

ESTIMATION OF VEHICLE DYNAMICS FOR AUTOMATED COMMERCIAL VEHICLES

Zoltán Gyurkó, László Lindenmaier, András Szappanos and Huba Németh

Knorr-Bremse R&D Center Budapest

H-1119 Budapest, Major u. 69, Hungary

Received: November 9, 2020.

ABSTRACT

A major area of automotive development is the vehicle automation at various levels, starting from driver assistance to full automation systems. In case of commercial vehicles these functions are not only comfort and safety relevant, but supported by commercial advantages as well, especially in case of high or full automated heavy-duty vehicles (SAE Automation Levels 4 and 5), where the presence of a driver is not compulsory for a specific use case. The most relevant among these applications for commercial vehicles is the highway automation: an autonomous heavy-duty vehicle capable of driving on highways and performing a hub-to-hub transport of goods. Such systems can be specified to be capable for in-lane manoeuvres only or for multi-lanes as well. Safe operation of commercial vehicles, operating at high velocity (~80-90 km/h) with heavy gross weight (~40 t) requires several measures in the system architecture on many levels:

- Redundant brake system,
- Redundant steering system (e.g. steer-by-brake),
- Redundant power supply system,
- Redundant perception system,
- and finally, a redundant control system architecture.

An automated vehicle utilizes a versatile sensor cluster, with multiple environmental sensors, like radars, LiDARs and cameras. The provided data of these sensors have to be fused into a common environment model description that can be forwarded to the decision-making functions. This paper focuses to the estimation of vehicle dynamics of ego and peer traffic participants. Dynamic modelling of vehicles is utilized in many functions of the automated vehicles's perception and decision-making levels. In the sensor data fusion, the motion estimation of the detected peer objects requires appropriate state estimation models (e.g. Extended Kalman-filter). Another application of vehicle dynamic models is the scenario analysis, in which the dynamics of the ego vehicle (and its dynamic limits), the curvature of the road and the dynamics of peer vehicles are analysed together to determine the optimal target velocity for the ego vehicle. The applicability of the above-mentioned vehicle dynamic models and algorithms is supported by detailed vehicle simulations (Software-in-the-Loop tests) and real vehicle measurements performed on test tracks and public roads as well. The results indicate a reliable performance on both fusion and scenario analysis side.

Keywords: redundant sensor architecture, dynamic state estimation, Kalman-filter, autonomous vehicle, scenario analysis

1. INTRODUCTION

Automated commercial vehicles offer a significant safety and operation cost advantages over manually driven vehicles, especially for the so called long haulage vehicles, which are driving mostly on highways between logistic hubs. Such vehicles are usually composed of a combination of a tractor towing vehicle and a semi-trailer. The most essential information for an automated driving function are the vehicle dynamic states of both the ego vehicle-combinations as well as for peer traffic participants. The former is obtained from inertial and localisation sensors, while the latter ones from environmental perception system of the automated vehicle. This paper focuses how to provide these vehicle dynamic states in a reliable way to make a solid foundation for the driving maneuver decision making function.

2. REDUNDANT PERCEPTION SYSTEM

Automated driving systems require a precise environment model to plan a safe trajectory for the host vehicle. A general environment model consists of dynamic objects (vehicles and any traffic participants) and static objects (obstacles, road network elements). The most common representation of the environment is a schematic bird-eye view (Fig. 1) using the ego vehicle coordinate system fixed to the center of front bumper. Because of the high relative velocity between ego vehicle and other traffic participants and due to sensor uncertainty the environment perception of automated commercial vehicles is challenging. Here the focus is put to the perception of dynamic objects.

2.1 Redundant sensor cluster

A sensor cluster consists of different sensing units. A sensing unit contains a physical sensor module (like radar, camera, LiDAR, ultrasonic, differential GPS (dGPS), inertial measurement unit (IMU) etc.) and applied filtering/detecting algorithms. The filtering/detecting algorithm might be embedded within the sensor module electronic system (smart-sensor) or can be tailored to a standalone computational platform (co-processor). Instead of measuring sensor properties and algorithmic performance the relevant question for sensor cluster evaluation is the contribution of a given sensing unit on the perceived and fused environment representation. This approach highlights the bounded state of the detection algorithm and its physical sensor [1-4].

The environment model can be split into different zones as shown in Fig. 1. Front zone, Side zone (left/right), Rear looking zone (left/right). To achieve fail operational behaviour each zones are covered by multiple sensing units, i.e. redundant. The redundancy of detection zones must be ensured by safety and diagnostic modules.

Fig. 1 Bird-eye view representation of the ego vehicle environment and sensor field of views

By taking advantage of each sensors' physical properties, redundancy can further increase the accuracy of environment model constructed by using sensor fusion. For example a radar can provide good longitudinal position estimation but its angular resolution is lower. However, a camera based object detector can provide precise lateral in-lane positioning although its longitudinal estimation is less reliable.

2.2 Peer vehicle detection for highway automation

Highway operation limits the environmental complexity in comparison to an urban environment. However, the higher the travel speed, the more precise localization and peer vehicle state estimation is required. Peer vehicles have to be detected as approaching from behind, passing through the side zones up to far out of the frontal detection region. Each of these zones (rear, side and front) are introducing challenges and different types of danger for the decision-making functionality.

Rear looking zone is occupied by a trailer, so trailer articulation angle must be obtained to define the region, where peer vehicles are expected. Control algorithms are not dealing with the lane configuration behind host vehicle, but for object to lane assignment the lane configuration has to be identified in backward zone as well.

Backward lane detection does not mean a directional flip in algorithmic sense (e.g. tracking algorithms have to be tailored). In lane-line perception system HD Maps can also provide data as a standalone sensing source. HD Maps are required to be supported by precise localization (up to a couple of centimeters accuracy), in which case in addition they can be considered as an extracted lane detector and can be fused with other lane detectors to improve lane detection performance.

Side looking zone requires sensors with wide field of views rather than of far range scanning capability. Radars with near scanning zone are preferred and LiDARs can efficiently contribute to near range object detection because of dense point cloud. Fish-eye cameras are able to cover the side zone of host vehicle without leaving gaps between the backward and forward looking ones. Fish-eye viewpoints are also beneficial for ego lane positioning while they provide the perfect spot for lane mark center point detection.

Forward looking zone is the most extended one in terms of longitudinal distance. The biggest challenge is to precisely determine the dynamic state of peer objects including their distance, velocity and acceleration which is detailed in the following section. Lane configuration and curvature estimation have to reach its highest precision within the forward looking zone because this is the base for lateral vehicle control. This can be reached by fusing lane information derived from different camera sources and different perspectives such as fish-eye and forward looking cameras. Fish-eye camera can provide a centimeter precise ego lane relative positioning, while perspective cameras are able to estimate the relative angle and shape of lanes. HD Maps might provide the training data for machine learning algorithms for lane-line detection purposes.

2.3 State estimation problem of object detections

The object detection is most commonly performed by bounding boxes identifying the object boundary on the sensor space, e.g. image plane for cameras. The estimation of longitudinal and lateral distance of objects is highly sensitive to the bounding box. This uncertainty can be reduced by defining the most appropriate point for position estimation among the regional points defined by the bounding box. The transformation from sensors space to vehicles coordinates could be more accurate if the sensors points were obtained where the rear plane of the object is projected to the ground plane.

The lateral uncertainty that might lead to wrong lane associations can be decreased by selecting sensed points closer to the symmetry axis of the peer vehicle. Due to the perspective nature of environment sensors the peer vehicle's center point is shifted from the object's bounding box center towards the edges as the object deviates from the sensor space center. The more the peer vehicles is on the left side of the sensor space, the more it's center point is shifted towards the left edge of it's bounding box in sensor coordinates. To deal with this effect the symmetry axis of the peer vehicle is detected. The reliability of dynamic objects' state estimation can be further increased by other methods detailed in the following sections.

3. ESTIMATION OF DYNAMIC STATE OF PEER TRAFFIC PARTICIPANTS

The dynamic state of peer traffic participants, peer objects in the followings is usually estimated based on recursive Bayesian state estimation in environment perception of

driving assistant and highly automated driving functions. The Bayes-filter is derived from the Bayes' theorem and it estimates the system state recursively by the process model (motion model of target objects) and sensor model. Since in Bayes-filter the system models are given as general probability density functions the required computational capacity of this filter would be significant in most cases. Hence, usually some simplifications are introduced in system models that leads to numerically less complex algorithms such as Kalman-filter (KF) and Extended Kalman-filter (EKF) [5]. These algorithms are detailed in the following subsections. The state of peer objects are modelled by random state variables constructing a stochastic vector. In highly automated driving systems the most commonly estimated dynamic states of objects are the longitudinal and lateral components of relative position, velocity and acceleration with respect to the ego vehicle coordinate system. The $\mathbf{x}$ state vector of peer objects are given by (1).

$$\mathbf{x} = [d_x \quad d_y \quad v_x \quad v_y \quad a_x \quad a_y]^T \quad (1)$$

Since the state vector consists of random variables, the state uncertainty is given by the covariance matrix $\mathbf{P}$ of $\mathbf{x}$.

3.1 Theoretical background

The recursive Bayesian state estimation assumes that the system was a Markov process and the measurements $\mathbf{z}$ were the observations of a Hidden Markov model (HMM) represented by Figure 1. The model assumes that the $\mathbf{x}_k$ state in the k -th cycle depended only on the previous state and the measurement $\mathbf{z}_k$ is generated by the state of the corresponding cycle.

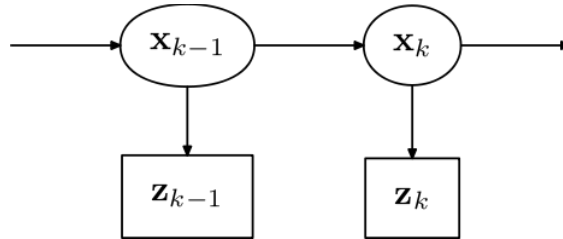


Fig. 2 Hidden Markov model

The probability density function of estimated state in k -th cycle is computed by (2).

$$p(\mathbf{x}_k | \mathbf{Z}_{1:k}) = \frac{p(\mathbf{z}_k | \mathbf{x}_k) p(\mathbf{x}_k | \mathbf{Z}_{1:k-1})}{p(\mathbf{z}_k | \mathbf{Z}_{1:k-1})} \quad (2)$$

In equation (2) $p(\mathbf{z}_k | \mathbf{x}_k)$ denotes the so-called sensor model, $p(\mathbf{x}_k | \mathbf{Z}_{1:k-1}) = \int p(\mathbf{x}_k | \mathbf{x}_{k-1}) p(\mathbf{x}_{k-1} | \mathbf{Z}_{1:k-1}) d\mathbf{x}_{k-1}$ is the predicted state based on the previous estimated state and the $p(\mathbf{x}_k | \mathbf{x}_{k-1})$ process model, while $p(\mathbf{z}_k | \mathbf{Z}_{1:k-1}) = \int p(\mathbf{z}_k | \mathbf{x}_k) p(\mathbf{x}_k | \mathbf{Z}_{1:k-1}) d\mathbf{x}_k$ is the denominator.

3.2 State estimation of peer objects with Kalman-filter

The Kalman-filter is derived from the general Bayes-filter by introducing some restrictions on the system models. The Kalman-filter assumes linear process- and sensor models. In addition, the measurement and process noise are assumed to follow normal

distribution with zero mean value. Similarly to Bayes-filter the Kalman-filter consists of two steps: the prediction and update. Since random variables are assumed to follow normal distribution the prediction of mean state (point estimation) and the state covariance is performed independently by (3) and (4).

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{F}_k \hat{\mathbf{x}}_{k-1|k-1} \quad (3)$$

$$\hat{\mathbf{P}}_{k|k-1} = \mathbf{F}_k \hat{\mathbf{P}}_{k-1|k-1} \mathbf{F}_k^T + \mathbf{Q}_k \quad (4)$$

where $\hat{\mathbf{x}}_{k-1|k-1}$, $\hat{\mathbf{x}}_{k|k-1}$, $\hat{\mathbf{P}}_{k-1|k-1}$ and $\hat{\mathbf{P}}_{k|k-1}$ denote the previous and predicted estimated state and covariance matrix respectively. $\mathbf{F}_k$ and $\mathbf{Q}_k$ are the transition matrix and the covariance matrix of process noise. In this application the u_k control input could be neglected since it is not observable at all. There are several dynamic models of peer objects such as constant velocity and constant rate of turn model, however the most commonly used is the constant acceleration model, for which the transition matrix is given by (5) as follows:

$$\mathbf{F}_k = \mathbf{F} = \begin{bmatrix} 1 & 0 & t_s & 0 & t_s^2/2 & 0 \\ 0 & 1 & 0 & t_s & 0 & t_s^2/2 \\ 0 & 0 & 1 & 0 & t_s & 0 \\ 0 & 0 & 0 & 1 & 0 & t_s \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (5)$$

where t_s denotes the sampling time.

Since the process model assumes that objects are travelled with constant acceleration, the process noise is given as acceleration noise. The $\mathbf{Q}_k$ process noise covariance is given by (6) to (8) based on the σ_{a_x} and σ_{a_y} acceleration noise values as filtering parameters.

$$\mathbf{v}_{k,x} = \sigma_{a_x} [t_s^2/2 \quad 0 \quad t_s \quad 0 \quad 1 \quad 0]^T \quad (6)$$

$$\mathbf{v}_{k,y} = \sigma_{a_y} [0 \quad t_s^2/2 \quad 0 \quad t_s \quad 0 \quad 1]^T \quad (7)$$

$$\mathbf{Q}_k = \mathbf{v}_{k,x} \mathbf{v}_{k,x}^T + \mathbf{v}_{k,y} \mathbf{v}_{k,y}^T \quad (8)$$

The update step is given by the following formulae:

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k (\mathbf{z}_k - \hat{\mathbf{x}}_{k|k-1}) \quad (9)$$

$$\hat{\mathbf{P}}_{k|k} = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \hat{\mathbf{P}}_{k|k-1} \quad (10)$$

where $\mathbf{K}_k$ is the optimal Kalman gain given by (11).

$$\mathbf{K}_k = \hat{\mathbf{P}}_{k|k-1} \mathbf{H}_k^T (\mathbf{H}_k \hat{\mathbf{P}}_{k|k-1} \mathbf{H}_k^T + \mathbf{R}_k)^{-1} \quad (11)$$

In equation (11) $\mathbf{H}_k$ denotes the measurement matrix that transforms the estimated state to measurement space and $\mathbf{R}_k$ is covariance matrix of measurement noise. The modern smart sensors provide sensor data in Cartesian coordinate system. For instance a camera is able to measure the relative longitudinal and lateral position of peer objects, radar can

observe the velocity components as well. The sensor matrices of these sensors are given by formulae (12) and (13).

$$\mathbf{H}_{k,scam} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (12)$$

$$\mathbf{H}_{k,srad} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \quad (13)$$

While the $\mathbf{R}_k$ measurement covariance is given as a diagonal matrix with the variance of measured variables.

3.3 State estimation of peer objects with Extended Kalman-filter

In many cases, however the process or the measurement model or both of them are non-linear. In these cases, the simple Kalman-filter is not applicable. Thus, the Extended Kalman-filter is applied that linearize the models in the small subspace of the estimated state. For example traditional radars and LiDARs provide their reflexions in polar coordinate system which leads to a nonlinear measurement model. In these cases the process model is the same as described in the previous subsection. The measurement model that transforms the state to measurement space is given as following:

$$\hat{\mathbf{z}}_{k|k} = g_k(\hat{\mathbf{x}}_{k|k}) \quad (14)$$

The measurement vector of a LiDAR and radar contain different elements. While LiDARs measure only the Euclidean-distance and azimuth angle of targets, radars are able to observe the radial velocity of them based on the Doppler-effect. Thus, the measurement model functions of these sensors are different, given by (15) and (16) where d , φ and v denote the Euclidean distance, the azimuth angle and the radial velocity of the objects in the sensor's vehicle coordinate system.

$$g_{k,LiDAR} = \begin{bmatrix} d \\ \varphi \end{bmatrix} = \begin{bmatrix} \sqrt{\hat{x}_1^2 + \hat{x}_2^2} \\ \tan^{-1} \frac{\hat{x}_2}{\hat{x}_1} \end{bmatrix} \quad (15)$$

$$g_{k,RADAR} = \begin{bmatrix} d \\ \varphi \\ v \end{bmatrix} = \begin{bmatrix} \sqrt{\hat{x}_1^2 + \hat{x}_2^2} \\ \tan^{-1} \frac{\hat{x}_2}{\hat{x}_1} \\ \sqrt{\cos^2(\varphi) \hat{x}_3^2 + \sin^2(\varphi) \hat{x}_4^2} \end{bmatrix} \quad (16)$$

Then, to apply the previously given Kalman-equations, the $\mathbf{H}_k$ sensor model matrix is estimated by the Jacobian-matrix of these sensor models. The sensor model matrices are described by following formulae:

$$\mathbf{H}_{k,LiDAR} = \left. \frac{\partial g_{k,LiDAR}}{\partial \mathbf{x}} \right|_{\hat{\mathbf{x}}_{k|k-1}} \quad (17)$$

$$\mathbf{H}_{k,RADAR} = \left. \frac{\partial g_{k,RADAR}}{\partial \mathbf{x}} \right|_{\hat{\mathbf{x}}_{k|k-1}} \quad (18)$$

Based on these recursive state estimation algorithms, the dynamic state of objects can be efficiently observed. They perform an optimal estimated state by filtering the measurement noise of sensors and they are able to reconstruct the hidden state variables that are not measured directly by the sensors. For instance, the relative velocity of objects detected by a camera can be estimated.

4. SCENARIO ANALYSIS BASED ON EGO AND PEER VEHICLE STATES

The above described dynamic states of peer vehicles are essential inputs of the decision preparation and decision making layer of highly autonomous vehicles' software. This data are utilized in various ways. In the present case it will be shown how it can be applied for the situation analysis (SIT) of a SAE Level 4 autonomous heavy-duty vehicle designed for highway applications [6]. One important subsystem of the situation analysis that utilizes the peer vehicle is the so-called scenario analysis (SA). SA is based on environmental information of the current scenario, in which the ego vehicle is present, several variables are determined that support the decision making process of the further step, the behaviour planner (BP).

4.1 Input information

Besides the dynamic states of peer objects and the ego vehicle there are several relevant inputs of SA. It considers the description of the road (lanes, lane lines), dynamic limits of the ego vehicle in its current state and other relevant information about the peer vehicles (classification, size, etc.). The dynamic state $\mathbf{x}$ of peer objects is described in the local ("front bumper") Cartesian coordinate system (x – longitudinal; y – lateral).

Based on these inputs the SA is able to determine several relevant decision supporting information:

- Safe driving speed: a global value for all lanes that depends on road conditions, ego vehicle capabilities, etc.
- Safe following distance: it can be defined as a composition of distance and time (TiV – Time-in-between-Vehicles) dimensions, depending on the velocity of the ego vehicle.
- Average driving speed: a global value for all lanes depending on the dynamics of the peer and ego vehicles as well as on the lane information (e.g.: maximum speed limit).
- Average following distance: a global value for all lanes depending on the dynamics of the peer and ego vehicles.

4.2 Calculation of safe driving speed

Safe driving speed is dependent on many external and internal factors:

- View range: defined as the maximum reliable range of lane line and object detection capability of the sensor cluster. It might be limited e.g. by unfavourable weather conditions (e.g. snow or fog) or heavy traffic density. Based on the view range, the road conditions (e.g. wet or icy road surface) and the dynamic limits (in this case most importantly the longitudinal deceleration capability (a_{xmin})) the maximum safe driving speed can be determined.

- Dynamic state of front peer objects: to avoid any collision with other traffic participants their dynamic state has to be considered and the driving speed has to be adjusted according to that.
- Speed limits: the maximum legally allowed speed limit for a given lane.
- Curvature of lanes (κ): depending on the road conditions and the dynamic state of the ego vehicle it can happen even on a highway that the driving speed has to be decreased.

Focusing in more detail to the last point, the following calculations can be performed. One typical representation of lanes lines are polynomials; for highway applications at least a 3rd order polynomial is required for the correct representation of the lane lines. From this representation the curvature can be calculated as follows:

$$\kappa_{lane}(x) = \frac{|f''(x)|}{(1+|f'(x)|)^{\frac{3}{2}}} \left[\frac{1}{m} \right] \quad (19)$$

Where:

$$f'(x) = 3 \cdot a_3 \cdot x^2 + 2 \cdot a_2 \cdot x + a_1 \quad (20)$$

$$f''(x) = 6 \cdot a_3 \cdot x + 2 \cdot a_2 \quad (21)$$

Where, a_1 , a_2 and a_3 are the coefficients of a 3rd order polynomial and x is the longitudinal distance, where the curvature is calculated. The choice of longitudinal distance depends on many factors, like the capabilities of the vehicle as well as the expected comfort of the driving behaviour. Taking into account the previously determined dynamic state of the vehicle (here the maximum possible lateral acceleration limit ($a_{y,max}$)) the safe driving speed can be determined.

$$v_{safe} = \frac{a_{y,max}}{\kappa_{lane}} \left[\frac{m}{s} \right] \quad (22)$$

To be on the safe side it is worth to consider the application of safety factors concerning the dynamic limits of the ego vehicle that takes into account the uncertainty of these values, caused by the limitations of sensors and detection algorithms.

4.3 SiL and vehicle tests

The developed algorithms have been evaluated using Software-in the-Loop (SiL) and vehicle testing as well (Fig. 3). Vehicle tests have been performed on proving grounds and public roads as well. The results showed good agreement between the simulation and vehicle tests.



Fig. 3 SiL and public road testing in various self driving scenarios

Fig. 4 shows test results from a test track (with one steeper curve around the middle) and from SiL of the same test track reconstructed. At 20 sec the safe driving speed is has been influenced by a slower front peer object, thus it is decreased to 21.7 m/s. As seen in Fig. 4 in case of small curvatures there is no limitation in maximum allowed speed (80 km/h $\sim$ 22.22 m/s or the speed of the peer object), while at around 57 sec an approximately 100 m radius curve is approached, where the safe driving speed is more limited. Without the limitation the lateral acceleration in the curve would have exceed the maximum allowed lateral acceleration of the vehicle, resulting in skidding or roll-over.

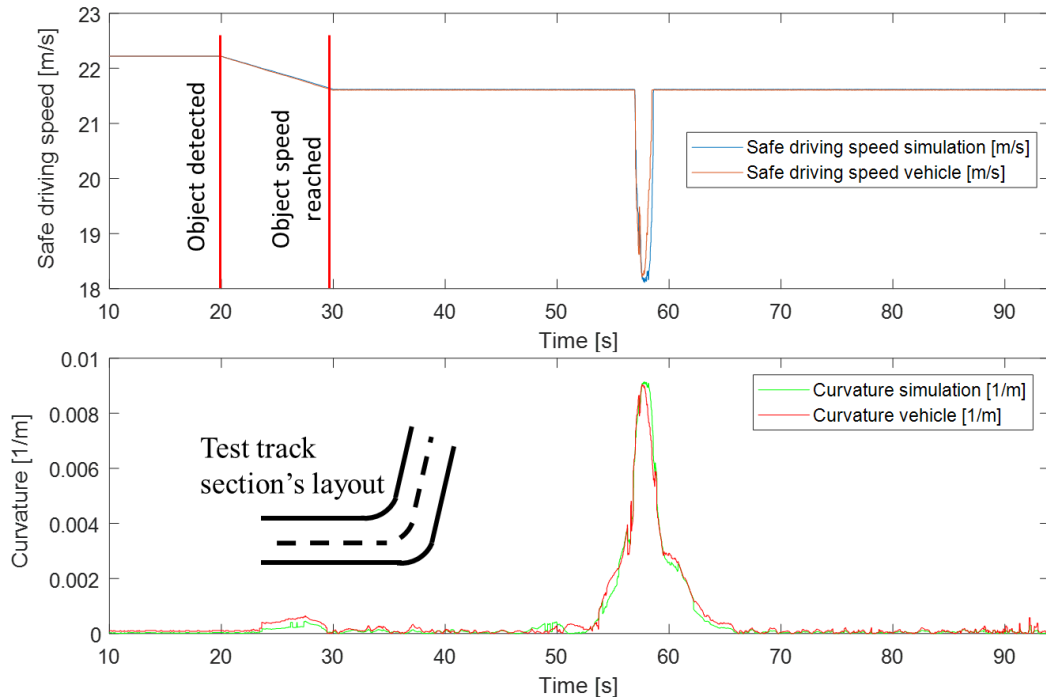


Fig. 4 Test results of safe driving speed calculation from SiL and vehicle tests

5. CONCLUSIONS

This paper aimed to provide a framework for the estimation of vehicle dynamic states of ego and peer vehicles of an automated commercial vehicle.

First the redundant environment perception system has been reviewed. The peer object detection has been discussed how different detection zones are needed to be covered with multiple sensors to achieve a fail operational capability for high automation. The object localisation in the vehicle coordinate system makes a challenge in case of off axis perspective position of the peer objects. The resulted peer object detections give then the baseline for the multi-object fusion framework according to the Bayes-filtering theorem, which has been proposed in its restricted form according to the Extended Kalman-filtering (EKF) framework to achieve reasonable performance-to-calculation effort ratio.

The peer environment objects states provided by the EKF fusion are then further processed by the situation analysis (SIT) layer that includes a scenario analyser module, which processing environmental information of the current scenario, in which the ego vehicle is present. Here supporting informations are processed for the maneuver decision, such as the safe driving speed, safe following distance, average driving speed or average following distance. The algorithms have been evaluated using Software-in the-Loop (SiL) and vehicle testing as well. The results showed a consistent and reliable performance in a wide range of operational conditions.

6. REFERENCES

- [1] **Liu, J. - Sun, Q. – Fan, Z. – Jia, Y.:** TOF Lidar Development in Autonomous Vehicle, IEEE 3rd Optoelectronics Global Conference, 2018.
- [2] **Steinbaeck, J. - Steger, C. – Holweg, G. – Druml, N.:** Next generation radar sensors in automotive sensor fusion systems, Sensor Data Fusion Trends, Solutions, Applications, 2017.
- [3] **Kuramoto, A. - Aldibaja, M. A. – Yanase, R. – Kameyama, J. – Yoneda, K. – Suganuma, N.:** Mono-Camera based 3D Object Tracking Strategy for Autonomous Vehicles, IEEE Intelligent Vehicles Symposium, 2018.
- [4] **Chavez-Garcia, R. O. - Vu, T. D. – Burler, J. – Avcard, O.:** Frontal Object Perception using radar and mono-vision, IEEE Intelligent Vehicles Symposium, 2012
- [5] **Carola, O.:** Fusion of Data from Heterogenous Sensors with Distributed Fields of View and Situation Evaluation for Advanced Driver Assistance Systems, KIT (Karlsruher Institut für Technologie) Scientific Publishing 2013. (p. 27-33)
- [6] **SAE, J3016:** Taxonomy and Definitions for Terms Related to On-Road Motor Vehicle Automated Driving Systems 2014.

