. Vehicle Sideslip Angle Estimation Based on General Regression Neural Network. Mathematical Problems in Engineering. 2016;2016:1–7. Available from: http://www.hindawi.com/journals/mpe/2016/3107910/.
31. Boada BL, Boada MJL, Gaucha A, Olmeda E, Daz V. Sideslip angle estimator based on ANFIS for vehicle handling and stability. Journal of Mechanical Science and Technology. 2015 Apr;29(4):1473–1481. Available from: http://link.springer.com/10.1007/s12206-015-0320-x.
32. Acosta M, Kanarachos S, Fitzpatrick ME. Robust Virtual Sensing for Vehicle Agile Manoeuvring: A Tyre-model-less Approach. IEEE Transactions on Vehicular Technology. 2017;p. 1–1. Available from: http://ieeexplore.ieee.org/document/8089352/.
33. Boada BL, Boada MJL, Diaz V. Vehicle sideslip angle measurement based on sensor data fusion using an integrated ANFIS and an Unscented Kalman Filter algorithm. Mechanical Systems and Signal Processing. 2016 May;72-73:832–845. Available from: http://linkinghub.elsevier.com/retrieve/pii/S0888327015004999.
34. Fukada Y. Slip-Angle Estimation for Vehicle Stability Control. Vehicle System Dynamics. 1999 Nov;32(4-5):375–388. Available from: http://www.tandfonline.com/doi/abs/10.1076/vesd.32.4.375.2079.
35. Hagan MT, Demuth HB, Beale MH, De Jess O. Neural network design. S. l.: s. n.; 2016. OCLC: 958124711.
36. Hochreiter S, Schmidhuber J. Long short-term memory. Neural computation. 1997;9(8):1735–1780. Available from: http://www.mitpressjournals.org/doi/abs/10.1162/neco.1997.9.8.1735.
37. Cho K, Van Merrinboer B, Gulcehre C, Bahdanau D, Bougares F, Schwenk H, et al. Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:14061078. 2014;Available from: https://arxiv.org/abs/1406.1078.
38. Guiggiani M. The Science of Vehicle Dynamics. Dordrecht: Springer Netherlands; 2014. DOI: 10.1007/978-94-017-8533-4. Available from: http://link.springer.com/10.1007/978-94-017-8533-4.
39. Julier SJ, Uhlmann JK, Durrant-Whyte HF. A new approach for filtering nonlinear systems. In: American Control Conference, Proceedings of the 1995. vol. 3. IEEE; 1995. p. 1628–1632.