An Extension Proposal for the Collective Perception Service to Avoid Transformation Errors and Include Object Predictions*

Jan Strohbeck, Martin Herrmann, Johannes Müller, Michael Buchholz

Abstract— **The collective perception service, which is in progress of standardization by the European Telecommunication Standards Institute, allows to share perception information among connected vehicles and road side units and thus can increase both safety and traffic efficiency. However, based on our practical experience from our research on infrastructure support of automated vehicles on a pilot installation in real traffic, in this work, we outline some drawbacks of the existing draft when applied to real-world environments. We observe that the strict cartesian representation does not fit well with typical models used to predict the motion of vehicles in automated driving applications. Thus, a transformation overhead as well as a transformation error builds up that increases the uncertainty about the perceived objects and, thus, the performance of the collective perception service at all. In this work, we demonstrate the effect of such transformation errors using simulations and propose an extension of the standard to circumvent these. Additionally, we show that the collective perception service can further be enhanced, allowing the optional transmission of motion predictions of perceived objects. This is, receivers benefit from saving computation time for object predictions and from the reception of high-quality motion predictions from road side units that are more accurate due to their knowledge of local peculiarities.**

I. INTRODUCTION

Environment perception systems that allow the detection, classification, and tracking of dynamic traffic participants are a key module of advanced driver assistance (ADAS) systems. The improvement of these systems is seen as a key enabler of completely autonomous vehicles [1]. However, there are many circumstances that restrict the performance of such systems, like sensory limits, occlusions, weather conditions, or questions of cost. To overcome these limitations, the idea of collaborative perception has been investigated and a concept of collaborative perception was already introduced in 2004 [2]. Although sensor manufacturers achieved tremendous improvements for the perception on-board a vehicle, the idea of sharing information about perceived objects among connected vehicles and RSUs remains popular and is ascribed a high potential based on simulation studies [3], [4]. Together with our partners within the projects MEC-View (`www.mec-view.de`) and ICT4CART (`www.ict4cart.eu`), we extended this idea to infrastructure support of automated vehicles using infrastructure sensors and a centralized local environment model generation on a low-latency multi-access edge computing (MEC) server [5], [6]. For practical evaluation, we operate a pilot installation on public roads in Ulm, Germany, where we use the AVs' handling of occlusions when merging on a priority road at a T-junction as an exemplary use case for our research [5]. For this, we have developed and implemented an infrastructure environment model generation [7] together with a prediction of the environment model [8], which counteracts latency and additionally allows for predictive planning [9].

To enable the exchange of such environment model information, the European Telecommunication Standards Institute (ETSI) is developing a standard for the Collective Perception Service (CPS) [10], which is accompanied by active research, e.g. [11]–[14]. Based on publicly available information [15], this service has its own message type, the Collective Perception Message (CPM) [16], which can encode information about perceived objects as well as a transmitter's state, sensor setup, detection accuracy, and position. Using information exchange via the CPS, connected vehicles can enlarge their effective field of view and/or improve the accuracy of their environment model via information fusion methods.

In the currently published technical report about the CPS [15], the CPM represents an object via the *PerceivedObject* data type, which basically consists of a state vector and confidences for each state variable. The state vector includes, among others, the position, velocity v_x, v_y, v_z, acceleration a_x, a_y, a_z, orientation φ, and yaw rate $\dot{\varphi}$, where $v_z, a_x, a_y, a_z, \varphi$ and $\dot{\varphi}$ are optional. The confidences encode how large a 95% confidence interval for the respective value is, which corresponds to 1.96 times the standard deviation for a Gaussian distributed variable. However, this confidence information determines only the diagonal of the covariance matrix of the state vector, i.e., all state variables are modeled as independent random variables. Real data, however, reveals that this assumption hardly corresponds to reality, which can introduce significant errors when information of different sources is fused using information fusion methods like covariance intersection.

To perform the information fusion correctly and efficiently, the CAR 2 CAR Communication Consortium (C2C-CC) has proposed changes to the work-in-progress message definition, which aim to also include covariance data along with each detected object [17]. Hence, each object state

∗ Part of this work has been conducted as part of ICT4CART project which has received funding from the European Union's Horizon 2020 research & innovation program under grant agreement No. 768953. Content reflects only the authors' view and European Commission is not responsible for any use that may be made of the information it contains. This work was financially supported by the Federal Ministry of Economic Affairs and Energy of Germany within the program "Highly and Fully Automated Driving in Demanding Driving Situations" (project LUKAS, grant number 19A20004F).

The authors are with the Institute of Measurement, Control and Microtechnology, Ulm University, D-89081 Ulm, Germany
{first name.family name@uni-ulm.de},johannes-christian.mueller@uni-ulm.de

is represented by the first two moments of its multivariate distribution, which fully specifies it in the case of normal distributions. The extension proposal of C2C-CC adds additional fields to the *PerceivedObject* data type that can encode the information of the off-diagonal entries of the covariance matrix as correlation values. Thus, the full covariance matrix can be recovered in combination with the original confidence information. Using this extension, reliable and accurate information fusion between an internal environment model of the receiving vehicle and an environment model that was shared via CPM becomes possible.

However, the strictly cartesian definitions used for representing velocities and accelerations in the CPM can lead to large transformation errors, as we show in Section II using simulated examples. Particularly, common tracking filters often use kinematic models from the constant turn-rate family [18], which need to be transformed to the cartesian frame to be represented by the current *PerceivedObject* data type definition. Unfortunately, these transformations are nonlinear, which leads to a different density representation. Then, a transformed density has to be approximated by its original type again, e.g. a normal density, which can introduce large errors that we denote as transformation errors. To avoid these, we propose a further change to the CPM definition.

Moreover, we present a possible extension of the CPM in Section III, which adds the possibility to encode path predictions for *PerceivedObjects*. Such predictions are commonly required by automated vehicles during motion planning [9] and can additionally account for latencies in the data processing and communication chain of a CPM. A stationary intelligent transportation system (ITS) station like an RSU is not limited to detect objects via connected sensors, but it can also predict the future trajectories of the respective road users using e.g. modern deep learning approaches such as [8], [19]. Due to their stationary nature, such RSU can potentially have a much deeper knowledge of their surroundings than the moving connected vehicles, and hence predict the behavior of local traffic participants more accurately. Also, performing the path prediction once on the RSU saves the receiving vehicles the effort and computational time required to perform the prediction themselves. However, the existing proposal of the CPM does not allow to encode object predictions. Therefore, we present a possible modification to the CPM definition to incorporate this possibility.

Lastly, in Section IV, we evaluate the impact of our modifications on the encoded size of *PerceivedObjects* in order to confirm that the channel usage does not increase inappropriately.

II. DISTRIBUTION TRANSFORMATION ERRORS

For multi-target tracking, commonly used kinematic models include the constant turn-rate and velocity (CTRV) and constant turn-rate and acceleration (CTRA) models [18]. With these, velocity and acceleration are measured in polar coordinates, where v_{pol} and a_{pol} encode the (possibly negative) magnitude in the direction of the orientation of the object (specified by the yaw angle φ), instead of using two-dimensional vectors in cartesian coordinates (i.e., $\underline{v} = [v_x, v_y]^T$, $\underline{a} = [a_x, a_y]^T$). That is, the parts of the densities that are relevant for the velocity and the acceleration are represented using a polar coordinate system, which is not a perfect representation, but yet has been shown to often outperform simpler models strictly relying on cartesian coordinates [18]. Turning maneuvers can be represented using the yaw rate $\dot{\varphi}$, but drifting can not be represented, as it is not modeled by CTRA/CTRV anyways. However, the current CPM standard is not designed for the transmission of polar values such as v_{pol}, which renders the mentioned transformation into cartesian values v_x, v_y necessary. This leads to the possible problems we show below. We only refer to velocity here, but the same reasoning applies to acceleration analogously.

Common tracking algorithms like the Kalman filter assume a multivariate normal probability distribution over all state features of an object [20]. The joint probability distribution over v_{pol} and φ is then $\mathcal{N}([\bar{v}_{\text{pol}}, \bar{\varphi}]^T, \Sigma_{v_{\text{pol}},\varphi})$, where $\bar{v}_{\text{pol}}$ and $\bar{\varphi}$ denote the mean state values and

$$\Sigma_{v_{\text{pol}},\varphi} = \begin{bmatrix} \text{cov}(v_{\text{pol}}, v_{\text{pol}}) & \text{cov}(v_{\text{pol}}, \varphi) \\ \text{cov}(\varphi, v_{\text{pol}}) & \text{cov}(\varphi, \varphi) \end{bmatrix},$$

denotes the covariance matrix. This bivariate distribution can be interpreted as an uncertainty ellipse in a polar coordinate frame, as shown in Fig. 1a, that describes the uncertainty about the velocity of an object. The example distribution in this example is $\mathcal{N}([\bar{v}_{\text{pol}}, \bar{\varphi}]^T, \Sigma_{v_{\text{pol}},\varphi})$ with

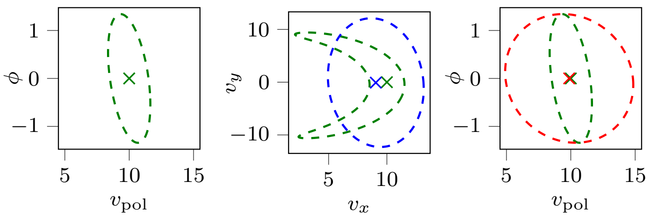
$$[\bar{v}_{\text{pol}}, \bar{\varphi}]^T = [10, 0]^T, \quad \Sigma_{v_{\text{pol}},\varphi} = \begin{bmatrix} 0.3 & -0.1 \\ -0.1 & 0.2 \end{bmatrix}.$$

However, the CPM definition requires velocities to be encoded in cartesian coordinates. Hence, the uncertainty about an object's velocity is described by $\mathcal{N}([\bar{v}_x, \bar{v}_y]^T, \Sigma_{v_x,v_y})$, with mean $[\bar{v}_x, \bar{v}_y]^T$ and covariance

$$\Sigma_{v_x,v_y} = \begin{bmatrix} \text{cov}(v_x, v_x) & \text{cov}(v_x, v_y) \\ \text{cov}(v_y, v_x) & \text{cov}(v_y, v_y) \end{bmatrix}.$$

This can be interpreted as an ellipse in a cartesian coordinate frame. The transformation of the polar covariance ellipse into a cartesian one is, however, generally not possible without loss of information, which, e.g., has been analyzed in [21].

One method of approximation is to draw a large number N of samples $[v^i_{\text{pol}}, \varphi^i]^T$, $i \in 1..N$ from the distribution $\mathcal{N}([\bar{v}_{\text{pol}}, \bar{\varphi}]^T, \Sigma_{v_{\text{pol}},\varphi})$, to transform them using the nonlinear function $[v^i_x, v^i_y]^T = [v^i_{\text{pol}} \cos(\varphi^i), v^i_{\text{pol}} \sin(\varphi^i)]^T$, and finally to estimate the statistics of the transformed samples to obtain $\mathcal{N}([\bar{v}_x, \bar{v}_y]^T, \Sigma_{v_x,v_y})$. A faster, yet potentially less accurate method is to use an unscented transformation, which effectively reduces the number of required sampling points, but necessitates the same statistics estimation step. Then, irrespective of the chosen method, an error results from the fact that the resulting statistics are interpreted as being normal distributed, but actually they are not. That is, the ellipse in the polar coordinates does not correspond to an ellipse in the cartesian coordinates, but becomes a different shape due to the nonlinear transformation, sometimes



(a) (b) (c)

Fig. 1. (a) shows an example bivariate normal distribution in polar coordinates as an ellipse of the 3σ area. (b) shows the 3σ area after the distribution has been transformed nonlinearly to cartesian space (green), and after the subsequent approximation as a bivariate normal distribution in cartesian coordinates (blue). When the approximated distribution is transformed back to polar coordinates and approximated again, the red distribution shown in (c) results. The original distribution is shown in green again as reference.

referred to as uncertainty banana. However, performing the statistics estimation to obtain a cartesian multivariate normal distribution for CPM transmission reshapes the uncertainty region into an ellipse, which then leads to an incorrect uncertainty estimate (see Fig. 1b).

While the mentioned errors are unavoidable in the case of mixed frames, they are unnecessary if the receiver reverses the transformation from a cartesian to a polar coordinate system, e.g., because a similar motion model is used. This inverse conversion can be achieved using the inverse nonlinear transformation function, but at the cost of additional approximation errors. The inverse transform, which accounts for the sign of the polar velocity, is given by

$$\tilde{\varphi}^i = ((\varphi^i - \operatorname{arctan2}(v_y^i, v_x^i) + \pi) \bmod 2\pi) - \pi$$

$$v_{\mathrm{pol}}^i = \begin{cases} -\sqrt{(v_x^i)^2 + (v_y^i)^2} & \text{if } |\tilde{\varphi}^i| > \frac{\pi}{2} \\ \sqrt{(v_x^i)^2 + (v_y^i)^2} & \text{otherwise.} \end{cases}$$

Finally, the distribution is reconstructed by a subsequent approximation of the statistics as a normal distribution, which again causes a transformation error. Fig. 1c shows the resulting distribution compared to the original distribution. As can be seen, the uncertainty about the state increased significantly only due to the transformations and approximations, which makes the reconstructed distribution less valuable.

The required transformations cause unnecessary inaccuracies in real-world applications, since constant turn-rate kinematic models are widely used in tracking models for automated driving. Our proposed solution is to allow the sender to choose between a cartesian and a polar representation for encoding velocity and acceleration (see Fig. 2). For the cartesian representation, we reuse the existing types from the CPM draft, while for the polar values, we suggest using new types with the shown precision. This renders the conversion in the encoding step unnecessary. Naturally, if the receiver uses a cartesian representation of velocities and accelerations in its environment model, a conversion from polar to cartesian is still unavoidable. But, the lossy conversions across the data transmission are, however, reduced to those that are really necessary.

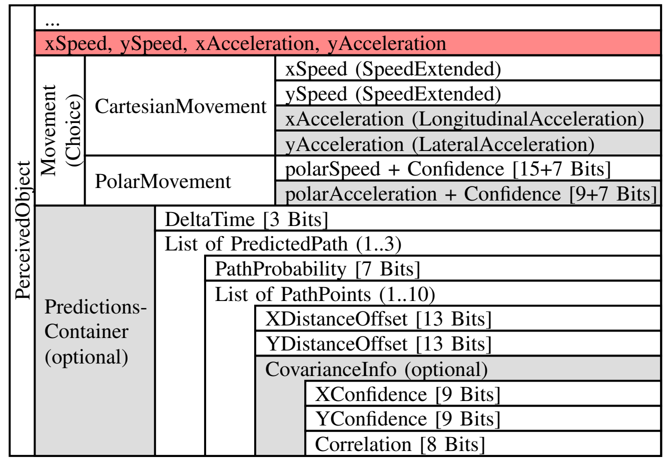


Fig. 2. Proposed modifications to the CPM definition from [15] enhanced with [17]. We use "…" to represent the (unmodified) fields, except for fields that we remove (red areas). Areas shown in grey denote optional fields.

III. PREDICTIONS

CPMs are intended to transmit information about perceived objects between ITS stations. Stationary ITS stations can use tailored algorithms for the prediction of the future motion of detected traffic participants, which is potentially better than one in a moving ITS station that has less prior knowledge of the surveilled area and has to use a more generic method. Thus, we propose to allow the optional transmission of motion predictions via CPMs, which is shown in Fig. 2.

For the encoding, we propose to allow a maximum of three predicted paths per perceived object, where each should be assigned a probability, i.e., how likely it is that the respective path will be taken. Providing multiple predictions per object is important, since many situations, e.g., intersections, can introduce multimodality in the possible driving behaviors.

For the representation of a predicted path, we suggest encoding of up to 10 points per path. A point consists of an $[x, y]$ pair of predicted coordinates, as well as an optional covariance matrix $\Sigma_{x,y}$, as it is generated by many modern deep trajectory prediction models like [8], [19]. Such neural networks generate output that describes the distribution of the predicted path points, e.g., a bivariate normal distribution for each path point. The covariance matrices can be used to further assess the accuracy of the predicted path, which is important, e.g., for calculating risks of collision [9].

For encoding this covariance data, we suggest using the same format as proposed in [17], i.e., encoding the variances using confidence values and an additional value for x-y-correlation information. The path points shall be equally spaced in time, and the path delta time shall be configurable as a parameter with a restricted range of $[100\,\mathrm{ms}, 500\,\mathrm{ms}]$ in steps of $100\,\mathrm{ms}$, which allows for both short-term predictions as well as long-term predictions. For example, for latency compensation, one second in steps of $100\,\mathrm{ms}$ could be encoded, or for predictive planning, up to five seconds using steps of $500\,\mathrm{ms}$ could be sent.

TABLE I

EVALUATION OF THE MAXIMUM ENCODED SIZE PER OBJECT.

Configuration	*CovarianceInfo*	polar/cartesian	Encoded size per *PerceivedObject*
No predictions	—	cartesian	763 Bits
1 Prediction	no	cartesian	1026 Bits
1 Prediction	yes	cartesian	1286 Bits
3 Predictions	no	cartesian	1546 Bits
3 Predictions	yes	cartesian	2326 Bits
No predictions	—	polar	557 Bits
1 Prediction	no	polar	820 Bits
1 Prediction	yes	polar	1080 Bits
3 Predictions	no	polar	1340 Bits
3 Predictions	yes	polar	2120 Bits

IV. MESSAGE ENCODING

To evaluate the impact of our proposal on the encoded size per perceived object, we calculate the size for several cases: sending no, one, or three predictions as well as with or without covariance information for the predictions. We also compare the size when using polar velocity/acceleration values versus their cartesian counterparts and present the results in Table I. For polar data, we use a state comprised of position, $v_{\text{pol}}, a_{\text{pol}}, \varphi, \dot{\varphi}, l, w, h$, where l, w, h denote the object length, width and height. The baseline is the case with no predictions and with data that could be calculated from the polar data using the nonlinear transformations, i.e., a state comprised of position, $v_x, v_y, a_x, a_y, \varphi, \dot{\varphi}, l, w$ and h.

Using polar velocity and acceleration leads to slightly smaller sizes, since there are fewer state variables to transmit. The size of the correlation information that is encoded via the C2C-CC proposal increases also quadratically with the number of state variables, making the reduction very favorable. We can, however, see that encoding predictions has a considerable impact on the message size, which we justify by the possibly significant information gain due to the improved predictions. To reduce the impact on the channel usage, predictions could, for example, be sent at a lower transmission rate than object detections, especially in case of long-term predictions, since they can be useful for a longer period of time.

V. CONCLUSIONS

In this work, we showed that large transformation errors occur when encoding data from constant turn rate models into CPMs and back, which decreases the value of the data immensely. To alleviate this issue, we proposed a simple modification to the CPM definition. Further, we presented possible additions to the CPM definition in order to incorporate multimodal predictions. We have analyzed the impact of this additional data on the message size. Using the additional optional fields for the improved velocity and acceleration encoding by themselves does not increase the message size, but can rather decrease it in practice by omitting the cartesian values instead. Although the message size can increase noticeably with the predictions, we still think it can be reasonable to send them, because they can provide a huge benefit to the receiver.

REFERENCES

[1] H. Winner, S. Hakuli, F. Lotz, and C. Singer, Eds., *Handbook of Driver Assistance Systems*. Springer International Publishing, 2016.

[2] B. Mourllion, A. Lambert, D. Gruyer, and D. Aubert, "Collaborative perception for collision avoidance," in *2004 IEEE International Conference on Networking, Sensing and Control*, vol. 2, 2004, pp. 880–885.

[3] H.-J. Günther, O. Trauer, and L. Wolf, "The potential of collective perception in vehicular ad-hoc networks," in *2015 14th International Conference on ITS Telecommunications (ITST)*, 2015, pp. 1–5.

[4] K. Garlichs, M. Wegner, and L. C. Wolf, "Realizing collective perception in the artery simulation framework," in *2018 IEEE Vehicular Networking Conference (VNC)*, 2018, pp. 1–4.

[5] M. Buchholz, J. Müller, M. Herrmann, J. Strohbeck, B. Völz, M. Maier, J. Paczia, O. Stein, H. Rehborn, and R.-W. Henn, "Handling occlusions in automated driving using a MEC server-based environment model from infrastructure sensors," *IEEE Intelligent Transportation Systems Magazine*, 2021, accepted for publication.

[6] M. Buchholz, J. Strohbeck, A.-M. Adaktylos, F. Vogl, G. Allmer, S. C. Barros, Y. Lassoued, M. Wimmer, B. Hätty, G. Massot, C. Ponchel, M. Bretin, V. Sourlas, and A. Amditis, "Enabling automated driving by ict infrastructure: A reference architecture," in *Proceedings of 8th Transport Research Arena (TRA) 2020*, 2020.

[7] M. Herrmann, A. Piroli, J. Strohbeck, J. Müller, and M. Buchholz, "Lmb filter based tracking allowing for multiple hypotheses in object reference point association," in *2020 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, 2020, pp. 197–203.

[8] J. Strohbeck, V. Belagiannis, J. Müller, M. Schreiber, M. Herrmann, D. Wolf, and M. Buchholz, "Multiple trajectory prediction with deep temporal and spatial convolutional neural networks," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 1992–1998.

[9] J. Müller and M. Buchholz, "A risk and comfort optimizing motion planning scheme for merging scenarios," in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, 2019, pp. 3155–3161.

[10] *ETSI TS 103 561 - Intelligent Transport Systems (ITS); Cooperative Perception Services*, ETSI Early Draft Proposed Standard, 2021.

[11] H.-J. Günther, B. Mennenga, O. Trauer, R. Riebl, and L. Wolf, "Realizing collective perception in a vehicle," in *2016 IEEE Vehicular Networking Conference (VNC)*, 2016, pp. 1–8.

[12] H.-J. Günther, R. Riebl, L. Wolf, and C. Facchi, "Collective perception and decentralized congestion control in vehicular ad-hoc networks," in *2016 IEEE Vehicular Networking Conference (VNC)*, 2016, pp. 1–8.

[13] K. Garlichs, H.-J. Günther, and L. C. Wolf, "Generation rules for the collective perception service," in *2019 IEEE Vehicular Networking Conference (VNC)*, 2019, pp. 1–8.

[14] G. Thandavarayan, M. Sepulcre, and J. Gozalvez, "Analysis of message generation rules for collective perception in connected and automated driving," in *2019 IEEE Intelligent Vehicles Symposium (IV)*, 2019, pp. 134–139.

[15] *ETSI TR 103 562 - Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Analysis of the Collective Perception Service*, ETSI Technical Report, Rev. V2.1.1, 2019.

[16] A. Rauch, F. Klanner, and K. Dietmayer, "Analysis of v2x communication parameters for the development of a fusion architecture for cooperative perception systems," in *2011 IEEE Intelligent Vehicles Symposium (IV)*, 2011, pp. 685–690.

[17] "Technical report on cpm object quality," CAR 2 CAR Communication Consortium, Tech. Rep., 3 2021, version 1.0. [Online]. Available: https://www.car-2-car.org/fileadmin/documents/General_Documents/C2CCC_TR_2089_CPM_ObjectQuality_V1.pdf

[18] R. Schubert, E. Richter, and G. Wanielik, "Comparison and evaluation of advanced motion models for vehicle tracking," in *2008 11th Int. Conf. on Information Fusion*. IEEE, 2008, pp. 1–6.

[19] J. Li, F. Yang, M. Tomizuka, and C. Choi, "Evolvegraph: Multi-agent trajectory prediction with dynamic relational reasoning," in *Advances in Neural Information Processing Systems*, vol. 33. Curran Associates, Inc., 2020, pp. 19 783–19 794.

[20] Y. Shalom, X.-R. Li, and T. Kirubarajan, *Estimation with applications to tracking and navigation*. New York: Wiley, 2001.

[21] S. Julier, "The scaled unscented transformation," *Proceedings of the 2002 American Control Conference*, vol. 6, pp. 4555–4559, 2002.