sensors

MDPI

Article

Fault Detection and Isolation via the Interacting Multiple Model Approach Applied to Drive-By-Wire Vehicles

Vincent Judalet [1,*,†,‡], Sébastien Glaser [2,‡], Dominique Gruyer [1,‡] and Saïd Mammar [3,‡]

[1] LIVIC Laboratory, IFSTTAR, 25 Allée des Marronniers, 78000 Versailles, France; dominique.gruyer@ifsttar.fr
[2] Centre for Accident Research and Road Safety (CARRS-Q), Queensland University of Technology (QUT), Brisbane City, QLD 4000, Australia; sebastien.glaser@qut.edu.au
[3] IBISC Laboratory, University of Evry Val d'Essonne, 40 rue du Pelvoux, 91020 Evry, France; said.mammar@ibisc.univ-evry.fr
* Correspondence: vincent.judalet@estaca.fr; Tel.: +33-(0)176-52-12-04
† Current address: École Supérieure des Techniques Aéronautiques et de Construction Automobile (ESTACA), 12 Avenue Paul Delouvrier, 78180 Montigny-le-Bretonneux, France.
‡ All authors contributed equally to this work.

Received: 15 May 2018; Accepted: 16 July 2018; Published: 18 July 2018

Abstract: The place of driving assistance systems is currently increasing drastically for road vehicles. Paving the road to the fully autonomous vehicle, the drive-by-wire technology could improve the potential of the vehicle control. The implementation of these new embedded systems is still limited, mainly for reliability reasons, thus requiring the development of diagnostic mechanisms. In this paper, we investigate the detection and the identification of sensor and actuator faults for a drive-by-wire road vehicle. An Interacting Multiple Model approach is proposed, based on a non-linear vehicle dynamics observer. The adequacy of different probabilistic observers is discussed. The results, based on experimental vehicle signals, show a fast and robust identification of sensor faults while the actuator faults are more challenging.

Keywords: Interacting Multiple Model; fault detection and isolation; drive-by-wire; extended Kalman filter

1. Introduction

To enable advanced driver-assistance systems (ADAS) to operate more efficiently without disrupting the driver, decoupled control systems, known as drive-by-wire technologies, have been developed for the control of the steering or braking systems.

The drive-by-wire technology (DBW) consists in introducing electronic control systems between the driver interfaces (steering wheel and pedals) and the vehicle actuators (motor, brakes, and steering rack) in substitution for the conventional mechanical or hydraulic physical connections. According to the driver actions on both the steering wheel and the pedals, these systems automatically adapt and control the actuators to produce the desired vehicle behaviour.

The DBW systems present many advantages when compared to conventional systems, in terms of safety (via the active steering), comfort (configurable haptic feedback), vehicle design and costs (see [1]). However, by-wire systems pose significant challenges to manufacturers for their implementation. On the one hand, electrical systems are subject to failure in a less predictable and more frequent manner than mechanical systems. While the throttle-by-wire (or electronic throttle control) was successfully introduced to the market at the end of the 1990s, car manufacturers have faced many reliability and safety issues during the implementation in the market of other DBW systems, such as brake-by-wire (BBW) and steer-by-wire (SBW). For example, the first introduction of a SBW system by the manufacturer Infinity on the Q50 model led to a recall in 2013. This explains why the development of the DBW technology is still very limited.

In the aeronautic domain, the development of the fly-by-wire technology has been made possible by the use of the redundancy of sensors and actuators. This solution is however not compatible with cost-effectiveness constraints inherent to vehicles mass production, and the systems developed are generally limited to high-end models.

For a road vehicle, the steering and braking systems are independent, however they both take part in the control of the vehicle trajectory. We propose to take advantage of this diversity of control strategies to maintain a controllability (even though reduced) of the vehicle if a default occurs on one of these systems, as in [2]. For instance, if a fault occurs in the braking system, asymmetrical braking forces will disrupt the vehicle trajectory. In this case, the steering controller could automatically reject this perturbation. Similarly, if the steering system is faulty, differential braking could help to maintain the desired trajectory.

Such a reconfigurable control mechanism requires the real time diagnosis of a fault before its effect leads the vehicle to an uncontrollable state. As this fault may concern either the SBW or the BBW systems, both systems have to be supervised. Published research in the field of fault detection and isolation (FDI) applied to drive-by-wire vehicles treats separately the case of SBW [3–10] or BBW [3,11,12] system faults.

In this paper, we investigate a fast and reliable FDI scheme applied to a drive-by-wire vehicle. Both SBW and BBW systems fault are processed in a unique fault detection algorithm. To cope with the diversity of possible faults, the FDI algorithm relies on an Interacting Multiple Model (IMM) approach.

The considered DBW vehicle architecture is composed of different components:

- The human–machine interfaces (HMI) are typically the steering wheel and the pedals (brake and throttle). They collect the driver's input and provide him with haptic feedback (wheel alignment torque for instance) and eventual incentive information.
- The sensor set corresponds to a standard vehicle equipped with an ESC (electronic stability control) system: wheel speed sensors, an inertial navigation system (INS), including accelerometers and gyrometers, a steering wheel angle sensor (SAS), pedals stroke and braking pressure sensors. Moreover, a sensor for the steering angle of the front wheels is required as it is subjected to differ from the steering wheel angle.
- The actuators include the brakes (electronic or electro-hydraulic) and the electronic steering system.
- The control unit supervises the system, i.e., estimates the vehicle dynamics, identifies eventual faults, and controls the actuators consequently.

Faults are subjected to occur on each of these systems. However, the faults occurring on the electronic control unit or on the HMI sensors (pedal stroke and SAS) are generally difficult to process without physical redundancies [13,14]. In this paper, we focus on the detection of abrupt partial and total faults on the following sensors and actuators:

- Sensor faults: X- and Y-accelerations, vehicle yaw rate, wheel turn rates and the steering angle of front wheels.
- Actuator faults: Steering actuator and brakes.

While the detection of multiple faults is compatible with the IMM approach, this subject is not addressed in this study. We consider the subsystems (sensors and actuators) are independent, so a fault is unlikely to affect two subsystems, and that a single failure will be detected before the occurrence of a second fault.

Many FDI approaches have been proposed in the literature (see [15–19]). On the one hand, data-based methods are based on the knowledge of a priori observation of the system. Quantitative approaches use artificial intelligence algorithms (neural networks and Bayesian frameworks) or the statistical processing of the data, while qualitative methods generally consist of expert systems. These approaches do not require any analytical or structural knowledge on the physical system.

On the other hand, model-based methods involve the physical knowledge of the monitored system. This knowledge leads to one or many models (analytical or graphical), which describes the system operating in normal state or under a degraded mode (for instance, see [20–23]).

Qualitative methods rely on the modelling of the input–output relationships in terms of quantitative functions. These methods are relatively easy to implement because they do not require information of the internal physical relations of the system, but are limited to commutative systems.

Quantitative model-based methods are based on the analytical modelling of the relations between the input and the output of the system, generally via a state analysis.

In the field of automotive control, the observation of the vehicle dynamics has been intensively investigated, and several evolution models are available. This explains why the literature on FDI schemes dedicated to SBW or BBW systems mainly rely on model-based methods [4–8,10,22,24].

Model-based FDI generally implies the generation of fault indicators, called residuals, obtained by comparing the predicted state of the system to the actual observed state. If the system is in the normal state, the residuals remain close to zero, else a fault is detected. The classification of the residuals eventually leads to the fault identification.

When an abrupt fault occurs on a system, its behavior may be radically altered and compromised. Thus, it is difficult to model this system both in the normal state and in the different degraded modes with a unique model. Multiple model (MM) approaches have been developed to cope with this issue (see [25,26]). They consist in implementing different observers in parallel; each observer corresponds to a specific system state (normal or degraded).

Originally, the observers were running independently, without any mutual interaction. This method is adequate when the model structure is not or slightly modified (for instance, parameter estimation problems), but if the models differ radically (which is generally the case in a FDI scheme), the transition from one state to another may be difficult to evaluate, thus jeopardizing the fault detection or increasing the detection time.

The Interacting Multiple Model (IMM) approach has been introduced to solve this problem (see [27–31]). The different system states are modelled in a Markov chain. Before each algorithm iteration, the estimates for each mode are updated according to the transition probabilities from one mode to another.

In the case of road vehicles, precise analytical models are already available in the literature. They are mainly based on the modelling of the forces between the tyre and the road to estimate the vehicle dynamics.

For this study, we implemented an IMM-based FDI approach to investigate the detection and the isolation of faults occurring on a DBW vehicle.

The paper is organized as follows: Section 2 introduces the IMM algorithm and its implementation for a FDI purpose. Section 3 presents the probabilistic observers, which have been considered for the IMM implementation. The non-linear vehicle model is detailed in Section 4, and finally the experimental validation tests are shown in Section 5.

2. The IMM Approach for the Fault Detection and Isolation

2.1. Principles of the IMM Estimation

In an IMM scheme, each system state is modelled in a separate model. To enable the interactions between the different modes, the transition probabilities from one mode to the others are considered.

The interaction implies to correct the state estimates $\hat{x}_i$ for different modes towards the most likely estimate according to the transition probabilities π_{ij} defined by:

$$\pi_{ij} = p\{m_{j,k+1}|m_{i,k}\}, \qquad \forall [m_i, m_j] \in S^2 \tag{1}$$

and

$$\sum_{j=1}^{s} \pi_{ij} = 1, \qquad \forall i \in [1, s] \tag{2}$$

where $S = \{m_1, m_2, \ldots, m_s\}$ is the set of system modes, π_{ij} is the transition probability from mode m_i to mode m_j, and $m_{i,k}$ denotes that the system is in mode m_i at time k.

The IMM estimator is a recursive algorithm. For each cycle, four steps are carried out:

1. First, the different estimates and their covariance matrices are mixed, according to the probabilities of activation μ_i of the model at the current time k. The predicted mode probability μ_j, defined by:

$$\mu_{j,k+1|k} = \sum_{i} \pi_{ij}\mu_{i,k} \tag{3}$$

enables computing the mixed probability $\mu_{i|j,k}$:

$$\mu_{i|j,k} = \frac{\pi_{ij}\mu_{i,k}}{\mu_{j,k+1|k}} \tag{4}$$

The mixed estimates $\hat{x}^0_{j,k-1}$ are then defined by:

$$\hat{x}^0_{j,k} = \sum_{i=1}^{s} \mu_{i|j,k}\hat{x}_{i,k} \tag{5}$$

and their covariance matrices $P^0_{j,k-1}$ by:

$$P^0_{j,k} = \sum_{i=1}^{s} \left[P_{i,k} + \Delta\hat{x}_{i|j}\right] \mu_{i|j,k} \tag{6}$$

where $\Delta\hat{x}_{i|j}$ is the model uncertainty:

$$\Delta\hat{x}_{i|j} = \left(\hat{x}^0_{j,k} - \hat{x}_{i,k}\right)\left(\hat{x}^0_{j,k} - \hat{x}_{i,k}\right)^t \tag{7}$$

2. Second, a probabilistic filter is performed for each mode in parallel to obtain the updated estimates $\hat{x}_{j,k+1}$ and their covariance $P_{j,k+1}$. Starting from each mixed estimate $\hat{x}^0_{j,k}$, an evolution model is used to predict the next state vector, and then this prediction is corrected according to the last measured data. In Section 3, different probabilistic filters are investigated: the extended Kalman filter (EKF), the unscented Kalman filter (UKF), and the (first-order) divided differences filter (DD1).
3. The activation probability update is done with the computation of the likelihood L_j:

$$L_{j,k} = \frac{1}{(2\pi)^{d/2}\sqrt{\det \Lambda_{j,k}}} \exp\left[-\frac{1}{2}\nu^{\mathrm{T}}_{j,k}\Lambda^{-1}_{j,k}\nu_{j,k}\right] \tag{8}$$

where $\Lambda_{j,k}$ is the residual covariance matrix, as defined in Equation (17), and d is the dimension of the residual vector $\nu_{j,k}$. The updated probability for the mode j is obtained from the current likelihood and the past activation probability:

$$\mu_{j,k+1} = \frac{\mu_{j,k+1|k} L_{j,k+1}}{\sum_i \mu_{i,k+1|k} L_{i,k+1}} \tag{9}$$

4. Finally, the overall estimate $\hat{x}_{k+1}$ and an overall covariance matrix P_{k+1} are computed by weighting the different estimates by their respective probabilities. This step is optional and is required only if an overall estimate is needed, for instance for control purpose.

$$\hat{x}_{k+1} = \sum_j \mu_{j,k+1} \hat{x}_{j,k+1} \tag{10}$$

$$P_{k+1} = \sum_j \mu_{j,k+1} \left[P_{j,k+1} + \Delta\hat{x}_j \Delta\hat{x}_j^t \right] \tag{11}$$

where

$$\Delta\hat{x}_j = \hat{x}_{j,k+1} - \hat{x}_{k+1} \tag{12}$$

2.2. Dedicated Implementation for FDI Purpose

The first mode m_1, called *nominal* mode (NOM), corresponds to the normal (i.e., fault-free) state of the system. Next to the nominal mode, an additional *faulty* mode is implemented for each of the investigated faults.

A typical SBW vehicle is equipped with four wheel turn rate sensors, an inertial sensor (x and y accelerations and the vehicle yaw rate) and a steering angle sensor at the front wheels. Consequently, for the sensor faults detection, eight additional modes are proposed; their names are preceded by an "S" (for sensor):

- Four modes ($m_{2,3,4,5}$) (called S Ω_{FR}, S Ω_{FL}, S Ω_{RL} and S Ω_{RR}) correspond to a fault on the wheel turn rate signals (respectively at the front right, front left, rear left and rear right wheels.
- Three modes ($m_{6,7,8}$) are for the inertial sensor faults (S A_X and S A_Y for the longitudinal and lateral accelerations, and S YR for the yaw rate).
- One mode (m_9) is for a fault on the front steering angle signal (S δ).

Likewise, five modes are introduced for the actuator fault detection (preceded by an "A" for actuator):

- One mode (m_{10}) is for a fault on the steering angle actuator (A δ).
- Four modes ($m_{11,12,13,14}$) (called A Br_{FR}, A Br_{FL}, A Br_{RL} and A Br_{RR}) correspond to a fault on the braking system (respectively, at the front right, front left, rear left and rear right wheels).

In total, 14 modes have been considered.

No hypothesis is made on the presence of a fault at the start of the algorithm. Therefore, the different activation probabilities are initialized to the same value $\mu_0 = 1/n$ (where n is the number of modes).

In the fault-free state, the likelihood of the nominal mode should largely exceed those of the other (faulty) modes, thus the probability μ_1 should rapidly rise to 100%.

When a fault occurs, the likelihood of all modes should be affected, except the mode corresponding to the affected sensor. The probability μ_1 should then drop to 0 and the probability of the faulty mode should rise to 100%.

The detection and the isolation of the faults can consequently be carried out at the same time by comparing the probabilities of activation to a threshold μ_T.

$$\mu_j = \max_i \mu_i \begin{cases} \geq \mu_T & \Rightarrow & H_j \\ < \mu_T & \Rightarrow & H_0 \end{cases} \tag{13}$$

where $H_{j(j>0)}$ means that the ith state has been identified. H_0 means that the current state cannot be identified. The threshold μ_T must be above 50% to prevent that the activation probabilities of two different modes exceed the threshold at the same time (as the sum of the activation probabilities is 100%). The value of the threshold can empirically be selected strictly between 50% and 100% by doing a compromise between time to detection and robustness to false detection. A low value makes the fault detection faster, while a higher value increases the robustness to false positives.

In this study, μ_T was empirically set to 95%.

2.3. Transition Probabilities

The transition matrix π_{ij} indicates the probability for switching from a previous mode i to another mode j. These transition probabilities are implemented in the first IMM step to mix the previous estimates. For each line, the sum of the terms is equal to 1. The first line represents the transition from the nominal mode to another, and the diagonal terms indicate the probability that the system remains in the same mode between two algorithm iterations.

The terms of this matrix can be chosen to reach the best compromise between a rapid time to detection and a good robustness to false positive. Two extreme example can be highlighted:

- If π_{ij} is set equal to the identity matrix, we show easily from Equations (4) and (5) that the mixed probabilities and the mixed estimates remains equal to the previously estimated ones. This results in skipping the first step of the IMM algorithm. In this case, the IMM behaves exactly like a standard multiple model approach, with the risk of late detection explained the introduction.
- If each term is set equal to $1/n$ (where n is the number of modes), then, in Step 1, the mixed estimates for all modes becomes equal to the previous overall estimate, and the mixed probabilities of activation becomes equal to $1/n$. In this case, the previous iterations are totally forgotten, and the fault detection only relies on the current iteration. Thus, increasing the risk of false detection.

The transition matrix is generally empirically tuned between these two extreme cases. The algorithm becomes more robust to false positive if the transition matrix is close to the identity matrix. The fault detection can be made faster by increasing the non-diagonal terms. For the experimental part of this study, we adopted the following transition matrix:

$$(\pi_{ij}) = \begin{pmatrix} 0.5 & \frac{0.5}{n-1} & \dots & \dots & \frac{0.5}{n-1} \\ 0.1 & 0.9 & 0 & \dots & 0 \\ \vdots & 0 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ 0.1 & 0 & \dots & 0 & 0.9 \end{pmatrix}$$

2.4. Immunization to Faults

To make the modes insensitive to a defined fault, the corresponding model has to be modified, compared to the nominal model.

For sensor fault, different immunization methods have been proposed:

- For a total default, the easiest way is to cancel the corresponding line in the measurement matrix H.

- For a partial default, the corresponding parameter can be increased in the measurement noise covariance matrix R.

We opted for the second solution to detect partial faults. Figure 1 displays the range of the sensor data. In the nominal state, the sensor data is close to the actual measured value; the error variance can be set according to the nominal sensor noise standard deviation σ: $\mathrm{Var}_{\mathrm{NOM}} = \sigma^2$.

To set the variance $\mathrm{Var}_{\mathrm{fault}}$ for the immunized mode, we have to define minimal and maximal values, above which the measurement are incoherent and can be easily rejected. For instance, the steering angle is not supposed to exceed the maximal actuator amplitude. For the acceleration, the typical maximal value is ± 15 m/s^2.

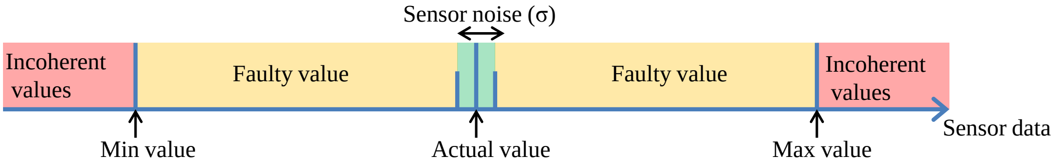

Figure 1. Sensor fault range.

The faulty mode standard deviation $\sigma_{\max}$ is tuned to cover the whole coherent range. We get: $\mathrm{Var}_{\mathrm{fault}} = \sigma^2_{\max}$.

A similar method has been adopted for the actuator faults. The parameter of the model noise covariance Q is increased for the line of the evolution function which is affected by the corresponding actuator.

3. Probabilistic Vehicle State Observer

As explained in the previous section, the IMM algorithm compares the likelihood of different state estimations to identify system faults. The computation of the likelihoods requires an estimation of both the system state and the error on the prediction. This is typically provided by the probabilistic state observers such as the Kalman filters family.

The Kalman filter is a well known method for the state estimation of a linear system. It is based on a linear evolution model of the system, and provides both the estimate of the state vector and the covariance information about the error of this estimate.

This filter is considered recursive, as it only takes into account the previous state estimate and the current measurement.

It consists of a two-step process. In the *prediction* step, the previous state estimate is updated according to the evolution function. The linearity properties of the evolution function enable to directly compute the predicted error covariance from the previous one.

Then, the *update* step corrects the predicted state according to the current measurement. The correction gain depends on the relation between the estimation and the measurement error covariance.

For non-linear system, however, the evolution function cannot be applied to the covariance directly. To cope with non-linear system, different variants have been proposed, such as the extended Kalman filter [32], the unscented Kalman filter [33] and the divided differences filters [34]. This section introduces these filters.

3.1. The Extended Kalman Filter

To cope with non-linearity of the evolution function, the EKF linearises the function in the vicinity of the current state. The derivation, based on the first order Taylor development, involves the computation of the partial derivatives matrix (the Jacobian) of the evolution function. It consists of a two step process.

1. During the *prediction* step, the predicted state estimate $\hat{x}(k+1|k)$ and its predicted covariance $P(k+1|k)$ are estimated following the non-linear evolution function f and its Jacobian matrix F, according to Equations (14) and (15).

$$\hat{x}_{k+1|k} = f\left(\hat{x}_k, u_k\right) \tag{14}$$

$$P_{k+1|k} = F P_k F^t + Q \tag{15}$$

2. In the *update* stage, the predicted estimate is corrected according to the updated output vector y_{k+1}. The residual ν and its covariance Λ are evaluated according to the measurement matrix H.

$$\nu_{k+1} = y_{k+1} - H\hat{x}_{k+1|k} \tag{16}$$

$$\Lambda_{k+1} = H P_{k+1|k} H^{\mathrm{T}} + R \tag{17}$$

The filter gain K can then be calculated according to Equation (18).

$$K_{k+1} = P_{k+1|k} H^{\mathrm{T}} \Lambda_{k+1}^{-1} \tag{18}$$

Finally, the updated state estimate $\hat{x}_k$ and the updated covariance P_k can be computed following Equations (19) and (20).

$$\hat{x}_{k+1} = \hat{x}_{k+1|k} + K_{k+1}\nu_{k+1} \tag{19}$$

$$P_{k+1} = P_{k+1|k} - K\Lambda K_{k+1}^{\mathrm{T}} \tag{20}$$

Q and R are, respectively, the model and the measurement noise covariance matrices.

3.2. The First-Order Divided Differences Filter

The DD1 filter locally linearises the system and measurement dynamics via first-order divided differences, rather than the first order Taylor development used in the EKF. The computation of the Jacobian matrices is also no longer needed.

First, we have to define the Cholesky (or square root) decomposition of the predicted state covariance $P_{k+1|k}$, the updated state covariance P_k, the process noise covariance Q and the measurement noise covariance R:

$$\begin{array}{lclclcl} Q & = & S_v S_v^{\mathrm{T}} & \qquad & R & = & S_w S_w^{\mathrm{T}} \\ P_{k+1|k} & = & S_{x_{k+1|k}} S_{x_{k+1|k}}^{\mathrm{T}} & \qquad & P_k & = & S_{x_k} S_{x_k}^{\mathrm{T}} \end{array} \tag{21}$$

The first-order divided differences matrices are defined as:

$$\begin{aligned} S_{x\hat{x}}(i,j) &= \frac{f_i(\hat{x}_k+\xi S_{x_k,j},v_k)-f_i(\hat{x}_k-\xi S_{x_k,j},v_k)}{2\xi} \\ S_{xv}(i,j) &= \frac{f_i(\hat{x}_k,v_k+\xi S_{v,j})-f_i(\hat{x}_k,v_k-\xi S_{v,j})}{2\xi} \\ S_{y\hat{x}}(i,j) &= \frac{g_i(\hat{x}_k+\xi S_{x_{k+1|k},j},n_k)-g_i(\hat{x}_k-\xi S_{x_{k+1|k},j},n_k)}{2\xi} \\ S_{yv}(i,j) &= \frac{g_i(\hat{x}_k,n_k+\xi S_{v,j})-g_i(\hat{x}_k,n_k-\xi S_{v,j})}{2\xi} \end{aligned} \tag{22}$$

where ξ is the disrupting parameter.

The two algorithm steps become:

1. In the *prediction* step, the predicted state vector $\hat{x}_{k+1|k}$ and its covariance $P_{k+1|k}$ are computed according to:

$$\hat{x}_{k+1|k} = f\left(\hat{x}_k, u_k\right) \tag{23}$$

$$P_{k+1|k} = (S_{x\hat{x}})(S_{x\hat{x}})^{\mathrm{T}} + (S_{xv})(S_{xv})^{\mathrm{T}} \tag{24}$$

The state covariance prediction can be factored using the square root decomposition $P_{k+1|k} = S_{x_{k+1|k}} S^{\mathrm{T}}_{x_{k+1|k}}$ to yield:

$$S_{x_{k+1|k}} = \mathcal{H}\left(\left[\begin{array}{cc} S_{x\hat{x}} & S_{xv} \end{array}\right]\right) \tag{25}$$

where $\mathcal{H}\,(.)$ is the Householder transformation.

2. In the *update* step, the predicted state will be corrected according to the new measured data y. The residual ν_{k+1} and its predicted covariance Λ_{k+1} are defined by:

$$\nu_{k+1} = y_{k+1} - H\hat{x}_{k+1|k} \tag{26}$$

$$\Lambda_{k+1} = (S_{y\hat{x}})(S_{y\hat{x}})^{\mathrm{T}} + (S_{yv})(S_{yv})^{\mathrm{T}} \tag{27}$$

Λ can be decomposed into $\Lambda_{k+1} = S_y S_y^{\mathrm{T}}$ to yield:

$$S_y = \mathcal{H}\left(\left[\begin{array}{cc} S_{y\hat{x}} & S_{yv} \end{array}\right]\right) \tag{28}$$

The DD1 gain matrix K which minimizes the trace of P_{k+1} is defined by:

$$K_{k+1} = S_{x_{k+1|k}} S_{y\hat{x}} \Lambda^{-1}_{k+1} \tag{29}$$

Finally, the updated state vector $\hat{x}_{k+1}$ and its error covariance matrix P_{k+1} are calculated with:

$$\hat{x}_{k+1} = \hat{x}_{k+1|k} + K_{k+1}\nu_{k+1} \tag{30}$$

$$P_{k+1} = P_{k+1|k} - K_{k+1}\Lambda_{k+1}K^{\mathrm{T}}_{k+1} \tag{31}$$

Here, again, the error covariance matrix can be decomposed into $P_{k+1} = \hat{S}_x \hat{S}_x^{\mathrm{T}}$ with:

$$S_{x_{k+1}} = \mathcal{H}\left(\left[\begin{array}{cc} S_{x_{k+1|k}} - K_{k+1}S_{y\hat{x}} & K_{k+1}S_{yv} \end{array}\right]\right) \tag{32}$$

3.3. The Unscented Kalman Filter

The UKF is a derivative-free filter based on a constellation of $2n + 1$ points (the sigma points) in the vicinity of the current state vector (where n is the length of the state vector).

Instead of derivating the non-linear evolution function, the predicted covariance is estimated according to the weighted distribution of the predicted evolution of the sigma points.

The first step consists in calculating the sigma points $\mathcal{X}_{k-1}$ and the corresponding weights:

$$\mathcal{X}_k = \left[\hat{x}_k \quad \hat{x}_k + \gamma\sqrt{P_k} \quad \hat{x}_k - \gamma\sqrt{P_k}\right] \tag{33}$$

where $\sqrt{P_k}$ is the Cholesky decomposition of P_k.

The weights $W_i^{(m)}$ and $W_i^{(c)}$ are defined according to the parameters κ and η:

$$\begin{array}{ll} W_0^{(m)} = \frac{\kappa}{\eta+\lambda} & W_0^{(c)} = \frac{\kappa}{\eta+\lambda} + 1 - \alpha^2 + \beta \\ W_i^{(m)} = W_i^{(c)} = \frac{1}{2(\eta+\lambda)} & \forall i \in [1, 2n] \end{array} \tag{34}$$

1. In the *prediction* step, the predicted state vector $\hat{x}_{k+1|k}$ and its covariance $P_{k+1|k}$ are computed according to:

$$\mathcal{X}^*_{k+1|k} = f(\mathcal{X}_k, u_k) \tag{35}$$

The estimated state is the weighted centre of the sigma points.

$$\hat{x}_{k+1|k} = \sum_{i=0}^{2n} W_i^{(m)} \mathcal{X}^*_{i,k+1|k} \tag{36}$$

The estimated covariance is estimated from the distribution of the sigma points:

$$P_{k+1|k} = \sum_{i=0}^{2n} W_i^{(c)} \left[\mathcal{X}_i^* - \hat{x}_{k+1|k}\right] \left[\mathcal{X}_i^* - \hat{x}_{k+1|k}\right]^{\mathrm{T}} + Q \tag{37}$$

2. In the *update* step, new sigma points $\mathcal{X}_{k+1|k}$ are drawn with the estimated covariance to estimate the predicted measurement vector $\hat{y}_k$.

$$\mathcal{X}_{k+1|k} = \left[\hat{x}_{k+1|k} \quad \hat{x}_{k+1|k} \pm \gamma\sqrt{P_{k+1|k}}\right] \tag{38}$$

$$\mathcal{Y}_{k+1} = H\mathcal{X}_{k+1|k} \tag{39}$$

$$\hat{y}_{k+1} = \sum_{i=0}^{2n} W_i^{(m)} \mathcal{Y}_{i,k+1} \tag{40}$$

The residual covariance Λ is estimated by:

$$\Lambda_{k+1} = \sum_{i=0}^{2n} W_i^{(c)} \left[\mathcal{Y}_{i,k+1} - \hat{y}_{k+1}\right] \left[\mathcal{Y}_{i,k+1} - \hat{y}_{k+1}\right]^{\mathrm{T}} + R \tag{41}$$

The filter gain K is then computed with:

$$K_{k+1} = P_{\hat{x}\hat{y}_{k+1}} \Lambda_{k+1}^{-1} \tag{42}$$

where

$$P_{\hat{x}\hat{y}_{k+1}} = \sum_{i=0}^{2n} W_i^{(c)} \left[\mathcal{X}_{i,k+1|k} - \hat{x}_{k+1|k}\right] \left[\mathcal{Y}_{i,k+1} - \hat{y}_{k+1}\right]^{\mathrm{T}} \tag{43}$$

Finally, the state estimate and its covariance are updated according to the gain and the current measurement vector y:

$$\hat{x}_{k+1} = \hat{x}_{k+1|k} + K_{k+1}\left(y_{k+1} - \hat{y}_{k+1}\right) \tag{44}$$

$$P_{k+1} = P_{k+1|k} - K_{k+1}\Lambda_{k+1}^{-1}K_{k+1}^{\mathrm{T}} \tag{45}$$

4. Vehicle Dynamics Model

Model-based FDI schemes for SBW systems generally rely on a simplified linear single-track model, generally called the bicycle model (see [5,35,36]). However, BBW systems generates asymmetrical braking forces, which cannot be treated with the bicycle model. Moreover, this linear model does not cope with the non-linearities due to large tyre slips, which are likely to occur in case of braking failure.

Thus, we have opted for simplified non-linear two-track model, which is presented in this section.

4.1. Two-Track Vehicle Model

Based on Newton's second law of motion (see Equation (46), the longitudinal, lateral and yaw accelerations a_x, a_y, and $\ddot{\psi}$ at the centre of gravity G are expressed according to the estimated tyre–road forces F_{x_i}, where $i \in [1, 2, 3, 4]$ indicates the wheel index, as represented in Figure 2. I_z is the vehicle moment of inertia, F_{ext} represents the external forces acting on the vehicle (rolling and air resistance), supposed longitudinal and steady.

$$\left\{\begin{array}{lcl} a_x & = & \frac{1}{M}\left(F_{ext} + \sum_{i=1}^{4} F_{x_i}\right) \\ a_y & = & \frac{1}{M}\sum_{i=1}^{4} F_{y_i} \\ \ddot{\psi} & = & \frac{1}{I_z}\sum_{i=1}^{4} \|\overrightarrow{GP_i} \wedge \overrightarrow{F_i}\| \end{array}\right. \tag{46}$$

The evolution of the vehicle longitudinal and lateral velocities v_x and v_y at point G are then expressed according to Equation (47).

$$\left\{\begin{array}{lcl} \dot{v}_x & = & a_x + v_y\dot{\psi} \\ \dot{v}_y & = & a_y - v_x\dot{\psi} \end{array}\right. \tag{47}$$

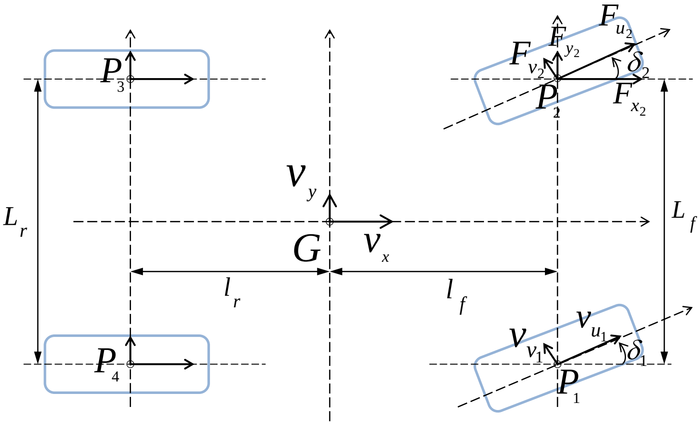

Figure 2. The two-track model.

4.2. Tyre–Road Force Estimation

Several models have been presented in the literature to estimate the tyre–road forces, according to the tyre longitudinal and lateral tyre slip rates λ_i and α_i defined by Equation (48).

$$\left\{\begin{array}{lcl} \lambda_i & = & \frac{R_i\omega_i - v_{u_i}}{\max(R_i\omega_i, v_{u_i})} \\ \alpha_i & = & \delta_i - \arctan\left(\frac{v_{v_i}}{v_{u_i}}\right) \end{array}\right. \tag{48}$$

where v_{u_i} and v_{v_i} are the expression of the wheel velocity $\overrightarrow{v_i}$ along the wheel main axis. This velocity is computed from the vehicle velocity at point G and its turn speed vector $\overrightarrow{\Omega}$.

$$\overrightarrow{v_i} = \overrightarrow{v_G} + \overrightarrow{\Omega} \wedge \overrightarrow{GP_i} \tag{49}$$

where P_i is the position of the centre of the wheel.

The model proposed by Dugoff et al. [37] has the advantages of relying on only two parameters, the longitudinal and lateral tyre stiffness C_x and C_y, which can be easily estimated with experimental tests.

The road–tyre force, which presents a saturation for high values of λ_i and α_i, is expressed in the wheel main axis by:

$$\left\{\begin{array}{lcl} F_{u_i} & = & C_x\frac{\lambda_i}{1-\lambda_i}k_i \\ F_{v_i} & = & C_y\frac{\tan\alpha_i}{1-\lambda_i}k_i \end{array}\right. \tag{50}$$

with

$$k_i = \begin{cases} (2-\sigma_i)\sigma_i & if \quad \sigma_i < 1 \\ 1 & if \quad \sigma_i \geq 1 \end{cases} \tag{51}$$

and

$$\sigma_i = \frac{(1-\lambda_i)\mu_i F_{z_i}}{2\sqrt{C_x^2\lambda_i^2 + C_y^2 \tan^2 \alpha_i}} \tag{52}$$

The repartition of the normal forces F_{z_i} on the wheels is given by:

$$\begin{cases} F_{z_1} = \frac{M}{l_f+l_r}\left[l_r g - ha_x\left(\frac{1}{2} + \frac{ha_y}{L_f g}\right)\right] \\ F_{z_2} = \frac{M}{l_f+l_r}\left[l_r g - ha_x\left(\frac{1}{2} - \frac{ha_y}{L_f g}\right)\right] \\ F_{z_3} = \frac{M}{l_f+l_r}\left[l_f g + ha_x\left(\frac{1}{2} - \frac{ha_y}{L_r g}\right)\right] \\ F_{z_4} = \frac{M}{l_f+l_r}\left[l_f g + ha_x\left(\frac{1}{2} + \frac{ha_y}{L_r g}\right)\right] \end{cases} \tag{53}$$

where l_f (respectively, l_r) is the distance between G and the front (respectively, rear) axle, h is the height of the centre of gravity, and L_f and L_r the front and rear axle widths. Finally, the forces are expressed in the vehicle referential according to the steering angle δ_i.

$$\begin{cases} F_{x_i} &= F_{u_i}\cos\delta_i - F_{v_i}\sin\delta_i \\ F_{y_i} &= F_{v_i}\cos\delta_i + F_{u_i}\sin\delta_i \end{cases} \tag{54}$$

4.3. Actuators Models

To detect actuator faults, it is mandatory to take into account the actuators in the model. In our case, the actuators are the brake callipers, and the motor on the steering rack to settle the front steering angle.

The steering motor is directly controlled by angle. Its response is very fast compared to the vehicle dynamics, so that the actuation delay can be neglected. The actual angle δ_i is then supposed equal to the commanded angle δ_u.

$$\delta_i = \delta_u \tag{55}$$

Concerning the brakes, the evolution of the wheel dynamics requires the estimation of the braking torque which is generally not measurable.

In the case of an electromagnetic braking (for instance, for regenerative braking), which are standard devices on electric vehicles, the torque can be estimated from the voltage and the current applied to the solenoid.

In the case of conventional hydraulic brakes, the braking torque in controlled according to the hydraulic pressure on the calliper. This braking pressure is typically measured by the ESC control unit and could be used to estimate the braking torque.

In our test vehicle (see Section 5), which was originally not fitted with an ESC system, a hydraulic unit was integrated into the braking circuit to maintain a commanded slip rate. However, the braking pressure cannot be measured and, therefore, the estimation of the actual braking torque is more difficult. The following model, detailed in [38], was consequently implemented.

The ESC control unit contains 12 solenoid valves and a pump. According to the combination of activated valves, the hydraulic pressure at the callipers can be controlled in the following four different modes:

- "normal" mode: The pressure from the master cylinder is normally transmitted to the calliper.
- "maintained" mode: The calliper pressure is maintained constant.
- "released" mode: The pump is activated to release the pressure.
- "braked" mode: The pressure is increased, without any driver action.

In a first step, the resulting braking torque T_{b_i} is supposed to vary linearly according to the state (see Equation (56). The parameters ΔT_b and ΔT_r are set experimentally. Of course, the braking torque cannot be negative.

$$T_{b_i}(t+\Delta t) = \begin{cases} T_{b_i}(t) + \Delta T_b & \text{if} \quad \text{state is ``braked''} \\ T_{b_i}(t) & \text{if} \quad \text{state is ``maintained''} \\ T_{b_i}(t) - \Delta T_r & \text{if} \quad \text{state is ``released''} \end{cases} \tag{56}$$

Finally, the wheel turn acceleration is computed from the estimated braking torque, the tyre–road longitudinal force and the effective wheel radius R_i:

$$\ddot{\omega}_i = \frac{1}{I_{w_i}} \left(T_{b_i} - R_i F_{u_i}\right) \tag{57}$$

4.4. Implementation of the Probabilistic Observers

The output vector is composed of the different sensor output variables: $y = [a_x, a_y, \dot{\psi}, \omega_1, \omega_2, \omega_3, \omega_4, \delta_i]^t$.

For a vehicle dynamics observer, the state vector $\hat{x}$ is composed of the estimates of the velocities $\hat{v}_x$ and $\hat{v}_y$. To allow the sensor fault detection, $\hat{x}$ has been augmented to obtain an estimate of each sensor data. Then, we obtain: $\hat{x} = [\hat{v}_x, \hat{v}_y, \hat{a}_x, \hat{a}_y, \hat{\dot{\psi}}, \hat{\omega}_1, \hat{\omega}_2, \hat{\omega}_3, \hat{\omega}_4, \hat{\delta}_i]^t$.

According to the previously mentioned vehicle model (see Equations (46), (47), (55) and (57), the following evolution function f is adopted for the observers implementation:

$$f = \begin{cases} \hat{v}_x(k+1) &=& \hat{v}_x(k) + \Big(\hat{a}_x(k) + \hat{v}_y(k)\hat{\dot{\psi}}(k)\Big)\Delta t \\ \hat{v}_y(k+1) &=& \hat{v}_y(k) + \Big(\hat{a}_y(k) - \hat{v}_x(k)\hat{\dot{\psi}}(k)\Big)\Delta t \\ \hat{a}_x(k+1) &=& \frac{1}{M}\Big(F_{ext} + \sum_{i=1}^{4} F_{x_i}\Big) \\ \hat{a}_y(k+1) &=& \frac{1}{M}\sum_{i=1}^{4} F_{y_i} \\ \hat{\dot{\psi}}(k+1) &=& \hat{\dot{\psi}}(k) + \ddot{\psi}(k)\Delta t \\ \hat{\omega}_1(k+1) &=& \hat{\omega}_1(k+1) + \frac{\Delta t}{I_{w_1}}\left(T_{b_1} - R_1 F_{u_1}\right) \\ \hat{\omega}_2(k+1) &=& \hat{\omega}_2(k+1) + \frac{\Delta t}{I_{w_2}}\left(T_{b_2} - R_2 F_{u_2}\right) \\ \hat{\omega}_3(k+1) &=& \hat{\omega}_3(k+1) + \frac{\Delta t}{I_{w_3}}\left(T_{b_3} - R_3 F_{u_3}\right) \\ \hat{\omega}_4(k+1) &=& \hat{\omega}_4(k+1) + \frac{\Delta t}{I_{w_4}}\left(T_{b_4} - R_4 F_{u_4}\right) \\ \hat{\delta}(k+1) &=& \delta_u(k+1) \end{cases} \tag{58}$$

The same evolution model (Equation (58)) is implemented for each of the 14 considered modes. The only differences between the different observers are the parameters of the noise matrices: measurement noise matrix R for the immunization to sensor faults, and model noise matrix Q for the actuator faults, as explained in Section 2.4.

5. Experimental Validation

The FDI algorithm was validated with experimental data, recorded with an experimental test vehicle on the Satory test tracks, in Versailles (France). The test scenarios included standards driving situations (normal urban driving) and critical maneuvers (emergency braking).

5.1. Experimental Vehicle

The test prototype was a SECMA f16, a leisure convertible rear-wheel drive vehicle. This prototype was modified to integrate drive-by-wire functionalities (SBW and BBW).

The vehicle has been fitted with proprioceptive sensors:

- an inertial measurement unit (IMU) with three-axis accelerometer and gyrometer (type Xsense MTI-G);
- four wheel turn rate sensors;
- a front wheel steering angle sensor; and
- a steering wheel angle sensor.

Thanks to an ESC hydraulic control unit, integrated on the braking hydraulic circuit between the master cylinder and the callipers, the braking pressure can be adapted to maintain a commanded slip rate for each wheel. More information on this system can be found in [38].

The sensor acquisition and the actuator control is ensured by a programmable electronic control unit (ECU, type AFT PROtronic), which integrates a micro-controller, standard interfaces (PWM, A/D converters) and CAN interfaces for communication between PC and data logging. The data logging sampling frequency is limited by the CAN interface features (max: 100 Hz).

5.2. Comparison of the Probabilistic Observers

A fast and reliable fault detection via a model-based FDI algorithm requires an accurate observation of the vehicle state. Moreover, to avoid false detection, the observation has to remain consistent, i.e., that the predicted error covariance stays compatible with the actual estimation error.

On another side, a major concern in the Multiple Model approaches is the computational load. Indeed, the probabilistic filters are computed for each model in parallel, thus multiplying the operation numbers. The observer has to achieve a reasonable computational time to stay compatible with an embedded real time architecture.

In a first step, we selected among the different considered observers (EKF, UKF and DD1) the one which suits the most the IMM approach, regarding the accuracy and the consistency of the observation and the computational time efficiency.

5.2.1. Observation Accuracy

The estimated state is composed of the longitudinal and the lateral speeds, and the estimates of the eight proprioceptive sensors (X- and Y-accelerations, yaw rate, wheel speeds and steering angle). To check the observers accuracy, we compared the estimates to reference sensor signals during highly dynamical driving situations like an emergency braking and sharp turns.

Figure 3 compares the estimation of the longitudinal speed during an emergency braking maneuver, with the three considered observers. The reference speed is given by a Correvit sensor (KISTLER S-350). The estimates given by the EKF, UKF and DD1 observers are, respectively, plotted in black, green and blue.

The driving maneuver consisted of a slight acceleration (from 1 s to 8 s), followed by a sharp right turn (from 8 s to 11 s), then a strong acceleration (from 11 s to 17 s), and finally an emergency braking (from 17 s to 20 s), with activation of the ABS system on the front wheels during the braking.

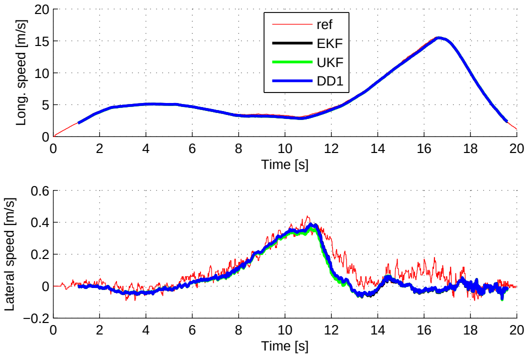

Figure 3. Longitudinal and lateral speed estimates during an emergency braking.

The three observers give a very good estimation of the vehicle velocity; the difference with the reference sensor remains below 0.3 m/s, and it is difficult to draw clear distinctions between the curves. Consequently, no significant differences can be made in terms of accuracy. The same conclusion can be made for the other estimated data. Table 1 details the root mean square (RMS) of the difference between the raw sensor signal and the estimated data for the vehicle sensors during the same maneuver. The estimation error remains of the order of the sensor noises.

Table 1. Estimation error RMS during normal driving situation.

Sensor	ω_i	a_x	a_y	$\dot{\psi}$	δ
EKF	0.1261	0.1199	0.1093	0.0046	0.00045
UKF	0.1215	0.1191	0.1234	0.0045	0.00046
DD1	0.1217	0.1198	0.1234	0.0045	0.00046
unit	rad/s	m/s^2	m/s^2	rad/s	rad

To sum up, EKF, UKF and DD1 provide accurate estimation of the vehicle state, even in the case of a highly dynamic driving situation.

5.2.2. Observer Consistency

Figure 4 compares the estimation error for the different vehicle sensors to the estimated precision margin. The precision margin correspond to ± 3σ where the standard deviation σ is issued from the covariance matrix. Only the results for the EKF observer are displayed. The other observers present very similar results.

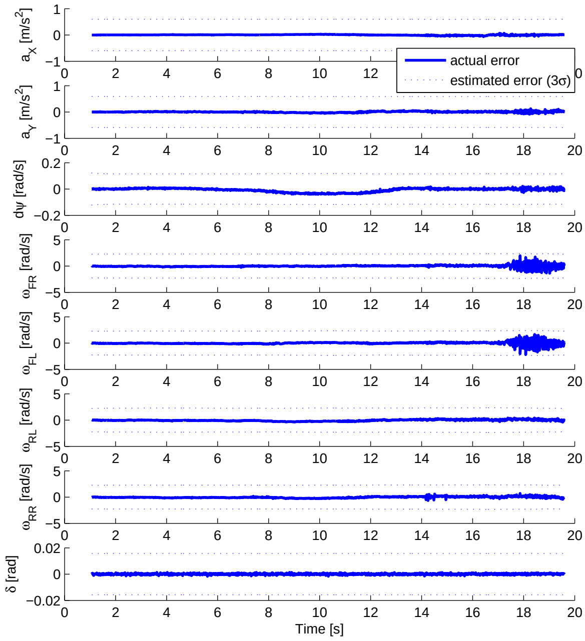

Figure 4. Estimation error consistency check during an emergency braking.

All signals remain inside the estimated precision margin, even during the ABS activation, observable at time 17 s on the front wheel speed signals. The observer is consistent.

5.2.3. Computational Time

The computational load has been measured with the Matlab function *profile* with the same input signals as previously (see Table 2).

Table 2. Computational load of the filters.

Observer	Call of the Evolution Function per Algo Cycle	Total Computational Time
EKF	1	1.37 s
UKF	21	11.5 s
DD1	37	26.6 s

The UKF needs six times more resources than the EKF. This is even more spectacular with the DD1, which is 15 times more time-consuming. This can be explained by the number of call of the evolution function during each algorithm cycle. The UKF does not require the computation of the Jacobian matrix, however the evolution function, which is quite complex in our case (computation of the tyre/road forces), has to be evaluated for each of the 21 sigma points, thus increasing dramatically the computational costs. The DD1 algorithm necessitates many calls of the evolution function for the computation of the divided differences matrix, together with the required resources for the Cholesky triangulations.

5.2.4. Conclusion on the Observers Comparison

This study has shown no significant differences regarding the accuracy and the consistency of the considered observers. However, in terms of computational time, the EKF is in our case much more efficient and therefore more adequate for the IMM implementation. In the following, we consequently implement the EFK for the IMM-based FDI algorithm.

5.3. Sensor Fault Detection Performances

The test scenario consists of an urban driving situation with a slight acceleration followed by a right turn.

For each sensor, a partial fault has been simulated by adding an offset to one of the initially measured data. The amplitude of this fault has been varied to investigate the detection of partial faults. The fault occurs at time $t = 4$ s and disappears at time $t = 5$ s.

For the first simulation (see Figure 5 , zoomed in at Figure 6), a fault on the X-acceleration sensor is simulated with a 5 m/s^2 amplitude.

The curves at the top display the estimates of X-acceleration for each mode. Before the fault occurrence, all the modes follow precisely the measured signal, except for the SA_X mode, which considers a very large noise for this sensor, thus estimating only roughly the actual signal. By looking at the likelihood (curves in the middle, with logarithmic scale), the nominal mode present a very good likelihood (value close to 1), while for the other modes, the likelihood is quite poor (<0.1), so that the probability of activation (at the bottom) of the nominal mode stays close to 100%.

When the fault occurs (time = 4 s), the estimate of all modes (except the SA_X mode) stick to the faulty sensor value, so that the likelihoods drop rapidly. The mode SA_X, which is almost not affected by the sensor fault, presents the best likelihood. Consequently, its probability of activation rises progressively to 100%, meaning that a fault occurred on the X-acceleration sensor: the sensor fault is thus correctly detected. The time to detection reaches 60 ms, i.e., six algorithm cycles.

Similar results are obtained with the other vehicle sensors.

To investigate the limit of the detectable faults, the same simulation was carried out with different fault amplitudes for each sensor. The detectable faults thresholds are given in Table 3.

Table 3. Amplitude of detectable sensor faults.

Sensor	ω_i	a_x	a_y	$\dot{\psi}$	δ
ampl.	5.0 rad/s	1.6 m/s^2	2.7 m/s^2	0.35 rad/s	0.055 rad

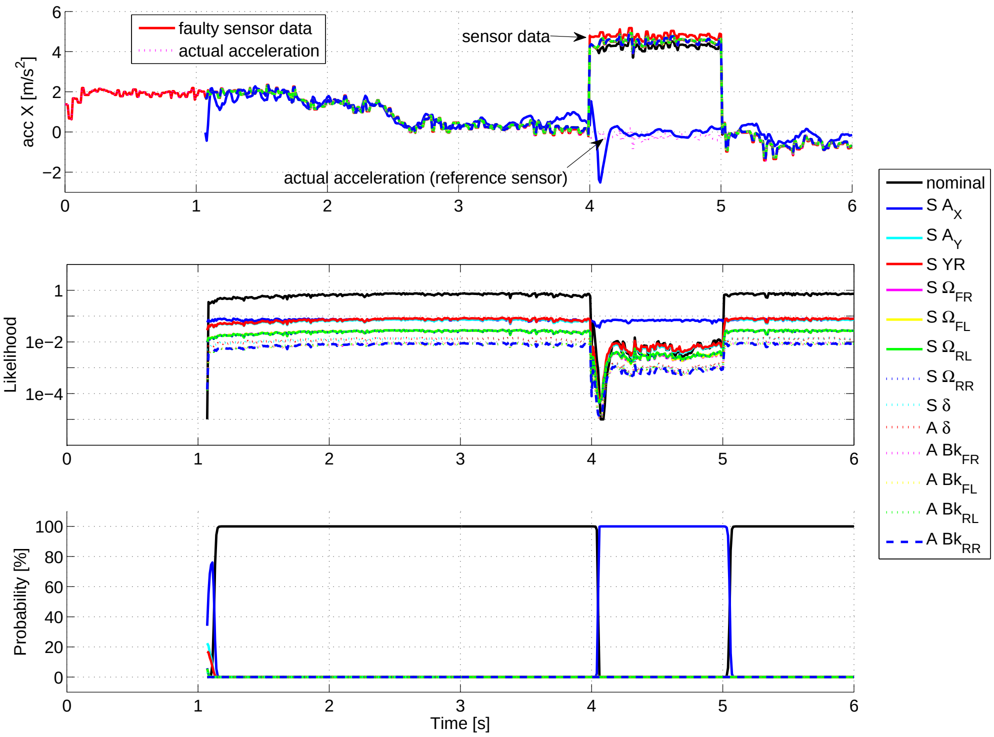

Figure 5. Fault on the X-acceleration sensor.

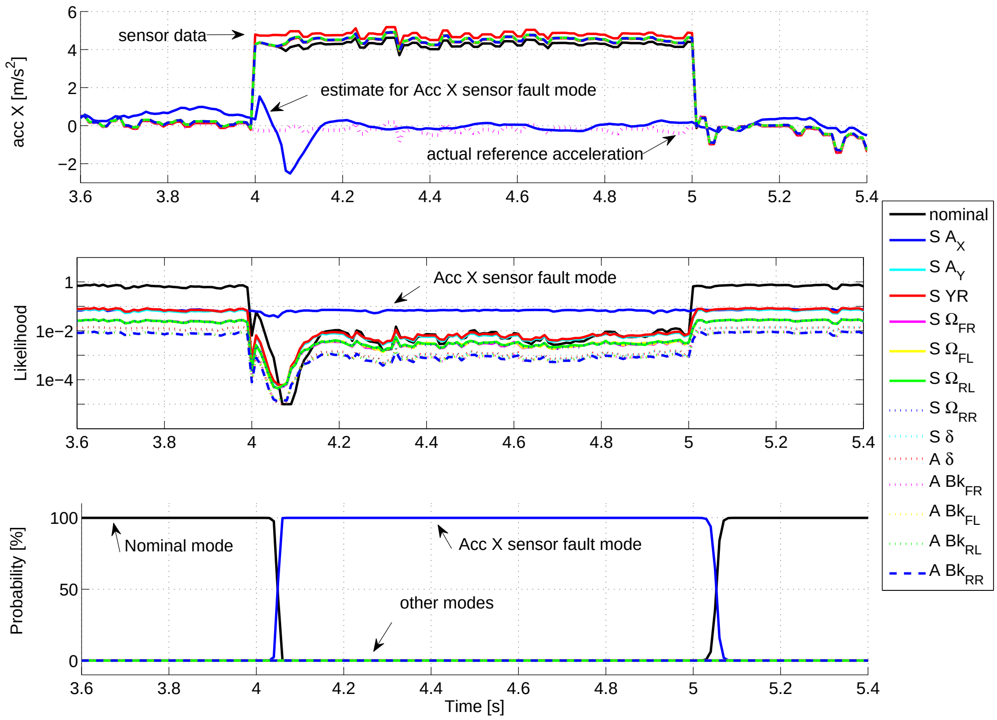

Figure 6. Fault on the X-acceleration sensor (zoom).

5.4. Actuator Fault Detection

The detection of actuator faults is generally more complicated than for sensor faults, as it requires a precise model of the actuator operation.

Concerning the steering actuator, the steering control system is supposed to be very accurate and fast, compared to the vehicle dynamics, so that the actuator delay can be neglected, as shown in Equation (55). To simulate an actuator fault, the target steering angle δ_u has been modified by adding an offset value, while the actual steering angle is not modified. The scenario is the same as for the sensor faults, the fault is activated between time = 4 s and 5 s with an amplitude of 0.1 rad.

The results are displayed in Figure 7. When the fault occurs, all the modes present a large likelihood drop, except the modes $S\delta$ and $A\delta$ which are, respectively, insensitive to steering sensor and steering actuator faults, for which the likelihood drop is weak. For the $S\delta$ mode, the estimated steering angle sticks to the target angle, as the measured angle is almost not taken into account, while the $A\delta$ estimate correctly the actual angle. Its likelihood remains consequently better than the $S\delta$ mode and the corresponding probability of activation rises to 100%. The fault detection threshold is crossed after 40 ms, confirming the correct detection and identification of the actuator fault.

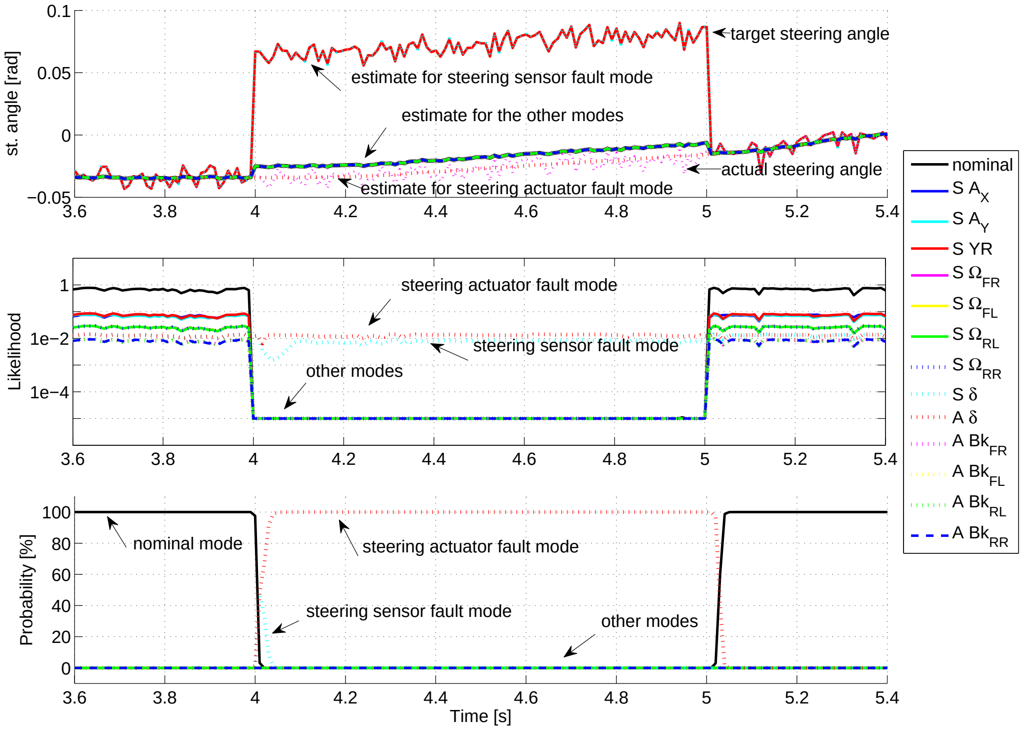

Figure 7. Fault on the steering actuator.

Concerning the brakes, as explained in Section 4.3, as no information on the braking pressure is available for the test vehicle, the actual braking torque is very difficult to estimate. The implemented model provides only a rough prediction of the braking torque. The detection of a brake actuator requires consequently a very sensitive setting for the corresponding modes. Under this condition, it was not possible to find a satisfying compromise between fault detection and robustness to false detection.

In future work, an implementation of this algorithm on a driving simulation software is planned to investigate the detection of brake faults under more favourable conditions (e.g., availability of the braking pressure and precise knowledge of the actuator properties).

5.5. Robustness to False Detection

To check the robustness to false detection, the sensor signals have been recorded during an urban driving scenario (150 s), including sharp turns, roundabout, emergency braking and road bumps. No false detection has been reported during this test (see Figure 8). The passing of a pothole (time = 14 s) and a speed bump (at time 84 s) provokes a slight nominal probability drop.

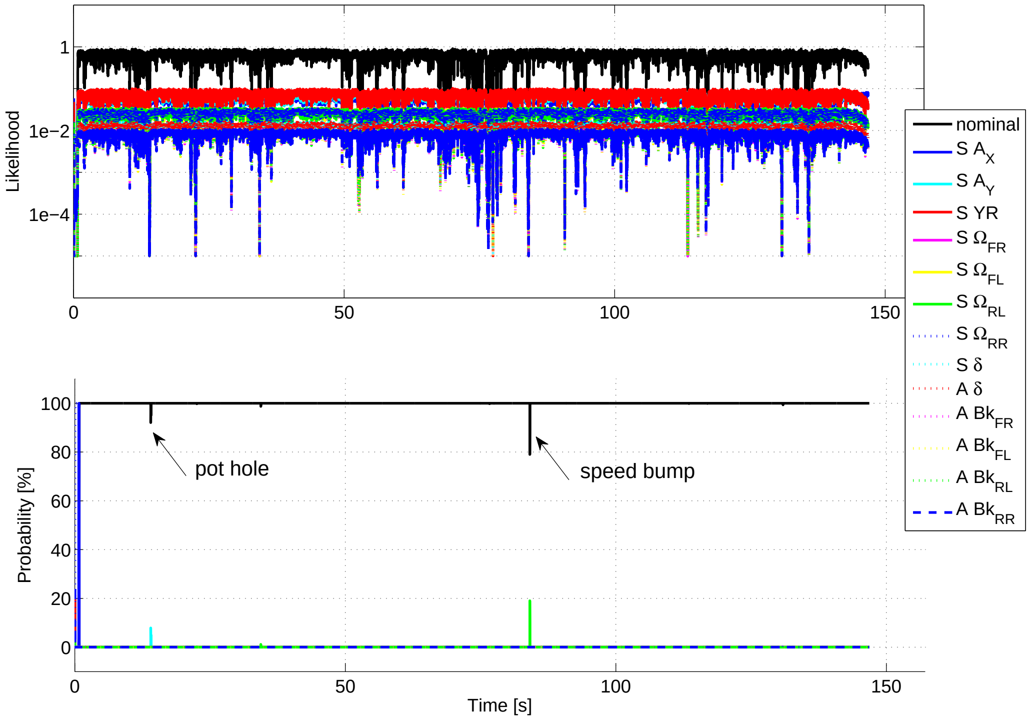

Figure 8. Robustness to false detection in urban driving.

5.6. Fault Tolerant Velocity Estimation

Next to the fault detection, the IMM algorithm provides a fault-tolerant observation via the overall estimate. This can be precious when implementing active fault-tolerant control of the vehicle, which relies on the velocity of the vehicle.

We check here the sensitivity of the velocity estimation to sensor faults. Figure 9 (respectively, Figure 10) shows the overall estimate of the longitudinal (respectively, lateral) vehicle velocity in case of a strong X-acceleration (respectively, Y-acceleration) sensor fault (+5 m/s^2 offset). For information, the estimate given by a simple model EKF observer is displayed.

It shows that the velocity estimation remains very accurate, even during a strong sensor fault.

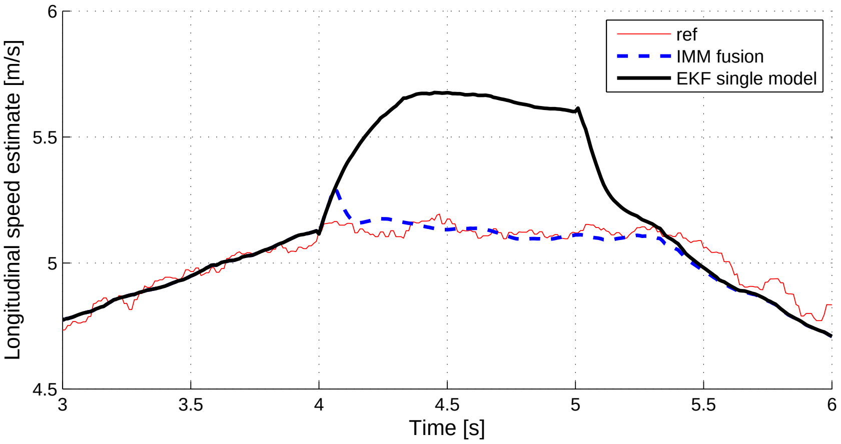

Figure 9. Longitudinal speed estimate with X-acceleration sensor fault.

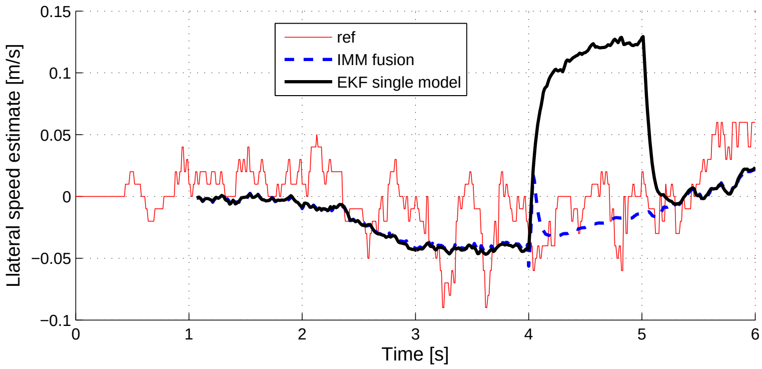

Figure 10. Lateral speed estimate with Y-acceleration sensor fault.

6. Conclusions

In this study, we discussed the detection and the isolation of embedded sensor and actuator faults for a drive-by wire vehicle via an adapted IMM algorithm. The compatibility of different probabilistic observers was investigated: the extended Kalman filter, the unscented Kalman filter and the first-order divided differences filter. If the three filters give similar performances, the EKF outperforms the others in terms of computational costs. The results, based on experimental data, show that total and partial faults for each of the proprioceptive sensors can be detected and identified within a short number of algorithm cycles. The detection of actuator faults seems more tedious: while a steering actuator fault can be identified correctly, the detection of braking faults has not been achieved and would require more investigations. The robustness of the detection (false detection rate) seems to be adequate, and should be confirmed with complementary experimental data.

In parallel, this approach allows a fault-tolerant estimation of the vehicle dynamics, which is interesting for vehicle control purpose. Future work will focus on the implementation of a fault-tolerant control strategy, which could take into consideration the identified faults to maintain a degraded vehicle controllability state in case of faults.

Author Contributions: V.J., S.G. and D.G. defined the methodology of the study; V.J. conceived and designed the experiments; S.G. and S.M. supervised the study; V.J. wrote the paper.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

LIVIC	Laboratory on Interactions Vehicle-Infrastructure-Driver
IBISC	Informatique, Biologie Intégrative et Systèmes Complexes
DBW	Drive-By-Wire
BBW	Brake-By-Wire
SBW	Steer-By-Wire
FDI	Fault Detection and Isolation
IMM	Interacting Multiple Models
HMI	Human Machine Interface
ESC	Electronic Stability Control
INS	Inertial Navigation Sensor
SAS	Steering wheel Angle Sensor
MM	Multiple Models
EKF	Extended Kalman Filter
DD1	First-order Divided Differences
UKF	Unscented Kalman Filter

References

1. Ackermann, J.; Bünte, T.; Odenthal, D. *Advantages of Active Steering for Vehicle Dynamics Control*; The Pennsylvania State University: State College, PA, USA, 1999.
2. Hayama, R.; Higashi, M.; Kawahara, S.; Nakano, S.; Kumamoto, H. Fault-tolerant automobile steering based on diversity of steer-by-wire, braking and acceleration. *Reliab. Eng. Syst. Saf.* **2010**, *95*, 10–17. [CrossRef]
3. Zhang, J.; Rizzoni, G.; Cordoba-Arenas, A.; Amodio, A.; Aksun-Guvenc, B. Model-based diagnosis and fault tolerant control for ensuring torque functional safety of pedal-by-wire systems. *Control Eng. Pract.* **2017**, *61*, 255–269. [CrossRef]
4. Wilwert, C.; Navet, N.; Song, Y.Q.; Simonot-Lion, F. Design of automotive X-by-Wire systems. In *The Industrial Communication Technology Handbook*; CRC: Boca Raton, FL, USA, 2005.
5. Anwar, S. Fault Detection, Isolation, and Control of Drive by Wire Systems. In *Fault Detection*; InTechOpen Access Publisher: London, UK, 2010.
6. Pisu, P.; Serrani, A.; You, S.; Jalics, L. Adaptive threshold based diagnostics for steer-by-wire systems. *J. Dyn. Syst. Meas. Control* **2006**, *128*, 428–435. [CrossRef]
7. Gadda, C.D.; Laws, S.M.; Gerdes, J.C. Generating diagnostic residuals for steer-by-wire vehicles. *IEEE Trans. Control Syst. Technol.* **2007**, *15*, 529–540. [CrossRef]
8. He, L.; Zong, C.; Wang, C. A Steering-by-Wire fault-tolerance control strategy based on multi-dimension Gauss Hidden Markov model. In Proceedings of the 2010 International Conference on Intelligent Control and Information Processing (ICICIP), Dalian, China, 13–15 August 2010; pp. 227–230.
9. Dos Santos, B.; Araújo, R.E.; Lopes, A. Fault Detection Scheme for a Road Vehicle with Four Independent Single-Wheel Electric Motors and Steer-By-Wire System. In *Advanced Vehicle Control, Proceedings of the 13th International Symposium on Advanced Vehicle Control (AVEC'16), Munich, Germany, 13–16 September 2016*; CRC Press: Boca Raton, FL, USA, 2016; p. 417.
10. Yu, M.; Wang, D. Model-Based Health Monitoring for a Vehicle Steering System with Multiple Faults of Unknown Types. *IEEE Trans. Ind. Electron.* **2014**, *61*, 3574–3586.
11. Murphey, Y.L.; Masrur, A.; Chen, Z.; Zhang, B. A fuzzy system for fault diagnostics in power electronics based brake-by-wire system. In Proceedings of the 2005 Annual Meeting of the North American Fuzzy Information Processing Society (NAFIPS 2005), Detroit, MI, USA, 26–28 June 2005; pp. 326–331.
12. Skarin, D.; Karlsson, J. Software implemented detection and recovery of soft errors in a brake-by-wire system. In Proceedings of the 2008 Seventh European Dependable Computing Conference, Kaunas, Lithuania, 7–9 May 2008; pp. 145–154.
13. Isermann, R.; Schwarz, R.; Stölzl, S. Fault-tolerant drive-by-wire systems. *IEEE Control Syst.* **2002**, *22*, 64–81. [CrossRef]
14. Anwar, S. *Fault Tolerant Drive by Wire Systems: Impact on Vehicle Safety and Reliability*; Bentham Science Publishers: Emirate of Sharjah, UAE, 2012.
15. Isermann, R. Process fault detection based on modeling and estimation methods—A survey. *Automatica* **1984**, *20*, 387–404. [CrossRef]
16. Venkatasubramanian, V.; Rengaswamy, R.; Yin, K.; Kavuri, S.N. A review of process fault detection and diagnosis. *Comput. Chem. Eng.* **2003**, *27*, 293–346. [CrossRef]
17. Chen, J.; Patton, R.J. *Robust Model-Based Fault Diagnosis for Dynamic Systems*; Springer: New York, NY, USA, 2012.
18. Yin, S.; Gao, H.; Qiu, J.; Kaynak, O. Descriptor reduced-order sliding mode observers design for switched systems with sensor and actuator faults. *Automatica* **2017**, *76*, 282–292. [CrossRef]
19. Pourbabaee, B.; Meskin, N.; Khorasani, K. Sensor Fault Detection, Isolation, and Identification Using Multiple-Model-Based Hybrid Kalman Filter for Gas Turbine Engines. *IEEE Trans. Control Syst. Technol.* **2016**, *24*, 1184–1200. [CrossRef]
20. Ashari, A.E.E.; Nikoukhah, R.; Campbell, S.L. Active robust fault detection in closed-loop systems: Quadratic optimization approach. *IEEE Trans. Autom. Control* **2012**, *57*, 2532–2544. [CrossRef]
21. Xu, J.; Wang, J.; Li, S.; Cao, B. A Method to Simultaneously Detect the Current Sensor Fault and Estimate the State of Energy for Batteries in Electric Vehicles. *Sensors* **2016**, *16*, 1328. [CrossRef] [PubMed]
22. Jeon, N.; Lee, H. Integrated Fault Diagnosis Algorithm for Motor Sensors of In-Wheel Independent Drive Electric Vehicles. *Sensors* **2016**, *16*, 2106. [CrossRef] [PubMed]

23. Du, D. Fault detection for discrete-time linear systems based on descriptor observer approach. *Appl. Math. Comput.* **2017**, *293*, 575–585. [CrossRef]
24. Boukhari, M.R.; Chaibet, A.; Boukhnifer, M.; Glaser, S. Fault-Tolerant Control for Lipschitz Nonlinear Systems: Vehicle Inter-Distance Control Application. *IFAC-PapersOnLine* **2017**, *50*, 14248–14253. [CrossRef]
25. Nikoukhah, R.; Campbell, S.L.; Horton, K.G.; Delebecque, F. Auxiliary signal design for robust multimodel identification. *IEEE Trans. Autom. Control* **2002**, *47*, 158–164. [CrossRef]
26. Hallouzi, R.; Verhaegen, M.; Kanev, S. Multiple model estimation: A convex model formulation. *Int. J. Adapt. Control Signal Process.* **2009**, *23*, 217–240. [CrossRef]
27. Efe, M.; Atherton, D.P. The IMM approach to the fault detection problem. *IFAC Proc. Vol.* **1997**, *30*, 603–608. [CrossRef]
28. Zhang, Y.; Li, X.R. Detection and diagnosis of sensor and actuator failures using IMM estimator. *IEEE Trans. Aerosp. Electron. Syst.* **1998**, *34*, 1293–1313. [CrossRef]
29. Zhang, Y.; Jiang, J. Integrated active fault-tolerant control using IMM approach. *IEEE Trans. Aerosp. Electron. Syst.* **2001**, *37*, 1221–1235. [CrossRef]
30. Xu, Q.; Li, X.; Chan, C.Y. A Cost-Effective Vehicle Localization Solution Using an Interacting Multiple Model- Unscented Kalman Filters (IMM-UKF) Algorithm and Grey Neural Network. *Sensors* **2017**, *17*, 1431. [CrossRef] [PubMed]
31. Cosme, L.B.; Caminhas, W.M.; D'Angelo, M.F.; Palhares, R.M. A Novel Fault Prognostic Approach Based on Interacting Multiple Model Filters and Fuzzy Systems. *IEEE Trans. Ind. Electron.* **2018**. [CrossRef]
32. Jazwinski, A.H. *Stochastic Processes and Filtering Theory*; Courier Corporation: Chelmsford, MA, USA, 2007.
33. Julier, S.J.; Uhlmann, J.K. New extension of the Kalman filter to nonlinear systems. In Proceedings of the Signal Processing, Sensor Fusion, and Target Recognition VI (AeroSense'97), Orlando, FL, USA, 20–25 April 1997; pp. 182–193.
34. NøRgaard, M.; Poulsen, N.K.; Ravn, O. New developments in state estimation for nonlinear systems. *Automatica* **2000**, *36*, 1627–1638. [CrossRef]
35. Gadda, C.D.; Yih, P.; Gerdes, J.C. Incorporating a model of vehicle dynamics in a diagnostic system for steer-by-wire vehicles. In Proceedings of the 7th International Symposium on Advanced Vehicle Control, Arnhem, The Netherlands, 23–27 August 2004; Volume 4, pp. 779–784.
36. Yih, P.; Gerdes, J.C. Steer-by-wire for vehicle state estimation and control. In Proceedings of the 7th International Symposium on Advanced Vehicle Control, Arnhem, The Netherlands, 23–27 August 2004; Volume 4.
37. Dugoff, H.; Fancher, P.; Segel, L. *An Analysis of Tire Traction Properties and Their Influence on Vehicle Dynamic Performance*; Technical Report; SAE: Warrendale, PA, USA, 1970.
38. Judalet, V.; Dhaoui, M.; Glaser, S.; Mammar, S. Fuzzy Logic Slip Rate Controller for experimental Test Vehicle. In Proceedings of the 17th International IEEE Conference on Intelligent Transportation Systems (ITSC), Qingdao, China, 8–11 October 2014.