



HAL
open science

Distributed asynchronous cooperative localization with inaccurate GNSS positions

Elwan Héry, Philippe Xu, Philippe Bonnifait

► **To cite this version:**

Elwan Héry, Philippe Xu, Philippe Bonnifait. Distributed asynchronous cooperative localization with inaccurate GNSS positions. 22nd IEEE International Conference on Intelligent Transportation Systems (ITSC 2019), Oct 2019, Auckland, New Zealand. pp.1857-1863. hal-02407395

HAL Id: hal-02407395

<https://hal.science/hal-02407395>

Submitted on 12 Dec 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed asynchronous cooperative localization with inaccurate GNSS positions

Elwan Héry¹, Philippe Xu¹ and Philippe Bonnifait¹

Abstract—Localization remains a major issue for autonomous vehicles. Accurate localization relative to the road and other vehicles is essential for many navigation tasks. When vehicles cooperate and exchange information through wireless communications, they can improve mutually their localization. This paper presents a distributed cooperative localization method based on the exchange of Local Dynamic Maps (LDMs). Every LDM contains dynamic information on the pose and kinematic of all the cooperating agents. Different sources of information such as dead-reckoning from the CAN bus, inaccurate (i.e. biased) GNSS positions, LiDAR and road border detections are merged using an asynchronous Kalman filter strategy. The LDMs received from the other vehicles are merged using a Covariance Intersection Filter to avoid data incest. Experimental results are evaluated on platooning scenarios. They show the importance of estimating GNSS biases and having accurate relative measurements to improve the absolute localization process. These results also illustrate that the relative localization between vehicles is improved in every LDMs even for vehicles not able to perceive surrounding vehicles but which are instead perceived by others.

I. INTRODUCTION

Cooperative localization is a well-known issue in mobile robotics. Many works describe this problem as a rigid structure [1] where the different robots are the nodes and the bearing [21] or range [20] measurements are the links. This formulation can be used for robots in a swarm formation. Nevertheless, intelligent vehicles can also use range-only measurements with some sensors like Ultra Wideband technology [10]. These vehicles are often following each other and do not always see other vehicles on their sides. In this situation, few vehicles are visible and the cooperative localization must be done with less observation than in a swarm formation.

Different formulations exist to manage the fusion of the localization information of the vehicles. A simple Extended Kalman Filter (EKF) can be applied. However, this method can generate data incest issues because of the redundant use of the poses of the vehicles. Other formulations exist to avoid this problem. Each vehicle may communicate the raw measurements from sensors [14] instead of a fused pose. Another solution is to use two EKF [13], [2], the first one without the information from the other using only the sensors

¹The authors are with Sorbonne universités, Université de technologie de Compiègne, CNRS, Heudiasyc UMR 7253, France `firstname.surname@hds.utc.fr`

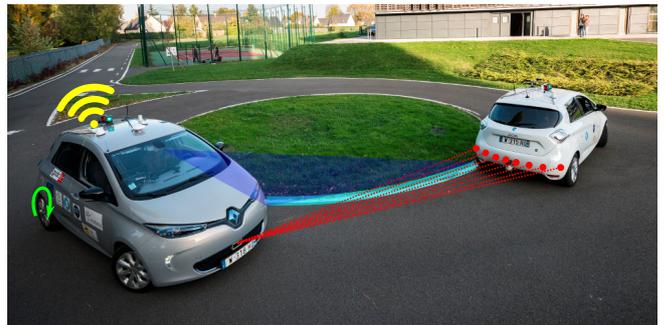


Fig. 1. Platooning of the two Renault Zoés on the test track Seville in the innovation campus of UTC in Compiègne, France. The two vehicles use GNSS localization from a Ublox 8 receiver. The direct Wifi 802.11p communication (in yellow) is simulated in this work, as the carriageway borders detections (in blue). A CAN bus gateway is used to get the dead-reckoning information (in green). Finally, the follower is equipped with an LDMRS LiDAR (in red).

of the vehicle. This first filter estimates the pose of the vehicle for the other, this pose can be communicated. The second filter uses the sensor information of the vehicle and the information of other vehicles to estimate the pose. This pose is not communicated. Other data fusion filters exist to avoid data incest. The covariance intersection filter (CIF) can solve this problem [12]. The covariance matrix of the pose is nevertheless often pessimist. The split covariance intersection filter (SCIF) can give a better estimation of the covariance matrix [11], [6], [19].

In this paper, we combine different sources of information for localization. The most straightforward way is to use a GNSS receiver to compute the vehicle position. However, many low-cost GNSS receivers are inaccurate, that is they suffer from non-centered errors that act as a bias. This bias needs to be compensated in order to have a consistent pose estimate [17]. One possible way to observe this bias is to perceive geo-referenced features in a map such as using a camera for lane markings detection [17]. Another commonly used sensor is the LiDAR. It is often used to detect other vehicles in the surrounding [22]. Cameras can also be used for vehicle detection [18]. This relative measurement between vehicles can also be used to improve the global localization of all the vehicles [16]. Adding also proprioceptive information from the CAN bus and wireless communication (see Fig. 1), we propose a real-time distributed asynchronous cooperative localization with GNSS biases estimation.

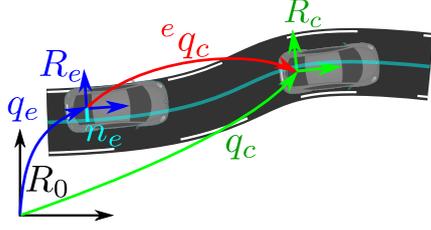


Fig. 2. Pose of the ego vehicle q_e (in blue), of the cooperative vehicle q_c (in green) and relative pose of the cooperative vehicle in the frame of the ego vehicle ${}^e q_c$ (in red). The signed distance to the center of the lane n_e is in light blue.

The main contribution of this work is to present experimental results on how cooperative localization can improve cooperative navigation by estimating accurate and consistent relative localization between vehicles. This work also shows that accurate localization can be propagated to improve inaccurate vehicles pose estimates.

In Sec. II, the distributed asynchronous data fusion is presented and the evolution and the different observations are described in detail. In Sec. III the experimental platform is presented and the results are analyzed.

II. DISTRIBUTED ASYNCHRONOUS DATA FUSION

For navigation purpose, an autonomous vehicle needs to be able to localize itself and the surrounding vehicles as shown in Fig. 2. The knowledge of its own absolute localization is important for autonomous driving, but can be inaccurate. To be able to stay on the road and avoid obstacles, the perception of the carriageway borders and obstacles is needed. The proposed data fusion algorithm uses a Local Dynamical Map (LDM) [5] to represent the other traffic participants in interaction. This map contains its own state and the states of the other agents. In this paper, the agents are only cars. The proposed method uses a low cost GNSS receiver along with lane markings and carriageway borders detection to improve localization. Therefore, the pose (position and heading) of each agent is described in a working frame by a state vector with an associated covariance matrix. This working frame is defined with its origin near the area of driving of the vehicles and two axes oriented toward the east (x axes) and the north (y axes). The relative poses between the agents with their covariance matrices can be computed from the LDM.

The estimated state $\hat{X} = (\hat{X}_1, \dots, \hat{X}_N)^T$ of the LDM contains the states of N agents in cooperation. Each state $\hat{X}_i = (\hat{q}_i, \hat{b}_i, \hat{u}_i)^T$ contains the pose of the agent $\hat{q}_i = (\hat{x}_i, \hat{y}_i, \hat{\theta}_i)^T$, the bias $\hat{b}_i = (\hat{b}_{xi}, \hat{b}_{yi})^T$ of the GNSS receiver of this agent and the kinematic information $\hat{u}_i = (\hat{v}_i, \hat{\omega}_i)^T$. We use the longitudinal speed \hat{v}_i and the yaw rate $\hat{\omega}_i$ to extrapolate the pose at any given time. Therefore, the state \hat{X}_i is a 7 dimensional vector.

Fig. 3 shows the asynchronous data fusion algorithm. At the center, the state of the LDM is represented in green. When an observation (in yellow) is sent by a source of

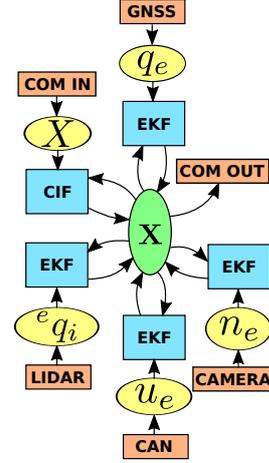


Fig. 3. Asynchronous data fusion. The LDM X is shared between the different processing modules and sent to the other agents in cooperation.

information (in orange) the state is updated by the module in blue. The state is firstly extrapolated at the observation time using the evolution model before using the observation model. The evolution model represents the motion of the vehicles and the observation function links the state to the observation. As each observation update is implemented in a thread, a mutex protects the state while it is processed. This method is used to avoid the loss of an observation if multiple updates are computed in parallel.

This algorithm is fully distributed: each vehicle executes the same algorithm and uses communication to send and receive LDMs. The cooperative localization is improved by this information exchange.

A. Evolution stage

When a new information (sensor measurement or LDM) is received, the state is extrapolated at the time of this new information before doing a Kalman update.

The evolution model uses the kinematic information of each agent i of the LDM :

$$X_i [t|t_{-1}] = X_i [t_{-1}|t_{-1}] + \begin{pmatrix} v_i [t_{-1}|t_{-1}] \Delta t \cos(\theta_i [t_{-1}|t_{-1}] + \omega_i [t_{-1}|t_{-1}] \frac{\Delta t}{2}) \\ v_i [t_{-1}|t_{-1}] \Delta t \sin(\theta_i [t_{-1}|t_{-1}] + \omega_i [t_{-1}|t_{-1}] \frac{\Delta t}{2}) \\ \omega_i [t_{-1}|t_{-1}] \Delta t \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad (1)$$

where $\Delta t = t - t_{-1}$ is the elapsed time between the observation time t and the last estimated time t_{-1} of the LDM.

Let $X [t|t_{-1}] = \text{evo}(X [t_{-1}|t_{-1}], \Delta t)$ the evolution model and Σ_{model} the covariance matrix of the model noise. We suppose that the covariance matrix of the model noise contains the variance of the bias noise and the longitudinal speed and yaw rate noise which are supposed to be constant. The model on the pose is supposed to be very accurate.

The covariance matrix estimate of the state $X[t|t_{-1}]$ is approximated by a first order Taylor expansion:

$$\Sigma_X[t|t_{-1}] = J\Sigma_X[t_{-1}|t_{-1}]J^T + \Sigma_{model}\Delta t^2 \quad (2)$$

where J is the Jacobian matrix of the evolution model

$$J = \frac{\partial \text{evo}}{\partial X[t_{-1}|t_{-1}]} \quad (3)$$

B. LDM update with ego-measurements

Once the state is predicted at exactly the same time as the observation, it can be updated. Two update methods are used depending on whether the information comes from the vehicle's sensors or from other vehicles.

1) *Extended Kalman Filter*: A classical Extended Kalman Filter update is used for internal information. The innovation $Y[t]$ is computed from the observation $Z[t]$, the predicted one using the observation function $\text{obs}()$ and the state $X[t|t_{-1}]$:

$$Y[t] = Z[t] - \text{obs}(X[t|t_{-1}]) \quad (4)$$

The Jacobian matrix of the observation function is then computed as

$$H = \frac{\partial \text{obs}}{\partial X[t|t_{-1}]} \quad (5)$$

The updated state with its covariance matrix is classically computed as follows:

$$\Sigma_Y[t] = H\Sigma_X[t|t_{-1}]H^T + \Sigma_Z[t] \quad (6)$$

$$K = \Sigma_X[t|t_{-1}]H^T\Sigma_Y^{-1}[t] \quad (7)$$

$$X[t|t] = X[t|t_{-1}] + KY[t] \quad (8)$$

$$\Sigma_X[t|t] = (I - KH)\Sigma_X[t|t_{-1}](I - KH)^T + K\Sigma_Z[t]K^T \quad (9)$$

All the observation functions required to implement the update of the Kalman filter are presented in the following.

2) *Dead-reckoning observation function*: The observation model of the dead-reckoning (DR) data is simply the identity for the kinematic part of the state:

$$\text{obs}_{CAN}(X[t|t_{-1}]) = u_e[t|t_{-1}] \quad (10)$$

In practice, the high frequency of the DR sensors (e.g. 50 Hz) guarantees frequent updates of the LDM.

3) *GNSS observation function*: One can observe experimentally the presence of a bias in the GNSS measurements. Fig. 4 pictures the error distribution for two vehicles following each other during a 10-minute test.

The GNSS observation function involves the bias on the position and not on the heading:

$$\text{obs}_{GNSS}(X[t|t_{-1}]) = q_e[t|t_{-1}] + \begin{pmatrix} b_e[t|t_{-1}] \\ 0 \end{pmatrix} \quad (11)$$

This formulation with the bias $b_e[t|t_{-1}]$ adds a degree of freedom to the position. Therefore, this bias is estimated from the difference between the position estimate and the GNSS observation of this position.

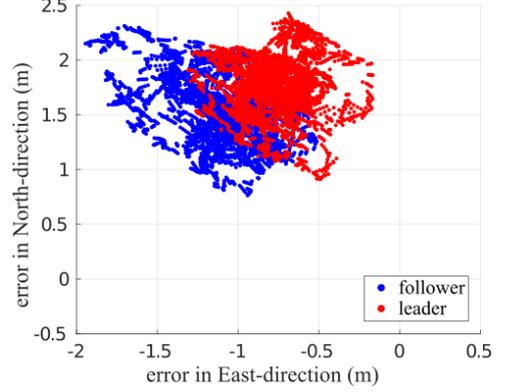


Fig. 4. GNSS errors of two vehicles driving in platoon. Biases are clearly visible.

4) *Camera observation function*: As the GNSS measurements are biased, an observation of the environment which is compared to a high-definition (HD)map is needed to observe the GNSS bias. The problem of changing coordinates between the HD map and the current GNSS system must be handled with great care to avoid introducing errors and maintain centimeter accuracy.

We suppose that each vehicle is equipped with a camera able to detect and localize the carriageway borders. This kind of sensor gives the signed distance directly between the vehicle and the center of the lane.

The observation model of this observation needs to map-match the pose of the ego vehicle $q_e[t|t_{-1}]$ included in the state of the LDM with the known map M :

$$\left[p_{frenet}, \vec{t}_{frenet}, \vec{n}_{frenet} \right] = \text{map_match}(q_e[t|t_{-1}], M) \quad (12)$$

The map matching algorithm gives the closest Frenet frame from the position

$$p_e[t|t_{-1}] = \begin{pmatrix} x_e[t|t_{-1}] \\ y_e[t|t_{-1}] \end{pmatrix}.$$

This frame is composed of its origin p_{frenet} , the unit tangent \vec{t}_{frenet} and the unit normal \vec{n}_{frenet} to the center of the lane.

To compute the signed lateral distance $n_e[t|t_{-1}]$, one can project the position $p_e[t|t_{-1}]$ into the normal \vec{n}_{frenet} :

$$\text{obs}_{CAM}(X[t|t_{-1}]) = (p_e[t|t_{-1}] - p_{frenet})^T \cdot \vec{n}_{frenet} \quad (13)$$

5) *Relative pose observation function*: The relative pose between agent i and the ego vehicle e is computed from their absolute poses q_i and q_e :

$$\text{obs}_{LiDAR}(X[t|t_{-1}]) = \begin{pmatrix} \cos(\theta_e[t|t_{-1}]) & \sin(\theta_e[t|t_{-1}]) & 0 \\ -\sin(\theta_e[t|t_{-1}]) & \cos(\theta_e[t|t_{-1}]) & 0 \\ 0 & 0 & 1 \end{pmatrix} \Delta q \quad (14)$$

where $\Delta q = q_i[t|t_{-1}] - q_e[t|t_{-1}]$. More information is given in [9].

C. LDM update with communicated information

When the LDM of another vehicle is received, the ego-LDM is first extrapolated at the time instant of the received one with the same evolution stage which aligns the LDM timestamps and compensates for the communication latency. Please note that a Kalman update cannot be applied. Indeed, as the agents are exchanging their states, some information cannot be considered as independent since they can be used several times in the data fusion process. A covariance intersection filter is used to avoid data-incest issues.

1) *Covariance Intersection Filter*: The application of this algorithm is very similar to the EKF. The difference is that $\Sigma_X [t|t_{-1}]$ and $\Sigma_Z [t]$ are weighted by $1/\omega$ and $1/(1-\omega)$, respectively. These weights are used to find the new covariance matrix $\Sigma_{X,\omega_{min}} [t|t]$ which gives the smallest uncertainty ellipsoid including the intersection of the uncertainty ellipsoid of the two data to be fused. This means that even if the two uncertainty ellipsoids are correlated, the estimate stays consistent as the intersection is not smaller. For example, if the two uncertainty ellipsoids are the same, the resulting uncertainty ellipsoid remains this ellipsoid and not a smaller one as would give a Kalman filter update.

For a given weight ω , compute (using the Joseph's form to improve the numerical stability):

$$\Sigma_{Y,\omega} [t] = H \left(\frac{1}{\omega} \Sigma_X [t|t_{-1}] \right) H^T + \frac{1}{1-\omega} \Sigma_Z [t] \quad (15)$$

$$K_\omega = \frac{1}{\omega} \Sigma_X [t|t_{-1}] H^T \Sigma_{Y,\omega}^{-1} [t] \quad (16)$$

$$\begin{aligned} \Sigma_{X,\omega} [t|t] &= (I - K_\omega H) \left(\frac{1}{\omega} \Sigma_X [t|t_{-1}] \right) (I - K_\omega H)^T \\ &\quad + K_\omega \frac{1}{1-\omega} \Sigma_Z [t] K_\omega^T \end{aligned} \quad (17)$$

Find ω which minimizes a given norm of the covariance matrix $\Sigma_{X,\omega} [t|t]$, often done using the determinant or the trace:

$$\omega_{min} = \mathbf{arg} \min_{\omega \in [0,1]} (\mathbf{norm} (\Sigma_{X,\omega} [t|t])) \quad (18)$$

The updated state with its covariance matrix is given as follows:

$$\begin{aligned} X [t|t] &= X [t|t_{-1}] + K_{\omega_{min}} Y [t] \\ \Sigma_X [t|t] &= \Sigma_{X,\omega_{min}} [t|t] \end{aligned} \quad (19)$$

2) *Communication observation function*: Let us suppose for simplification that the agents are organized in the same order in the different LDMs. The observation model in this case is simply given by the identity function:

$$\text{obs}_{COM} (X [t|t_{-1}]) = X [t|t_{-1}]. \quad (20)$$

This update with this observation function and the covariance intersection filter is the key stage for cooperative localization. In an experiment in which two vehicles were driving in a platoon, we have observed that the biases of the two vehicles were close to each other but did not result to the same value (see Fig. 4). Estimating the correlation between

the two biases is complex. In [4], the biases are supposed to be the same for all the vehicles. In this work, we choose to model the bias of each agent as an independent bias on its own state and estimate it during the filtering. In practice it is not necessary to include the bias in the state exchange, as we have assumed the biases as uncorrelated. For simplification we exchange the whole state in our experiment.

III. EXPERIMENTAL RESULTS

A. Experimental platform

Two experimental Renault Zoe cars drove in a platoon on a test road composed of two roundabouts linked by a straight lane. The two vehicles run nine laps of this test track and data were recorded. Both vehicles were equipped with a low cost, multi-constellation (GPS, GLONASS) GNSS receiver Ublox 8. The pose of the vehicle was provided by the Ublox receiver using its internal automotive model. Only loose coupling is presented in this work. The ground truth of the poses of both vehicles were obtained by an RTK GNSS receiver / IMU Novatel Span CPT. The dead-reckoning information (longitudinal speed and yaw rate) was retrieved from the CAN bus. These observations were measured by the encoders and the gyrometer of the vehicles. The follower was equipped with a SICK LD-MRS LiDAR. One horizontal layer of this sensor was used to estimate the relative pose of the leader vehicle using the algorithm described in [8], [15]. Finally, the carriageway borders detections were simulated from the observation model applied on the ground truth poses. A Gaussian noise with a standard deviation of 20 cm was added to the signed lateral distance. For map matching, we used the Lanelet projection described in [3], [7].

The data fusion was implemented with the LiDAR perception and the map-matching in python using the ROS (Robot Operating System) framework. These algorithms run with the dataset replay at the original speed. The proposed solution is fast enough for real-time applications on a classical computer.

B. Metrics

In this work, both vehicles have their own LDM in which the leader vehicle pose q_l and the follower vehicle pose q_f are represented. From these two absolute poses, the relative pose of the ego vehicle in the frame of the other can be computed using Eq. 14. In our case, the pose of the leader in the frame of the follower ${}^f q_l$ for the LDM of the follower and the pose of the follower in the frame of the leader ${}^l q_f$ for the LDM of the leader ($q_k = (x_k, y_k, \theta_k)^T$ is the pose of the vehicle at sample k and q_k^* denotes the ground truth).

e_p is the mean of the position error for the K samples:

$$e_p = \frac{1}{K} \sum_{k=1}^K \left\| \begin{pmatrix} x_k^* \\ y_k^* \end{pmatrix} - \begin{pmatrix} x_k \\ y_k \end{pmatrix} \right\| \quad (21)$$

e_θ is the mean of the absolute heading error for the K samples:

$$e_\theta = \frac{1}{K} \sum_{k=1}^K |\theta_k^* - \theta_k| \quad (22)$$

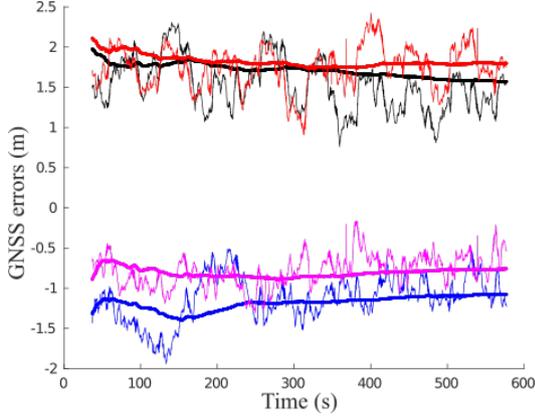


Fig. 5. GNSS errors with bias estimates (in bold) for the follower on x (in blue) and y (in black) and for the leader on x (in magenta) and y (in red).

Finally, the consistency is computed for a probability error of 5% with a χ^2 test. The consistency C corresponds to:

$$C = Pr \left\{ (q_k^* - q_k)^T (\Sigma_{q_k^*} + \Sigma_{q_k})^{-1} (q_k^* - q_k) < \chi_{3;0.05}^2 \right\} \quad (23)$$

where $\chi_{3;0.05}^2 = 7.815$ corresponds to the quantile of a χ^2 distribution with 3 degrees of freedom and a confidence of 95%. $\Sigma_{q_k^*}$ is the covariance matrix of the ground truth of q_k as this ground truth is not certain even with RTK correction. This uncertainty is not negligible when accurate LiDAR measurements are used and must be taken into account. In practice, C is measured as the percentage of samples verifying this test.

C. Biases estimation

When carriageway borders are detected and map-matched with a high-definition map, the GNSS bias can be estimated. Fig. 5 shows the GNSS errors and the biases estimated in the LDM of the follower for the follower for the blue and black curves and in the LDM of the leader for the leader for the magenta and red curves. One can see that the biases are well estimated and stay close to the averages of the errors like low-pass filter. Even if some correlation can be seen between the GNSS errors of the two Ublox receivers, the biases are not the same. These estimated biases are different between the vehicles and change also during the 10 min of the experiment. One can see that the filters are able to estimate quite accurately the biases in real-time.

Using the estimated biases, the GNSS errors can be corrected. Fig. 6 presents these unbiased errors with estimated 95% confidence intervals. The horizontal uncertainty provided by the Ublox 8 is assimilated as the standard deviation which leads to the same confidence intervals on x and y directions. One can see on Fig. 6 that the error bounds are quite pessimistic as the 5864 samples of the dataset are all in the 95% confidence intervals. One can also see in this figure, the periodicity of the confidence intervals. Some trees and buildings are near the test track and therefore the visibility of the satellites is not the same everywhere in

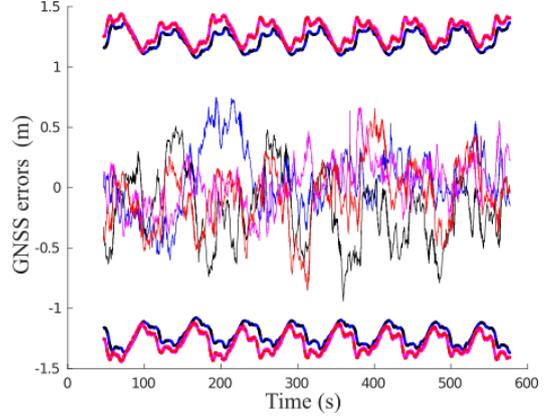


Fig. 6. GNSS errors with biases correction. The 95% confidence intervals have been computed using the uncertainty provided by the Ublox receiver, the uncertainty of the Span CPT and the estimated standard deviation of the biases. x is in blue and y in black for the follower and x is in magenta and y in red for the leader.

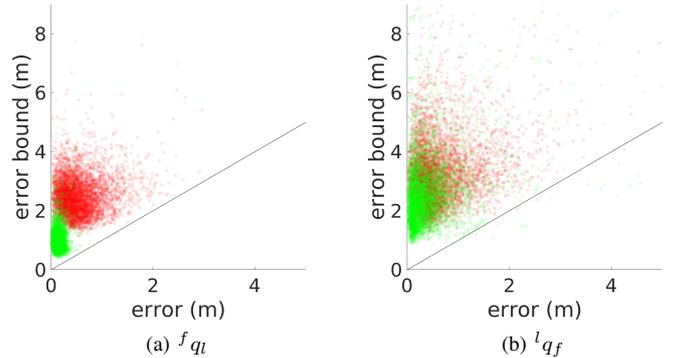


Fig. 7. Stanford diagrams of the relative position errors from the relative poses ${}^f q_l$ and ${}^l q_f$ using a CIF with (in green) and without (in red) LiDAR measurements.

this road. Nevertheless this visibility is similar for the two vehicles but with a small offset as the vehicles are following each other. The nine laps are clearly visible in this pattern.

Tab. I shows the position and heading errors and the consistency when carriageway borders are not available for both vehicles and therefore biases cannot be estimated. The position error drops when the bias can be estimated and the consistency becomes also pessimistic instead of being inconsistent. This consistency is similar as the one of the GNSS error when biases are compensated (see Fig. 6).

D. Cooperative localization accuracy and consistency

Without LiDAR observation both vehicles poses can be considered as uncorrelated. Even if it seems that the state does not improve much when LiDAR measurements are used (see Tab. II), the relative poses computed from this state are much more accurate. The relative position error decreases from 0.54m and 0.66m to 0.15m and 0.43m and the relative heading error from 3.45° and 4.02° to 1.83° and 2.17°. This result is illustrated by (simplified) Stanford diagrams in Fig. 7 where one can see the confidence bound in

TABLE I
POSITION e_p AND HEADING e_θ ERROR AND CONSISTENCY C OF THE COOPERATIVE LOCALIZATION WITH AND WITHOUT BIAS ESTIMATION AND WITHOUT LiDAR MEASUREMENTS.

	Without bias estimation						With bias estimation					
	LDM of f			LDM of l			LDM of f			LDM of l		
	q_f	q_l	Jq_l	q_l	q_f	lq_f	q_f	q_l	Jq_l	q_l	q_f	lq_f
e_p (m)	2.11	1.99	0.63	1.99	2.11	0.59	0.22	0.24	0.55	0.23	0.22	0.73
e_θ ($^\circ$)	1.67	1.36	1.93	1.37	1.67	1.95	2.20	3.37	4.04	3.66	2.10	4.27
C (%)	37.1	77.3	100	75.4	37.8	100	98.4	99.7	100	99.8	98.8	99.6

TABLE II
POSITION e_p AND HEADING e_θ ERROR AND CONSISTENCY C OF THE COOPERATIVE LOCALIZATION WITH A CIF, WITH AND WITHOUT LiDAR MEASUREMENTS AND WHEN THE FIELD OF VIEW OF THE LiDAR IS TIGHT.

	With LiDAR						Without LiDAR						Tight field of view					
	LDM of f			LDM of l			LDM of f			LDM of l			LDM of f			LDM of l		
	q_f	q_l	Jq_l	q_l	q_f	lq_f	q_f	q_l	Jq_l	q_l	q_f	lq_f	q_f	q_l	Jq_l	q_l	q_f	lq_f
e_p (m)	0.20	0.26	0.15	0.25	0.20	0.42	0.21	0.23	0.54	0.23	0.21	0.66	0.21	0.21	0.36	0.21	0.21	0.55
e_θ ($^\circ$)	1.23	2.38	1.83	2.47	1.15	2.17	2.04	2.65	3.45	3.47	1.74	4.02	1.57	2.40	2.85	2.85	1.40	3.21
C (%)	96.8	95.5	99.4	94.9	97.2	98.4	99.4	99.7	100	99.7	99.3	100	99.6	99.7	99.9	99.6	99.7	99.5

function of the relative position error. The confidence bound was computed with a confidence of 95% using a Rayleigh distribution as an approximation of the norm of the error. The use of the LiDAR detection (in green) improves greatly the accuracy while staying consistent as the dots are above the black lines for more than 95% of the samples (the green cloud is much more compact, closer to the vertical axis and the confidence bounds are much smaller). One can see that the relative poses computed from the state of the leader are improved, even if this vehicle is not equipped with a LiDAR and, therefore, not able to localize itself relatively to the other vehicle. The communication improves the states of the LDM of every agent if each agent is linked to others by one relative pose only. Therefore, bidirectional perceptions are not essential in this distributed data fusion scheme even if the vehicle with the perception sensor, the follower in this case, can compute a more accurate relative pose, as illustrated in Fig. 7.

Errors stay consistent in the different scenarios. A good estimate of the uncertainty is very important in this cooperative localization process, not only for reliability but also to obtain accurate estimates of the poses. Indeed, the bias gives a degree of freedom to the filter. If the uncertainty is badly estimated, the bias can have difficulties to properly converge. The communication update has been also tested using EKF instead of CIF. We have noticed that the data incest issues increase the rigidity of the bias which cannot converge properly. The results are in this case inaccurate and inconsistent.

A tight field of view is simulated for the LiDAR. In this scenario, the relative pose is not available in the curved parts of the road. One can see that in this case, when the LiDAR is able to detect other vehicles only on the straight lane, the relative pose is less accurate. Indeed, the perception algorithm gives better results on curved roads when the back and one side of the vehicle are in the field of view of the LiDAR (“L-shape”) instead of only the back (in this case, the heading of the other vehicle is difficult to estimate).

This configuration gives more accurate relative poses. One can also see that the poses are more consistent when the relative pose is not available on the curved roads. Indeed, the lateral part of this pose becomes very accurate and can become sometimes in conflict with the carriageway borders detections. In these two configurations, it can be noticed that the use of the perception presents the advantage to be used with only two vehicles minimum. In many other works using only ranges or bearings [21], [20] more vehicles are needed to be able to build the rigid structure [1] from these relative constraints.

E. Unavailability of carriageway borders

Two other scenarios are presented to test the robustness of this cooperative localization when perception of carriageway borders is not available.

One can see in Tab. III that even if one vehicle cannot detect the carriageway borders (and therefore estimate by itself its own GNSS bias), this bias can be estimated using the pose of the other vehicle and the relative pose between the two vehicles. The consistencies given when the carriageway borders are not available for the follower are less consistent than for the leader. Indeed the transformation from the leader frame to the follower frame presents more non-linearity issues than from the follower frame to the leader frame. Nevertheless, the relative poses stay consistent in both scenarios.

IV. CONCLUSION

This work has presented a cooperative localization architecture by state exchange of traffic participants. It is based on EKF and CIF to manage the correlation between the state of the different vehicles. The state of each vehicle is included in the LDM which is essential for its own navigation tasks. In this process, multiple sources of information are fused: dead-reckoning from the CAN bus, GNSS pose, relative poses between vehicles computed from LiDAR scans, carriageway

TABLE III

POSITION e_p AND HEADING e_θ ERROR AND CONSISTENCY C WHEN CARRIAGEWAY BORDERS ARE NOT AVAILABLE FOR THE FOLLOWER OR FOR THE LEADER.

	Carriageway borders available for both vehicles						Carriageway borders not available for the follower						Carriageway borders not available for the leader					
	LDM of f			LDM of l			LDM of f			LDM of l			LDM of f			LDM of l		
	q_f	q_l	$^f q_l$	q_l	q_f	$^l q_f$	q_f	q_l	$^f q_l$	q_l	q_f	$^l q_f$	q_f	q_l	$^f q_l$	q_l	q_f	$^l q_f$
e_p (m)	0.20	0.26	0.15	0.25	0.20	0.42	0.23	0.28	0.15	0.26	0.24	0.43	0.20	0.39	0.15	0.39	0.21	0.32
e_θ (°)	1.23	2.38	1.83	2.47	1.15	2.17	1.35	2.47	1.82	2.55	1.26	2.54	1.54	2.58	1.79	2.26	1.47	1.66
C (%)	96.8	95.5	99.4	94.9	97.2	98.4	90.4	93.7	99.3	93.2	90.6	97.9	96.4	94.0	99.5	93.5	96.7	98.5

borders measured by a camera and finally the LDMs of other vehicles received by the communication. Experimental results have shown that the use of a LiDAR to link the poses of the different vehicles improves the relative poses computed from the states of the LMDs. Therefore, platoon navigation is more accurate and safer. We have also shown in this work than, even if the carriageway borders are not available for one of the vehicles, the cooperation gives it the capacity to estimate its GNSS bias and improve its own localization.

In future works, this cooperative localization will be tested with more than two vehicles.

ACKNOWLEDGMENT

This work was carried out in the framework of the Equipex ROBOTEX (ANR-10- EQPX-44-01) and Labex MS2T (ANR-11-IDEX-0004-02) projects. It was also carried out within SIVALab, a shared laboratory between Renault and Heudiasyc UMR UTC/CNRS.

REFERENCES

- [1] A. Ahmad, G. D. Tipaldi, P. Lima, and W. Burgard. Cooperative robot localization and target tracking based on least squares minimization. In *IEEE International Conference on Robotics and Automation*, pages 5696–5701, May 2013.
- [2] R. Aufrère, N. Karam, F. Chausse, and R. Chapuis. A state exchange approach in real conditions for multi-robot cooperative localization. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4346–4351, October 2010.
- [3] P. Bender, J. Ziegler, and C. Stiller. Lanelets: Efficient map representation for autonomous driving. In *IEEE Intelligent Vehicles Symposium Proceedings*, pages 420–425, June 2014.
- [4] Luís Conde Bento, Philippe Bonnifait, and Urbano J. Nunes. Cooperative GNSS positioning aided by road-features measurements. *Transportation Research Part C: Emerging Technologies*, 79:42–57, June 2017.
- [5] T. Eiter, H. Füreder, F. Kasslatter, J. X. Parreira, and P. Schneider. Towards a Semantically Enriched Local Dynamic Map. *International Journal of Intelligent Transportation Systems Research*, 17(1):32–48, January 2019.
- [6] H. Li and F. Nashashibi. Cooperative Multi-Vehicle Localization Using Split Covariance Intersection Filter. *IEEE Intelligent Transportation Systems Magazine*, 5(2):33–44, 2013.
- [7] E. Héry, S. Masi, Ph. Xu, and Ph. Bonnifait. Map-based curvilinear coordinates for autonomous vehicles. In *IEEE Intelligent Transportation Systems Conference*, pages 16–19, Yokohama, Japan, 2017.
- [8] E. Héry, Ph. Xu, and Ph. Bonnifait. LiDAR based relative pose and covariance estimation for communicating vehicles exchanging a polygonal model of their shape. In *10th Workshop on Planning, Perception and Navigation for Intelligent Vehicles*, Madrid, Spain, 2018.
- [9] E. Héry, Ph. Xu, and Ph. Bonnifait. Pose and covariance matrix propagation issues in cooperative localization with LiDAR perception. In *IEEE Intelligent Vehicles Symposium*, pages 1091–1096, Paris, France, June 2019.
- [10] G. M. Hoangt, B. Denis, J. H  irri, and D. Slock. Cooperative Localization in VANETS: An Experimental Proof-of-Concept Combining GPS, IR-UWB Ranging and V2v Communications. In *Workshop on Positioning, Navigation and Communications (WPNC)*, pages 1–6, October 2018.
- [11] S. Julier and J. Uhlmann. General Decentralized Data Fusion with Covariance Intersection (CI). In David Hall and James Llinas, editors, *Multisensor Data Fusion, 2 Volume Set*, volume 3. CRC Press, June 2001.
- [12] S. J. Julier and J. K. Uhlmann. A non-divergent estimation algorithm in the presence of unknown correlations. In *Proceedings of the American Control Conference (Cat. No.97CH36041)*, volume 4, pages 2369–2373 vol.4, June 1997.
- [13] N. Karam, F. Chausse, R. Aufrère, and R. Chapuis. Localization of a Group of Communicating Vehicles by State Exchange. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 519–524, October 2006.
- [14] K. Lassoued, I. Fantoni, and P. Bonnifait. Mutual Localization and Positioning of Vehicles Sharing GNSS Pseudoranges: Sequential Bayesian Approach and Experiments. In *IEEE International Conference on Intelligent Transportation Systems*, pages 1896–1901, September 2015.
- [15] Kok-Lim Low. Linear Least-Squares Optimization for Point-to-Plane ICP Surface Registration. Technical Report TR04-004, Department of Computer Science University of North Carolina at Chapel Hill, February 2004.
- [16] A. U. Peker, T. Acarman,   Yaman, and E. Y ksel. Vehicle localization enhancement with VANETS. In *IEEE Intelligent Vehicles Symposium Proceedings*, pages 661–666, June 2014.
- [17] Z. Tao, Ph Bonnifait, V. Fremont, and J. Ibanez-Guzman. Mapping and localization using GPS, lane markings and proprioceptive sensors. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 406–412, Tokyo, November 2013. IEEE.
- [18] M. Tekeli,   Yaman, T. Acarman, and M. Akin. A Fusion of a Monocular Camera and Vehicle-to-Vehicle Communication for Vehicle Tracking: An Experimental Study. In *IEEE Intelligent Vehicles Symposium (IV)*, pages 854–859, June 2018.
- [19] T. R. Wanasinghe, G. K. I. Mann, and R. G. Gosine. Decentralized Cooperative Localization for Heterogeneous Multi-robot System Using Split Covariance Intersection Filter. In *Canadian Conference on Computer and Robot Vision*, pages 167–174, May 2014.
- [20] D. Zelazo, A. Franchi, H. H. B lthoff, and P. R. Giordano. Decentralized Rigidity Maintenance Control with Range Measurements for Multi-Robot Systems. *The International Journal of Robotics Research*, 34(1):105–128, January 2015.
- [21] D. Zelazo, A. Franchi, and P. R. Giordano. Rigidity Theory in SE(2) for Unscaled Relative Position Estimation using only Bearing Measurements. *European Control Conference (ECC)*, pages 2703–2708, June 2014.
- [22] Xiao Zhang, Wenda Xu, Chiyu Dong, and John M. Dolan. Efficient L-shape fitting for vehicle detection using laser scanners. In *IEEE Intelligent Vehicles Symposium (IV)*, pages 54–59, Los Angeles, CA, USA, June 2017. IEEE.