# A Novel Framework for Efficient Mobility Data Verification in Vehicular Ad-hoc Networks

Attila Jaeger[1] Norbert Bißmeyer[2] Hagen Stübing[3] Sorin A. Huss[1]

*Technische Universität Darmstadt, Integrated Circuits and Systems Lab* [1]
*Hochschulstraße 10, 64289 Darmstadt, Germany, {jaeger|huss}@iss.tu-darmstadt.de*

*Fraunhofer SIT, Secure Mobile Systems* [2]
*Rheinstraße 75, 64295 Darmstadt, Germany, norbert.bissmeyer@sit.fraunhofer.de*

*Adam Opel AG, Active Safety Systems* [3]
*Friedrich-Lutzmann-Ring, 65423 Rüsselsheim, Germany, hagen.stuebing@de.opel.com*

Most applications considered in *Vehicular Ad-hoc Networks* (VANETs) base their calculations on the location of vehicle and roadside units. Therefore, the trustworthiness of this data is essential in *Intelligent Transport System* (ITS) and can be addressed by digitally signing sent location information. However, we have to assume that an attacker is able to get valid secret keys and she or he thus may send authenticated messages with faked mobility information. In this work we therefore do not rely on encryption techniques only. Instead, we propose a novel framework for verifying mobility data, which aims at detecting messages representing non-plausible movement behaviour. A Kalman filter is exploited to detect malicious behaviour based on past vehicle movements. Regular changes of vehicle identifiers in the communication range due to privacy protection are made transparent in the mobility data verification framework. In order to enhance the framework, additional information from environmental sensors is integrated. To prove accuracy of our model, replaying of recorded traces and test drives were carried out.

***Keywords:*** *Car-to-X communication, mobility data verification, security, privacy, vehicle tracking*

## 1. Introduction

*Car-to-X* communication (C2X), which includes the communication between *ITS Vehicle Stations* (IVS) as well as the communication between vehicles and *ITS Roadside Stations* (IRS), is one of the most promising future technologies in order to reduce the number of fatal traffic accidents and enhance efficiency of road traffic. C2X communication is using a wireless ad-hoc network standardized by IEEE 802.11p [1]. For most traffic safety and traffic efficiency relevant applications on the IVS or IRS systems mobility data is essential. In addition, applications on the network or facility layer, e.g., GeoNetworking [2], depend on correct mobility information of adjacent ITS stations too. Therefore, the sender authentication and the message integrity are cryptographically secured as standardised in IEEE 1609.2 [3].

Besides protecting inter-vehicular communication the security of the sender's on-board network has to be trustable and protected from modification as proposed in the European research project EVITA [4]. Nevertheless, protecting on-board systems of every ITS station against manipulation, message injection, or modification is a very challenging and complex task.

To overcome the penetration dilemma nomadic devices will be present at early deployment phase of ITS [5], which may introduce certain security weaknesses. On the one hand security credentials stored in such devices may be more easily extracted than from sophisticated embedded vehicular on-board security hardware. Even if trusted platform modules are used the extraction of secure information, i.e., secret keys, is still possible as recently demonstrated [6]. On the other hand, as shown in [7], interfaces to vehicle on-board networks, e.g., CAN bus, as needed for nomadic devices, may be easily used to insert faked information into vehicle modules.

Consequently, especially in the deployment phase of VANET, the injection of bogus messages cannot be fully circumvented. Thus, countermeasures based solely on cryptography are not sufficient for a reliable protection of C2X communication.

In order to enhance trustworthiness of mobility information received from neighbour ITS stations, IEEE 1609.2 proposes a plausibility validation of message content. Thereby, application specific data, e.g., exterior temperature, should be verified best by the corresponding application. In contrast, common data, such as mobility information, should be checked in a common module in order to avoid multiple validations. In previous C2X projects, this part is rarely regarded. Consequently, we developed a framework for mobility data verification as proposed first in [8]. Therein, we advocate a Kalman filter-based approach to estimate a vehicle's future movements, which serve as a basis for mobility verification. Taking into account privacy considerations, vehicles may sporadically change their identifiers, which complicates mobility verification significantly. Nevertheless, our approach is tailored to provide reliable results in presence of such pseudonym changes [9].

Additionally to mobility verification applied in the sim$^{TD}$ field operational tests [8], in this work, we propose enhanced checks based on vehicles local sensors and an improved verification flow.

The structure of this work is as follows. After presenting general system assumptions and the assumed attacker model in section 2, related work considering mobility data verification is discussed in section 3. Detailed information about the framework and its vehicle tracker is presented in section 4. In section 5 an evaluation is presented, which involves test drives as well as supplemental replaying of recorded vehicle traces. Then some remarks on possible architecture integration into the communication stack follow in section 6. Finally, section 7 concludes the paper and discusses envisaged future work.

## 2. System Assumptions

Using commonly accepted system characteristics is essential to identify falsified messages in the network communication. Therefore, in this section system parameters are introduced followed by a discussion of possible attack scenarios.

### 2.1 System Model

The proposed Mobility Data Verification Framework is integrated in the communication stack of the vehicular communication system. Therefore, strong requirements regarding timing and resource consumption is given. The proposed framework considers all received single hop messages via the IEEE 802.11p communication link including mobility information.

The periodically sent *Cooperative Awareness Messages* (CAMs) and the event driven *Decentralized Environmental Notification* messages (DENs) contain the vehicle's position, velocity, heading, and vehicle dimensions, which are to be processed. The frequency of message transmission is controlled dynamically between 10Hz and 1Hz according to the vehicle's mobility [10]. Additionally, the maximum transmission range of 1000m may be reduced to 250m in case of channel congestion [11].

To ensure diver's privacy, in C2X communication spontaneously pseudonym changes may be performed by adjacent vehicles. The regular change of identifiers in the communication may be challenging for safety applications such as the Intersection Collision Warning, whose calculations rely on continuous traces of approaching vehicles. In order to support the correct operation of such applications, the pseudonym change has to be made transparent by assigning each vehicle a permanent identifier. It is important to mention that this identifier is only available for internal application processing and is not available outside the vehicle.

The proposed framework can be applied to each vehicle as well as to each road side station in order to support all kinds of applications that rely on correct mobility data from adjacent vehicles. The detection and exclusion of malicious messages sent by attackers is done locally on every system without sharing additional security information with other participants in the ad-hoc network.

### 2.2 Attacker Model

In the following, two use cases are described, which take periodically sent CAMs of neighbouring vehicles to trigger certain actions, e.g., displaying warning messages, and are thus highly sensitive with respect to incorrect mobility data. For our scenarios we assume a severe adversary with the following capabilities: The attacker is equipped with appropriate C2X sender hardware. Hence, she or he is able to send correctly encoded messages. Furthermore, the attacker is in possession of valid certificates. Therefore, messages sent by this attacker cannot be detected by means of cryptographic techniques.

The *Intersection Collision Warning* (ICW) according to [12] is a use case where vehicles are monitoring cross traffic at intersections in order to detect possible upcoming accidents. In the scenario illustrated in Figure 1 a static roadside attacker is sending a faked CAM, indicating a vehicle approaching the intersection at a very high speed. The application running on vehicle A's application unit then detects a potential hazard and notifies the driver accordingly. As we expect drivers to instantly react upon warnings, such a situation may lead to unexpectedly performed full brakes or lane changes.

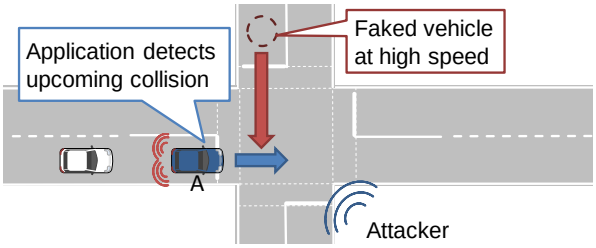


**Figure 1:** Attack on Intersection Collision Warning use case

Mobility data verification is not only necessary for Car-to-Car use cases, but can also prevent roadside facilities from processing faked data. In Figure 2 an attack manipulating the *Green Light Optimum Speed Advisory* (GLOSA) function is illustrated. Thereby, for traffic efficiency reasons, IRSs determine the tailback at every traffic light. Depending on the vehicle density on each lane, a prioritized flow control is performed. An attacker in vehicle A can take advantage of this functionality by simulating virtual cars on his lane in order to reduce his waiting time at the detriment of other road users.

The main objective of this work is thus aimed to a prevention of attacks on use cases, which derive their functionality from periodically sent CAMs.

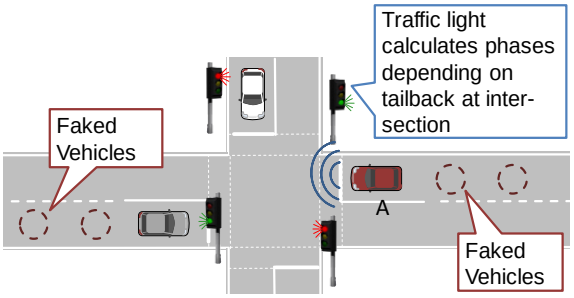


**Figure 2:** Attack on Green Light Optimum Speed Advisory use case

## 3. Related Work

Misbehaviour detection in C2X communication is considered in several different approaches due to the importance of correct neighbourhood information. A stationary roadside attacker is identified in [13] as the most threatening attack due to low complexity. As discussed in the attack on the GLOSA application, we additionally identify a moving attacker as rather threatening because its motivation may be higher compared to attackers that are not part of the road traffic.

The necessity of mobility data plausibility checks in the C2X communication has been also identified in [14]. The *Vehicle Behaviour Analysis and Evaluation Scheme* (VEBAS) proposed in [15] contains several basic checks to identify faked position claims. Testing the acceptance range of received messages, the maximum velocity and the message transmission frequency is adopted in our framework. The mobility verification presented in [8] adds additional movement analysis.

In [16] a concept is proposed that needs omni-directional Radar sensors to verify position claims from nodes in the direct neighbourhood. Due to the strongly limited observation range of Radar sensors these authors propose a routing topology that allows the usage of Radar information cooperatively in the neighbourhood.

Other position verification approaches according to [17] and [18] are based on techniques using radio modules that feature the *Received Signal Strength Indicator* (RSSI), which allows to calculate the sender's distance based on a radio model. Nevertheless, this position estimation technique is not very accurate. This is why it is not used for direct distance measurements. Instead, in [17] and [18] it is proposed, to apply an analysis of signal strength distribution indicating where the signal origins from.

The authors of [19] propose a relative location verification protocol using directional antennas to distinguish between vehicles in front and behind. Due to the inherent inaccuracy of position information we do not consider the RSSI technology in our work.

## 4. Mobility Data Verification Framework

In this section the framework, first proposed in [8], as well as additional enhancements for increasing the overall reliability are presented. An essential part of the evaluation is related to the comparison between received and predicted mobility data by means of a given mobility model. The mobility estimator consists of a dedicated Kalman filter. Its architecture and embedment into the framework are detailed. Discontinuities in received traces due to changing pseudonyms or shadowing effects are addressed and resolved.

### 4.1 Vehicle Tracking Based on Kalman Filtering

In this framework for mobility data verification we choose a Kalman filter [20] based approach to predict mobility data, i.e., geographical position, speed, and heading, of adjacent vehicles. Especially for object tracking, a Kalman filter represents an efficient and well-known solution [21]. Furthermore, as shown in [22], a Kalman filter based vehicle tracker seems to easily overcome pseudonym changes and therefore it is most appropriate for our purpose.

In the following we first give a brief introduction into Kalman filter theory. Then, we describe how the filter is to be adapted for the purpose of vehicle tracking.

**Kalman Filter Description**

To predict the state of a linear system, a Kalman filter repeats two successive phases for every time step $k$.

The first phase is the Prediction, whereby a prediction $\hat{x}_k$ of the *system state* is calculated by multiplying the last predicted state $\hat{x}_{k-1}^{+}$ with the *state transition matrix* $F_k$. The state transition matrix is the mathematical representation of the underlying system model.

$$\hat{x}_k = F_k \cdot \hat{x}_{k-1}^{+} \tag{1}$$

However, to get a more accurate prediction, Kalman filter provides the possibility to add additional *control values* $u_k$, via a *control matrix* $B_k$, to the system state before the state transition matrix is applied. This way, further information can be inserted, which are not part of the system model.

$$\hat{x}_{k-1}^{+} = \hat{x}_{k-1}^{+} + B_k \cdot u_k \tag{2}$$

In addition, a *prediction error* $P_k$ is calculated based on the transition matrix, the last calculated prediction error, and the *system fault matrix* $Q_k$. This system fault matrix represents errors that are inherent in the used system model.

$$P_k = F_k \cdot P_{k-1}^{+} \cdot F_k^T + Q_k \tag{3}$$

The prediction phase is followed by the second phase, the Correction. The predicted state is then corrected to achieve a more accurate system state by means of measurement values. Therefore, the *difference* $\Delta y_k$ between measured *measurement values* $\tilde{y}_k$ and predicted measurement values is calculated first. Based on the current system state, predicted measurement values are transformed by applying a *measurement matrix* $H_k$ to the system state.

$$\Delta y_k = \tilde{y}_k - H_k \cdot \hat{x}_k \tag{4}$$

Additionally, the *Kalman gain* $K_k$ is calculated based on the prediction error, taking into account *measurement variances* $R_k$.

$$K_k = P_k \cdot H_k^T \cdot (H_k \cdot P_k \cdot H_k^T + R_k)^{-1} \tag{5}$$

Now, the corrected system state $\hat{x}_k^+$ is established by weighting the difference $\Delta y_k$ with the Kalman gain and adding it to the system state.

$$\hat{x}_k^+ = \hat{x}_k + K_k \cdot \Delta y_k \tag{6}$$

Finally, the prediction error is corrected as well to achieve a more accurate prediction error $P_k^+$.

$$P_k^+ = P_k - K_k \cdot H_k \cdot P_k \tag{7}$$

Corrected system state and prediction error are used in the succeeding prediction phase at time step $k+1$.

The following schematic in Figure 3 illustrates the Kalman filter phases. Thereby, $z^{-1}$ denotes the time shift between step $k-1$ and $k$, respectively.

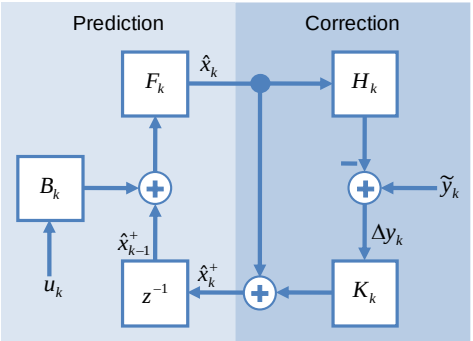


**Figure 3:** Kalman filter structure

**Applying Kalman Filter as a Vehicle Tracker**

As already mentioned, in C2X communication every message contains mobility information in the form of position, speed, and heading. According to [10] position is given in WGS84 co-ordinates, speed in m/s, and heading in degrees from north (clockwise). However, applying these data as state and measurement vectors will result in rather complex matrix calculations (especially of the position, as it is given in spherical co-ordinates). For reasons of efficiency, we decided to transform these mobility data as follows.

The position is converted into the Universal Transverse Mercator (UTM) co-ordinates system, which provides a two dimensional plane with an orthonormal basis. UTM co-ordinates are denoted as northing (y-axis) and easting (x-axis) in meters.

Speed and heading are combined and converted into speed related to each of these axes in m/s.

For our purpose, the state vector of the Kalman filter therefore consists of the vehicle's position $(p_x, p_y)$ and speed $(v_x, v_y)$ in the xy-plane.

$$\hat{x}_k = \begin{pmatrix} p_x \\ p_y \\ v_x \\ v_y \end{pmatrix} \tag{8}$$

To predict position and speed we now have to apply a vehicle mobility model, which is based on the *equation of linear motion*. Thereby, $\Delta t_k$ is the time difference to time step $k-1$, $p_k$ is the position, $v_k$ is the speed, and $a_k$ denotes the acceleration each at time point $k$.

$$p_k = p_{k-1} + v_{k-1} \cdot \Delta t_k + a_{k-1} \cdot \frac{{\Delta t_k}^2}{2} \tag{9}$$

According to this equation and the form of the chosen system state, the state transition matrix $F_k$ results in a four times four matrix.

$$F_k = \begin{pmatrix} 1 & 0 & \Delta t_k & 0 \\ 0 & 1 & 0 & \Delta t_k \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{10}$$

We add acceleration in x- and y-direction as a control factor $u_k$ to the system state.

$$u_k = \begin{pmatrix} a_x \\ a_y \end{pmatrix} \tag{11}$$

Since acceleration is not transmitted in C2X messages, we calculate its value from speed differences of the last received messages. Due to the fact that the acceleration is assumed to be constant within each time step, it is added with an according factor to the respective speed entries with the control matrix $B_k$ before $F_k$ is applied.

$$B_k = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ \frac{{\Delta t_k}^2}{2} & 0 \\ 0 & \frac{{\Delta t_k}^2}{2} \end{pmatrix} \tag{12}$$

As discussed in section 2.1, CAMs are sent in variable intervals, i.e., between 100ms and 1000ms. Therefore, $\Delta t_k$ cannot be assumed to be a constant.

As the measurement input $\tilde{y}_k$, used to correct our prediction, we take the delivered information present in received C2X messages. The contained position, speed, and heading are converted on reception of these messages, just like described for the system state.

$$\tilde{y}_k = \begin{pmatrix} p_x \\ p_y \\ v_x \\ v_y \end{pmatrix} \tag{13}$$

Accordingly, the state and the measurement vector are of identical form, the measurement matrix $H_k$ consists of the identity matrix only and, therefore, equations (4), (5), and (7) can significantly be simplified by eliminating $H_k$.

Since vehicle mobility prediction heavily depends on road scenarios, the system fault matrix $Q_k$ is chosen dynamically according to the road type. In analogy, the measurement variances matrix $R_k$ is chosen dynamically from current GPS accuracy delivered in related CAMs. Based on the described adoptions and chosen matrices the Kalman filter now can be used as a vehicle tracker in the proposed Mobility Data Verification Framework.

## 4.2 Mobility Verification Framework

Evaluation is performed upon every received C2X message in a serial order. The mobility data as well as the sender ID are extracted and handed over to the Mobility Verification Framework. After processing in this framework each message will be classified as *Approved*, *Neutral*, or *Erroneous*.

The first step of the verification consists of several threshold checks. Threshold verification prevents inconsistent data to corrupt the on-going mobility prediction. We intend to filter mobility data, which exceed some physical boundaries. For example, due to physics even in highway scenarios vehicles may only drive with a maximum speed. Furthermore, a message which indicates a position outside the host vehicles communication range is regarded as untrustworthy. In order to set the received mobility information into the context of the own vehicles movement, own GPS position information, velocity, and heading are updated regularly in the mobility verification framework. A timestamp check is applied to filter messages whose timestamp is either expired or dated to a future point in time. Another threshold check monitors the repetition frequency of CAM messages. To prevent *Denial-of-Service* attacks CAM messages, which are sent with higher frequency as defined in [10] by ETSI, are discarded. As those checks are very basic, we require each of them to be passed successfully to continue verification. If one of these checks fails, the entire message is marked as *Erroneous* (see D1 in Figure 4). In the following procedure the message must undergo a more in-depth analysis.

The framework now evaluates the message with respect to previously sent messages from that vehicle, i.e., it is observed if all message lie on a continuous trace. For this reason a tracker based on Kalman filter prediction is instantiated and maintained for every known vehicle within communication range.

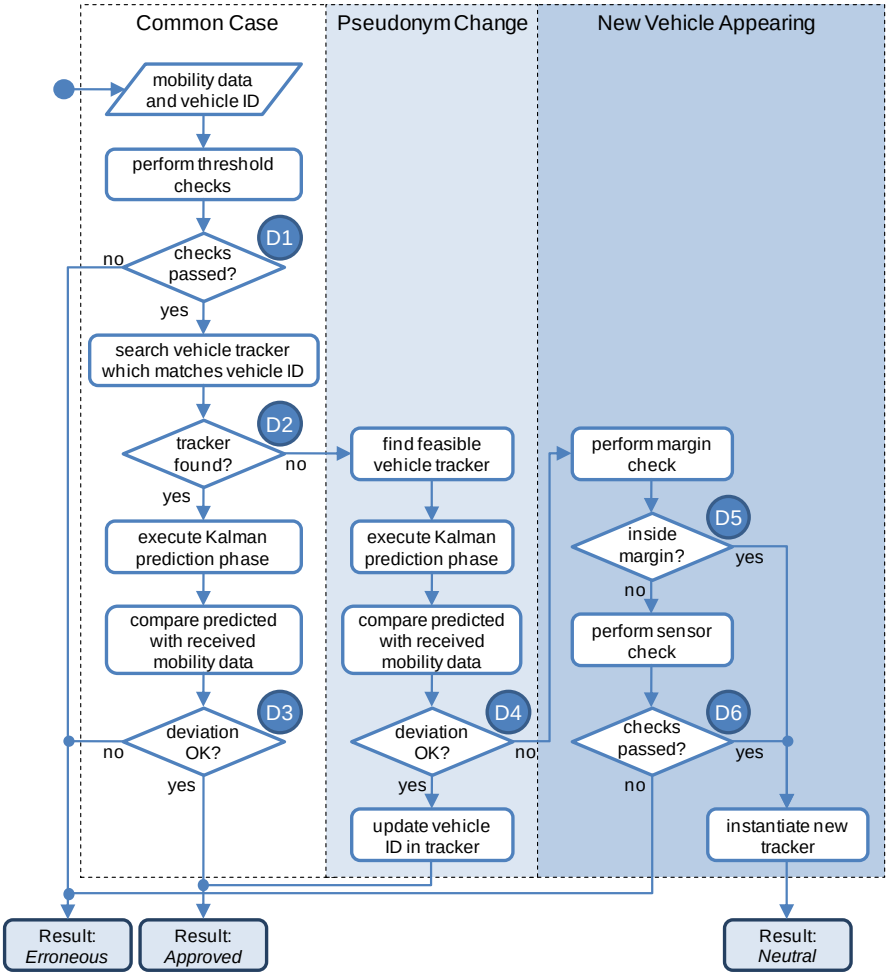


**Figure 4:** Mobility Data Verification Framework

Assuming that the framework receives an already known vehicle ID, the assigned tracker is used to compare the received mobility data to the deployed mobility model (see D2 in Figure 4). Based on the given timestamp inside the message the expected mobility data is predicted by triggering the Kalman prediction phase. During the Kalman filter correction phase the difference $\Delta y_k$ between predicted state $\hat{x}_k$ and received mobility data $\tilde{y}_k$ is calculated. Considering a maximum tolerable difference, the trustworthiness of the message is assessed. Thus, it may be evaluated as *Erroneous* or *Approved* (see D3 in Figure 4).

If no tracker was found, two possible reasons can be identified (D2 in Figure 4): Either an unknown vehicle is entering the host vehicle communication range, or, instead, an already known vehicle has performed a pseudonym change. A pseudonym change is made locally transparent by iterating the tracker list in order to find the candidate which is most likely to fit the received mobility data. For the most feasible tracker a prediction and correction phase of the Kalman filter is executed and the deviation is determined. If the vehicle movement fits the prediction of this tracker, then the message is evaluated as *Approved* and a pseudonym change is considered to be detected (see D4 in Figure 4). Consequently, a local *node ID* is assigned, which is matching both message IDs to a unique identifier, accessible and uniformly usable by other applications.

For detecting a new vehicle entering the communication range, a two-stage verification process is foreseen. The first stage is rather light-weighted as it is based on the simple assumption that vehicles generally first appear on the border of the current communication range $r_{max}$. In Figure 5, the tolerance margin of this check is illustrated. Accordingly, only messages indicating a vehicle appearing within the margin $r_{max} - d_{margin}$ and $r_{max}$ are marked as *Approved*.

However, starting vehicles may suddenly appear nearby the host vehicle. In order to exclude them from this *Acceptance Margin Range Check*, a minimum velocity is taken into account.

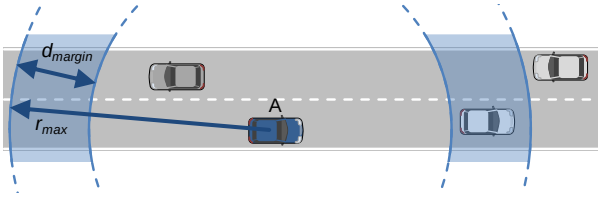


**Figure 5:** Acceptance margin range for appearing vehicles

Nevertheless, a fail of the Acceptance Margin Range check does not necessarily indicate an attacker. For instance, in Figure 6 a highway scenario is illustrated, where a vehicle A is starting to overtake the truck in front of it. Due to shadowing effects, caused by the truck in the rear, the CAM messages sent by vehicle A are being blocked to the approaching vehicle B. In consequence, vehicle B will evaluate the first message received from A as *Erroneous*, as it cannot distinguish this scenario from the one described in Figure 1. We therefore propose to perform complementary checks based on a vehicle's local sensors to overcome such difficulties.

For the prototypical implementation we used Radar with a total detection range of about 200m. The Radar constantly measures the distance and angle of vehicles driving ahead. Matching Radar objects to positions indicated by C2X messages is a non-trivial task when taking into account the different coordinate planes as well as error variances of both systems. In the scope of this work we apply an approach where the distance of an object measured by Radar is transferred into an absolute position with respect to the host's vehicle position by considering synchronization of the time. In case of a high update frequency of the own vehicle position and Radar data, it is sufficient to use a tolerance area in the Radar object detection. Only in case that the resulting Radar position can be matched to the C2X message's position, the message is evaluated as *Approved*. In this comparison of positions, the vehicle's dimensions are considered that are extracted from the C2X message. Note, that the local sensor check can be applied only for vehicles in *Line of Sight* (LOS). However, most of the safety related use cases perform their actions on vehicles in LOS.

For any vehicle appearing in non-LOS but within the near communication range, i.e., less than $r_{max} - d_{margin}$, the first message received is still evaluated as *Erroneous*.

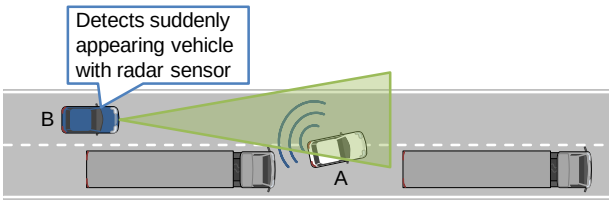


**Figure 6:** Acceptance Local Sensor Check for appearing vehicles

The possible results of the mobility verification framework are summarized in Table 1. Three validation classes are provided which can easily be interpreted and used by applications on the corresponding vehicle.

**Table 1:** Message Validation Classes

| Validation Class | Description |
|---|---|
| *Erroneous* | The mobility data does not match the mobility model of the verification framework. |
| *Neutral* | The framework cannot make a reliable and meaningful statement. |
| *Approved* | Mobility data of the message was checked and accepted. |

## 5. Evaluation

This Mobility Data Verification Framework has been evaluated on the sim$^{TD}$ platform. Therefore, it is implemented in Java/OSGi and tested by means of test drives with up to three vehicles and multiple recorded real word traces. Additionally, the Mobility Data Verification Framework will be extensively evaluated in upcoming large scale field operational trial sim$^{TD}$, without supplementary Local Sensor Check.

To determine prediction accuracy, we evaluate multiple test drives in urban, country road, and highway scenarios. Thereby, the applied CAM frequency is dynamic according to ETSI [10]. Figure 9 depicts one exemplary route of these test drives, which covers several different road classes.

In order to compare prediction accuracies at static and dynamic CAM frequencies, real world traces are recorded with 10Hz frequency. In the replay, the framework is supplied with messages at frequencies of 1Hz, 2Hz, 10Hz, and the dynamic frequency generated out of these traces by applying the corresponding CAM generation algorithm to them.

As depicted in Figure 7, the replaying results show that the prediction accuracy of the advocated Kalman filter-based vehicle tracker is best at the highest CAM frequency. Also with variable CAM intervals according to ETSI, the prediction error of 0.9m in 95% of the cases is significantly lower than the GPS accuracy.

Furthermore, we evaluated the effect of different road classes on the prediction error. Therefore, we compare test drives on highways and cities, each with CAM intervals according to ETSI. As shown in Figure 8, less predictable vehicle movement in urban scenarios has some effect on the prediction accuracy. However, even in city traces the position error is still negligibly low, but in special situations, e.g., a vehicle performs a full break or suddenly starts to overtake another vehicle, the prediction accuracy may decrease.

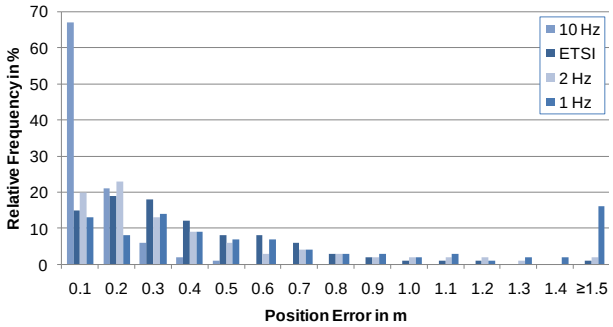


**Figure 7:** Comparison of measured position errors depending on different message frequencies

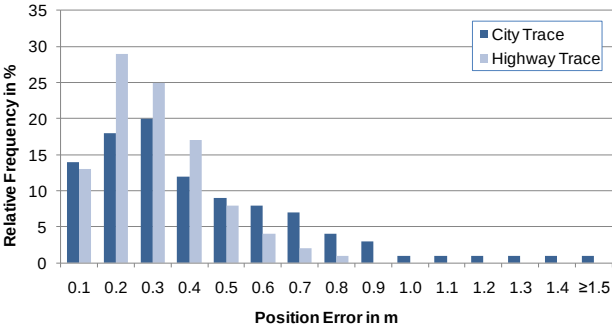


**Figure 8:** Comparison of position errors depending on road types measured at ETSI message frequency

Especially for safety critical use cases, e.g., Intersection Collision Warning, the message latency has to be as low as possible. Therefore, an efficient mobility data verification is needed, so that messages are not delayed more than necessary.

During multiple tests, as described above, an average overall message latency of about 2.7ms was achieved on the sim$^{TD}$ field operational test hardware. Thereby, function call, quick checks, and evaluation takes about 1.0ms, whereas Kalman filter prediction and correction take the remaining 1.7ms.

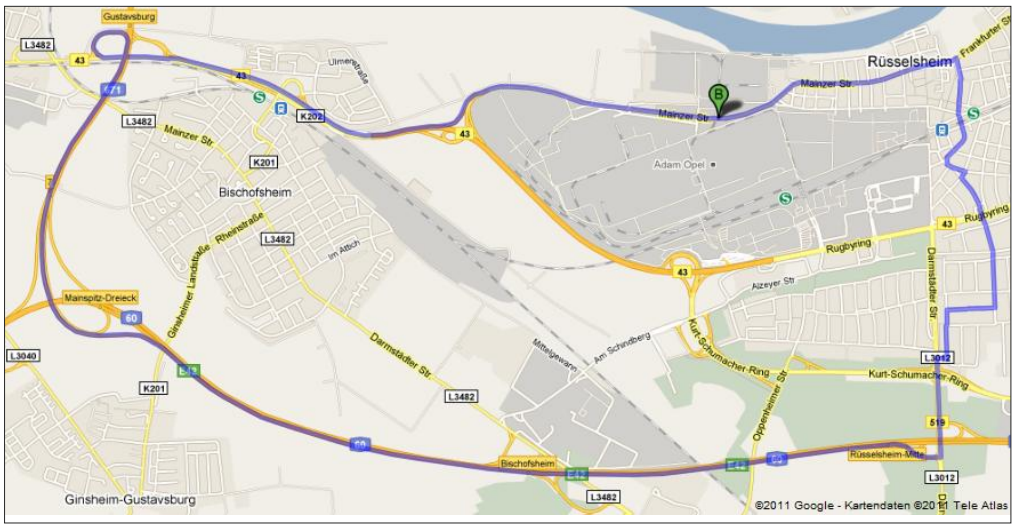


**Figure 9:** Exemplary route for one test drive, covering different road types

## 6. C2X-Architecture Integration

In Figure 10 the architectural integration in terms of involved components as well as the communication flow between them is illustrated. Accordingly, the message is parsed by lower communication layers and is handed over to the network layer for verification. In the assumed C2X architecture two concurrently running verification strategies are applied. The message contents with included mobility data is evaluated by the previously described Mobility Verification Framework as *Approved*, *Neutral* or *Erroneous*. In contrast, the cryptographic module is verifying signatures and certificates of messages to ensure integrity and sender authenticity, respectively. The returned result of the digital signature verification is binary, i.e., either the message can be verified or not. Despite the difference in evaluation goals of both verification strategies a central component will have to draw the decision upon forwarding of messages to the applications. Our general objective is to accumulate all security information within the message evaluation component and to forward only those messages, whose trustworthiness can be ensured by both components.

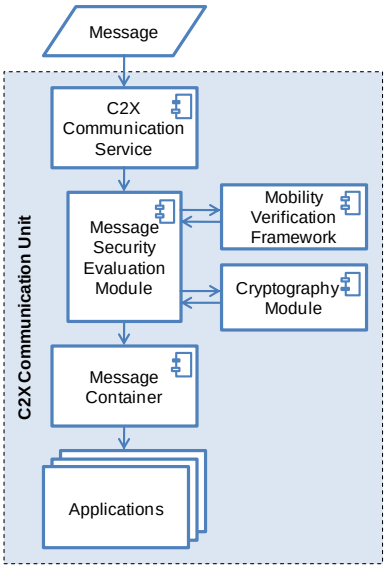


**Figure 10:** Architecture integration and message flow

To reduce communication overhead and delay we anticipate a case-sensitive execution of the cryptographic module and the mobility verification framework. We argue that a message, which fails the proposed mobility verification, will be of very low relevance for further applications, even if it was sent by an authenticated sender. Consequently, we propose to execute first the rather light-weighted mobility check and, if those checks have been passed, then to execute the more resource and time consuming signature verification. Thus, messages with non-plausible mobility content are not forwarded to the cryptographic module and are marked as *Erroneous*. A message, whose signature cannot be verified, is evaluated as untrustworthy and is not forwarded to the message container.

Please note that the architecture integration as depicted in Figure 10 is primary aimed towards periodically sent CAMs. As these messages need to be trustworthy in order to set up a reliable neighborhood table, we require security evaluation to be integrated as a mandatory layer within the communication stack. However, for application dependent messages, e.g., Black Ice warnings, IEEE 1609.2 foresees verification on application layer, which enables Verification-on-Demand schemes as proposed in [23]. For those messages security is much more to be seen as a service rather than a layer and, consequently, will require different integration strategies.

## 7. Conclusion and Future Work

Based on the assumption that a cryptographic protection of the C2X communication is not sufficient especially in the deployment phase, an efficient mobility verification framework is proposed that detects bogus mobility information in C2X messages produced by attackers or inaccurate sensors. This represents an attempt to fulfil IEEE 1609.2 requirement to verify mobility data in messages. Using the proposed model on both IVS and IRS enhance the protection of applications using only Car-to-Car communication or, additionally, Car-to-Infrastructure communication.

In order to avoid false-positive detections, especially in safety critical situations, we integrated sensor based position verification. The tracking algorithm approach detects pseudonym changes in the communication range and makes them transparent to the applications. As this identifier never leaves the vehicle AU, privacy is still preserved. In further implementations this information has to be securely protected by tamper proof devices in order to deny access to possible adversaries.

It has been shown in section 5 that the accuracy of position prediction is very high and, at the same time, the processing overhead is acceptable. More important, the scalability of the mobility verification framework has been demonstrated to be constant with up to 100 vehicles sending regularly CAM messages.

In future work it is reasonable to address and to integrate additional information sources such as different sensors, maps, and past vehicle behaviour. A parallel verification of mobility data on different layers followed by a sophisticated aggregation should be considered in future implementations too. In order to get realistic evaluation results, future enhancements should also be based on the automotive hardware and software components being already used in the field operational tests. Additionally, refinements on the implementation have to be done to overcome inaccuracies in full break situations or sharp overtaking maneuvers. There, context depending acceptance thresholds may be introduced in future work.

## References

[1] ETSI ES 202 663, "European profile standard for the physical and medium access control layer of intelligent transport systems operating in the 5 GHz frequency band," in *European Telecommunications Standards Institute*, Sophia Antipolis, France, 2010, p. 27.

[2] C. Maihöfer, "A Survey of Geocast Routing Protocols," *IEEE Communications Surveys & Tutorials*, vol. 6, no. 2, pp. 32-42, 2004.

[3] Intelligent Transportation Systems Committee, "IEEE Trial-Use Standard for Wireless Access in Vehicular Environments - Security Services for Applications and Management Messages," IEEE Vehicular Technology Society Standard 1609.2™-2006, 2006.

[4] O. Henninger, et al., "Security Requirements for Automotive On-board Networks," in *9th International Conference on Intelligent Transport System Telecommunications (ITST 2009)*, Lille, France, 2009.

[5] IntelliDrive. (2010) Enabling Aftermarket Devices with DSRC-Based Communications Capabilities: Summary of Input from Industry Stakeholders.

[6] C. Tarnovsky. (2010, Feb.) Blackhat DC 2010. [Online]. http://www.blackhat.com/html/bh-dc-10/bh-dc-10-briefings.html#Tarnovsky

[7] K. Koscher, A. Czeskis, F. Roesner, S. Patel, and T. Kohno, "Experimental Security Analysis of a Modern Automobile," in *IEEE Symposium on Security and Privacy*, Oakland, California, USA, 2010.

[8] H. Stübing, A. Jaeger, N. Bißmeyer, C. Schmidt, and S. A. Huss, "Verifying Mobility Data under Privacy Considerations in V2X Communication," in *17th ITS Wolrd Congress*, Busan, Korea, 2010.

[9] N. Bißmeyer, H. Stübing, M. Mattheß, J. P. Stotz, J. Schütte, M. Gerlach, and F. Friederici, "simTD Security Architecture," in *Embedded Security in Cars Conference (escar)*, Düsseldorf, Germany, 2009.

[10] European Telecommunications Standards Institute (ETSI), "Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Cooperative Awareness Basic Service," ETSI Technical Specification ETSI TS 102 637-2, 2010.

[11] M. Torrent-Moreno, P. Santi, and H. Hartenstein, "Distributed Fair Transmit Power Adjustment for Vehicular Ad Hoc Networks," in *IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, Reston, USA, 2006.

[12] CAR 2 CAR Communication Consortium, "C2C-CC Manifesto," CAR 2 CAR Communication Consortium, Report, 2007. [Online]. www.car-to-car.org

[13] T. Leinmüller, R. K. Schmidt, E. Schoch, A. Held, and G. Schäfer, "Modeling Roadside Attacker Behavior in VANETs," in *IEEE Workshop on Automotive Networking and Applications (AutoNet)*, New Orleans, USA, 2008.

[14] T. Leinmüller, E. Schoch, and F. Kargl, "Position Verification Approaches for Vehicular Ad Hoc Networks," *IEEE Wireless Communications*, Oct. 2006, VEBAS.

[15] R. K. Schmidt, T. Leinmüller, E. Schoch, A. Held, and G. Schäfer, "Vehicle Behavior Analysis to Enhance Security in VANETs," in *Workshop on Vehicle to Vehicle Communications (V2VCOM)*, Eindhoven, the Netherlands, 2008.

[16] G. Yan, G. Choudhary, M. C. Weigle, and S. Olariu, "Providing VANET Security Through Active Position Detection," in *VANET'07*, Montréal, Québec, Canada, 2007.

[17] B. Xiao, B. Yu, and C. Gao, "Detection and Localization of Sybil Nodes in VANETs," in *DIWANS '06*, Los Angeles, CA, USA, 2006.

[18] C. Laurendeau and M. Barbeau, "Probabilistic Localization and Tracking of Malicious Insiders Using Hyperbolic Position Bounding in Vehicular Networks," *EURASIP Journal on Wireless Communications and Networking - Special issue on wireless network security*, no. 2009, Feb. 2009.

[19] Z. Ren, W. Li, Q. Yang, S. Wu, and L. Chen, "Location Security in Geographic Ad hoc Routing for VANETs," in *ICUMT '09*, St. Petersburg, 2009.

[20] R. Kalman, "A new Approach to Linear Filtering and Prediction Problems," *Transactions of the ASME-Journal of Basic Engineering*, 1960.

[21] S. Blackman and R. Popoli, *Design and Analysis of Modern Tracking Systems*. Artech House Publishers, 1999.

[22] B. Wiedersheim, F. Kargl, Z. Ma, and P. Papadimitratos, "Privacy in Inter-Vehicular Networks: Why simple pseudonym change is not enough," in *7th International Conference on Wireless On-demand Network Systems and Services (WONS 2010)*, Kranjska Gora, Slovenia, 2010.

[23] H. Krishnan. (2008, Oct.) Verify-on-Demand - A Practical and Scalable Approach for for Broadcast Authentication in Vehicle Safety Communication. www.ip.com, IP.com number: IPCOM000175512D, IP.com Electronic Publication.

**Attila Jaeger** studied Computer Science at the Technische Universität Darmstadt, Germany and received his Diploma (Dipl.-Inform.) in October 2008. Since February 2009, he is engaged as Research Assistant at the Integrated Circuits and Systems Lab, Department of Computer Science, Technische Universität Darmstadt, Germany. In the context of Car-to-X communication, he is researching on Weather Hazard Warning application, system architectures, privacy aspects, vehicle tracking, and solutions for security by elliptic curve cryptography.

**Norbert Bißmeyer** studied Applied Computer Science at the FH Münster, Germany and received his Bachelor's degree in 2006. Afterwards he studied Advanced Security Engineering at the FH Joanneum in Austria and Ireland and received his Master's degree in 2008. Since November 2008 he is working at the Fraunhofer Institute for Secure Information Technology in

Darmstadt, Germany in the department Secure Mobile Systems. He is working in the field of vehicular ad hoc networks with focus on security and privacy concepts. Misbehavior detection with appropriate response mechanisms in decentralized ITS commutation is the primary research topic of Norbert Bißmeyer.

**Hagen Stübing** is a research engineer in the Advanced Engineering Active Safety Department at the Adam Opel AG. Prior to joining Opel, he was studying Electrical Engineering at the Technische Universität Darmstadt, Germany with emphasis on embedded system design. In 2004 he joined a double degree program with the Universitat Politècnica de Catalunya in Barcelona, Spain from where he received his Masters Degree in Information and Communication Technologies (M.Sc.) in 2006. He completed his Masters Degree in Electrical Engineering (Dipl.-Ing.) in 2008. Since July 2008 he is doing his Ph.D. at Adam Opel AG in the field of vehicular ad hoc networks. In particular his research interests are protection techniques for security and privacy issues as well as Car-to-X architectures in general.

**Sorin A. Huss** received the Dipl.-Ing. and Dr.-Ing. degrees in electrical engineering from the Technische Universität München, Germany, in 1976 and 1982, respectively. He worked from 1982 until 1990 with AEG Aktiengesellschaft in Ulm, Germany, as the CAD/CAE manager of the AEG Integrated Circuits Design Center. Since 1990, he has been a full professor in the Computer Science Department of the Technische Universität Darmstadt, Germany, and also a faculty member of the Electrical Engineering Department. In addition, Dr. Huss is a vice-president of the CASED Center for Advanced Security Research Darmstadt heading the "Secure Things" research group. He authored or coauthored two books, several book chapters, and more than 200 reviewed journal and conference papers; he received several Best Paper Awards. His current research interests are in the areas of embedded systems and reconfigurable HW/SW architectures aimed for IT security applications. He is a member of the ACM, the IEEE, the German Computer Science Association (GI), and the German Information Technology Society (ITG). He was the general chair of the FDL'06 and the COSADE'10 Conferences and served as a member of many conference program committees and editorial boards.