

Graph based Cooperative Localization for Connected and Semi-Autonomous Vehicles

1st Nikos Piperigkos
Computer Engineering
& Informatics Department
University of Patras, Greece
piperigkos@ceid.upatras.gr

2nd Aris S. Lalos
Industrial Systems Institute
Athena Research Center
Patras, Greece
lalos@isi.gr

3rd Kostas Berberidis
Computer Engineering
& Informatics Department
University of Patras, Greece
berberid@ceid.upatras.gr

Abstract—Cooperative Real-time Localization is expected to play a crucial role in various applications in the field of Connected and Semi-Autonomous vehicles (CAVs), such as collision avoidance/warning, cooperative adaptive cruise control, etc. Future 5G wireless systems are expected to enable cost-effective Vehicle-to-Everything (V2X) systems, allowing CAVs to share the measured data with other entities of the network. Typical measurement models usually deployed for this problem, are absolute position from Global Positioning System (GPS), relative distance to neighboring vehicles and relative angle or azimuth angle, extracted from Light Detection and Ranging (LIDAR) or Radio Detection and Ranging (RADAR) sensors. In this paper, we provide a cooperative localization approach that performs multi modal-fusion between the interconnected vehicles, by representing a fleet of connected cars as an undirected graph, encoding each vehicle position relative to its neighboring vehicles. This method is based on the so called Laplacian Processing, a Graph Signal Processing tool, that allows to capture intrinsic geometry of the undirected graph of vehicles rather than their absolute position on global coordinate system, significantly outperforming current state of the art approaches, in terms of localization mean square and maximum absolute error and computational complexity.

Index Terms—Cooperative Localization, 5G, CAVs, Multi-modal fusion

I. INTRODUCTION

Real-time Localization is one of the main pillars of Intelligent Transportation Systems (ITS). Although, Global Navigation Satellite Systems (GNSSs), e.g GPS, provide absolute position information, their accuracy is limited to 10 m or higher [8], [10], especially in harsh environments such as urban canyons and tunnels. Since the desired localization error in autonomous driving applications should be no greater than 5 m in the worst case [7], efficient methods for vehicle localization should be developed. In recent years, there is a growing interest in Cooperative Localization (CL) as a means to improve GPS accuracy. CL is based on the 5G communication technology V2X, allowing the vehicles of a Vehicular-Ad-hoc-NETwork (VANET) to share information among them. Useful information could be absolute position from GPS, relative distance or angle to neighboring vehicles from LIDAR/RADAR. Thus, CL consists in fusing different measurement modalities used in VANET, in order to improve the position accuracy.

Recent approaches of CL on VANETs, include [2] where an objective function, based on absolute positions, relative distances and relative angles measurements of the vehicles, is minimized by employing Alternating Direction Method of Multipliers (ADMM). In [3], vehicles share absolute position, relative position and motion state measurements and CL is performed by a covariance intersection filter (CIF), integrating those informations. The VANET of [4], fuses absolute position and range measurements, using Extended Kalman Filter (EKF) and CIF. In [5], a CL method in tunnels, that fuses V2X measurements using particle filtering, is presented. In [6], a CL method in urban canyons, that fuses absolute position and range measurements using EKF is proposed. In [7], CL is achieved by employing V2X measurements between vehicles and infrastructure. In [8], GPS and vehicles dynamics measurements are fed into a Kalman Filter (KF) and combined with any information available about surrounding features such as relative positions of people, traffic lights, etc. In [9], a Bayesian approach that fuses GPS and inter-vehicle distance measurements, is employed in order to perform CL. In [1], CL approaches in Wireless Sensor Networks are distinguished to Bayesian and Non-Bayesian, that rely on Maximum A Posteriori (MAP) estimation and Maximum Likelihood Estimation (MLE), respectively. In [12], distributed solutions that fuse received signal strength and angle of arrival measurements for Internet-of-Things nodes localization are presented. Finally, [10] and [11] provide an overview for cooperative and non-cooperative localization techniques in VANETs.

The previously discussed methods, focus only on the pairwise measurements of the interconnected vehicles. To the best of our knowledge, the method proposed here is the first one which, apart from the measurement models, it also takes into account the connectivity properties of the underlying graph formed by the involved vehicles. Moreover, we integrate those properties in our location estimation approach.

Our main contributions can be summarized as follows:

- A novel method for efficient CL in VANETs is proposed. The method performs cooperative multi-modal fusion exploiting the intrinsic geometric properties. This is achieved by encoding each vehicle's position, relative to its neighboring vehicles.
- The new method can be implemented both in a central-

ized (assuming the existence of a fusion center) and local ('distributed') fashion. In the latter case, each vehicle interacts directly only with its immediate neighborhood.

- As shown via extensive experiments, the new method outperforms existing state-of-the-art method in terms of both accuracy and execution time. The gain in execution time is as high as 2 or 3 orders of magnitude. Note that the method chosen for comparison is the one in [2] because it exploits exactly the same types of measurements.

The rest of the paper is organized as follows: Section II provides the background of traditional CL in VANETs; Section III presents the proposed method while Section IV is dedicated to the experimental setup and simulation results and Section V concludes our work.

II. COOPERATIVE NETWORK LOCALIZATION

Consider a 2-D region where N connected vehicles collect measurements while moving. An example of such a VANET, is shown in Fig. 1. The location of the i -th vehicle at k -th

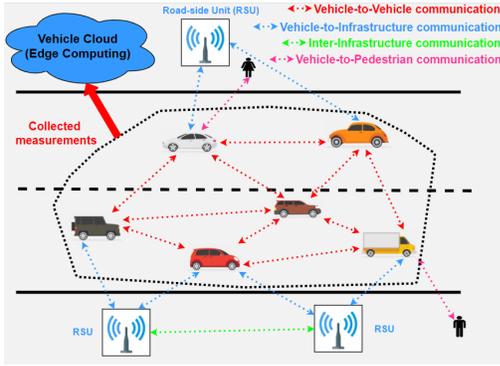


Fig. 1. Example of VANET

time instant is given by $\mathbf{x}_i^{(k)} = [x_i^{(k)} \ y_i^{(k)}]^T$. Each vehicle knows its absolute position from GPS and measures its relative distances and angles with respect to neighboring vehicles using LIDAR or RADAR. The true relative distance $z_{d,ij}^{(k)}$ between connected vehicles i and j is given by:

$$z_{d,ij}^{(k)} = \left\| \mathbf{x}_j^{(k)} - \mathbf{x}_i^{(k)} \right\| \quad (1)$$

, where $\|\cdot\|$ is the l_2 norm. The true relative angle (shown in Fig. 2) $z_{a,ij}^{(k)}$ between neighboring and connected vehicles i and j is given by:

$$z_{a,ij}^{(k)} = \arctan \frac{y_j^{(k)} - y_i^{(k)}}{x_j^{(k)} - x_i^{(k)}} \quad (2)$$

The acquired measurements are assumed to be described by the following models:

- Relative Distance measurement:

$$\tilde{z}_{d,ij}^{(k)} = z_{d,ij}^{(k)} + w_d^{(k)}, \quad w_d^{(k)} \sim \mathcal{N}(0, \sigma_d^2) \quad (3)$$

- Relative Angle measurement:

$$\tilde{z}_{a,ij}^{(k)} = z_{a,ij}^{(k)} + w_a^{(k)}, \quad w_a^{(k)} \sim \mathcal{N}(0, \sigma_a^2) \quad (4)$$

- Absolute position measurement:

$$\tilde{z}_{p,i}^{(k)} = \mathbf{x}_i^{(k)} + \mathbf{w}_p^{(k)}, \quad \mathbf{w}_p^{(k)} \sim \mathcal{N}(0, \Sigma_p) \quad (5)$$

Covariance matrix Σ_p is a diagonal matrix equal to $\text{diag}(\sigma_x^2, \sigma_y^2)$.

Note that the noise introduced in the measurements is assumed to be Gaussian, as commonly done in relevant literature [2], [3], [4], [5].

A typical approach in CL is to formulate an objective cost function $\mathcal{C}(\mathbf{x}^{(k)})$ according to MLE [1], [2] and to minimize it with respect to locations $\mathbf{x}_i^{(k)}$, in order to reduce the error of absolute position measurement. The probability density functions of the respective measurement models are defined below:

$$\mathcal{P}_1 = \mathcal{P}(\tilde{z}_{d,ij}^{(k)} | \mathbf{x}_i^{(k)}, \mathbf{x}_j^{(k)}) = \mathcal{N}(z_{d,ij}^{(k)}, \sigma_d^2) \quad (6)$$

$$\mathcal{P}_2 = \mathcal{P}(\tilde{z}_{a,ij}^{(k)} | \mathbf{x}_i^{(k)}, \mathbf{x}_j^{(k)}) = \mathcal{N}(z_{a,ij}^{(k)}, \sigma_a^2) \quad (7)$$

$$\mathcal{P}_3 = \mathcal{P}(\tilde{z}_{p,i}^{(k)} | \mathbf{x}_i^{(k)}) = \frac{1}{2\pi\sigma_x\sigma_y} \exp \left[-\frac{1}{2} \left(\frac{(\tilde{z}_{p,i}^{x,(k)} - x_i^{(k)})^2}{\sigma_x^2} + \frac{(\tilde{z}_{p,i}^{y,(k)} - y_i^{(k)})^2}{\sigma_y^2} \right) \right] \quad (8)$$

The likelihood function of the measurement models can be written as:

$$\mathcal{L}(\mathbf{x}^{(k)}) = \prod_{i \in N, j \in N(i)} \mathcal{P}_1 \prod_{i \in N, j \in N(i)} \mathcal{P}_2 \prod_{i \in N} \mathcal{P}_3 \quad (9)$$

, where $N(i)$ denotes the set of neighbors of the i -th vehicle. If we take the logarithm of (9), then the objective cost function (same as in [2] and similar to that of [1]) is given by:

$$\begin{aligned} \mathcal{C}(\mathbf{x}^{(k)}) = & \sum_{i \in N, j \in N(i)} \frac{(\tilde{z}_{d,ij}^{(k)} - z_{d,ij}^{(k)})^2}{2\sigma_d^2} + \sum_{i \in N, j \in N(i)} \frac{(\tilde{z}_{a,ij}^{(k)} - z_{a,ij}^{(k)})^2}{2\sigma_a^2} \\ & + \sum_{i \in N} \frac{1}{2} \left[\frac{(\tilde{z}_{p,i}^{x,(k)} - x_i^{(k)})^2}{\sigma_x^2} + \frac{(\tilde{z}_{p,i}^{y,(k)} - y_i^{(k)})^2}{\sigma_y^2} \right] \quad (10) \end{aligned}$$

For the minimization of (10), distributed ADMM [2], cooperative gradient descent [1] or interior point methods (provided by CVX software) can be applied.

III. SPARSE LAPLACIAN BASED LOCALIZATION

The method that will be derived in this Section is based on a proper extension of the Laplacian Processing technique [13] that was originally proposed for surface representation. The motivation was that by modelling a VANET as an undirected graph, one can exploit not only the different measurement modalities between connected vehicles, but also the connectivity properties of vehicles.

A. Centralized Scheme

Let $G^{(k)} = (\mathcal{V}^{(k)}, \mathcal{E}^{(k)})$ be an undirected graph, which changes over time, where $\mathcal{V}^{(k)}$ is the set of vertices and $\mathcal{E}^{(k)}$ the set of edges. Each vertex $v_i^{(k)}$ is represented by absolute cartesian coordinates as $v_i^{(k)} = [x_i^{(k)} \ y_i^{(k)}]$. The differential coordinates $\delta_i^{(k)}$ for each vertex $v_i^{(k)}$ are defined as:

$$\delta_i^{(k)} = [\delta_i^{(k),(x)} \ \delta_i^{(k),(y)}] = v_i^{(k)} - \frac{1}{d_i^{(k)}} \sum_{j \in N_{(i)}} v_j^{(k)}$$

, where $d_i^{(k)}$ is the number of neighbors of vertex $v_i^{(k)}$ and $N_{(i)}$ denotes the set of neighbors of $v_i^{(k)}$. Obviously, $\delta^{(k),(x)}, \delta^{(k),(y)} \in \mathbb{R}^{N \times 1}$. We also define the diagonal degree matrix $D^{(k)} \in \mathbb{R}^{N \times N}$ (with $D^{(k)}[i, i] = d_i^{(k)}$ and $D^{(k)}[i, j] = 0$, if $i \neq j$) and adjacency matrix $A^{(k)} \in \mathbb{R}^{N \times N}$ (with $A^{(k)}[i, j] = 1$, if $(i, j) \in \mathcal{E}^{(k)}$ and $A^{(k)}[i, j] = 0$, otherwise). Finally, the Laplacian matrix $L^{(k)} \in \mathbb{R}^{N \times N}$ of graph is equal to $L^{(k)} = D^{(k)} - A^{(k)}$. Relying on the previously defined $\delta^{(k)}$ coordinates, one can recover the true absolute coordinates of the vertices, represented by the vectors $x^{(k)} \in \mathbb{R}^{N \times 1}$ and $y^{(k)} \in \mathbb{R}^{N \times 1}$, by solving the systems:

$$L^{(k)} x^{(k)} = D^{(k)} \delta^{(k),(x)}, \quad L^{(k)} y^{(k)} = D^{(k)} \delta^{(k),(y)} \quad (11)$$

Each vehicle, using LIDAR or RADAR, can also exploit, the Azimuth Angle (shown in Fig. 2) measurement between vehicle observer i and vehicle target j :

$$\tilde{z}_{az,ij}^{(k)} = \begin{cases} \lambda\pi + \arctan \frac{|x_j^{(k)} - x_i^{(k)}|}{|y_j^{(k)} - y_i^{(k)}|} + w_{az}^{(k)}, & \lambda = 0, 1 \\ \lambda\pi + \arctan \frac{|y_j^{(k)} - y_i^{(k)}|}{|x_j^{(k)} - x_i^{(k)}|} + w_{az}^{(k)}, & \lambda = \frac{1}{2}, \frac{3}{2} \\ w_{az}^{(k)} \sim \mathcal{N}(0, \sigma_{az}^2) \end{cases} \quad (12)$$

For example, in Fig. 2, $\lambda = \frac{1}{2}$. We can also notice from Fig. 2

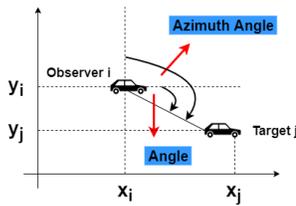


Fig. 2. Angle measurements

that:

$$\delta_i^{(k),(x)} = \frac{1}{d_i^{(k)}} \sum_{j \in N_{(i)}} -\tilde{z}_{d,ij}^{(k)} \sin \tilde{z}_{az,ij}^{(k)} \quad (13)$$

$$\delta_i^{(k),(y)} = \frac{1}{d_i^{(k)}} \sum_{j \in N_{(i)}} -\tilde{z}_{d,ij}^{(k)} \cos \tilde{z}_{az,ij}^{(k)} \quad (14)$$

By considering the vehicles of the network as vertices of a graph (example in Fig. 3) and the communication link between the neighbors as its edges, we can create matrices

$D^{(k)}, A^{(k)}, L^{(k)}$. Moreover, each vehicle utilizing measurement models (3) and (12), can define the $\delta_i^{(k)}$ coordinates and sent them to a fusion center. The latter will try to solve the systems of (11) and estimate the true locations of vehicles. However, $L^{(k)}$ is a singular matrix, which implies that systems in (11) are not solvable. Thus, we need to add into the systems some anchor points $c_i^{(k)} = [c_i^{(k),(x)} \ c_i^{(k),(y)}]$ with known absolute coordinates. Furthermore, if we assume as anchors

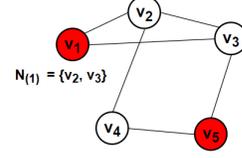


Fig. 3. Example of VANET graph

the vertices $v_1^{(k)}$ and $v_5^{(k)}$, then we define the extended $\tilde{L}^{(k)}$ which is equal to:

$$\tilde{L}^{(k)} = \begin{pmatrix} L^{(k)} \\ e_1 \\ e_5 \end{pmatrix}$$

, where $e_i \in \mathbb{R}^{1 \times N}$ is a vector with zeros and $e_i[i] = 1$. If the number of available anchors is α , then $\tilde{L}^{(k)} \in \mathbb{R}^{(N+\alpha) \times N}$. After adding the anchor points, the systems in (11) can be re-written as:

$$\tilde{L}^{(k)} x^{(k)} = b^{(k),(x)}, \quad b^{(k),(x)} = [d_1^{(k)} \delta_1^{(k),(x)} \dots c_\alpha^{(k),(x)}]^T \quad (15)$$

$$\tilde{L}^{(k)} y^{(k)} = b^{(k),(y)}, \quad b^{(k),(y)} = [d_1^{(k)} \delta_1^{(k),(y)} \dots c_\alpha^{(k),(y)}]^T \quad (16)$$

The solution of (15) and (16) corresponds to the minimizing of:

$$\operatorname{argmin}_{x^{(k)}} \left\| \tilde{L}^{(k)} x^{(k)} - b^{(k),(x)} \right\|^2 \quad (17)$$

$$\operatorname{argmin}_{y^{(k)}} \left\| \tilde{L}^{(k)} y^{(k)} - b^{(k),(y)} \right\|^2 \quad (18)$$

The required location vectors $x^{(k)}$ and $y^{(k)}$ can be computed as follows:

$$x^{(k)} = \left(\tilde{L}^{(k)T} \tilde{L}^{(k)} \right)^{-1} \tilde{L}^{(k)T} b^{(k),(x)} \quad (19)$$

$$y^{(k)} = \left(\tilde{L}^{(k)T} \tilde{L}^{(k)} \right)^{-1} \tilde{L}^{(k)T} b^{(k),(y)} \quad (20)$$

In practice, as anchor points the noisy GPS positions (measurement model (5)) of the vehicles of the network can be used. This differentiate us with the majority of CL methods, where the location of anchors is considered to be the true and without any ambiguity. Furthermore, each $\delta_i^{(k)}$ must be multiplied with $d_i^{(k)}$, so as to remain in accordance with (11).

Moreover, due to the limited communication range of vehicles' transceivers, matrix $\tilde{\mathbf{L}}^{(k)}$ is actually sparse. Thus, one may use a sparse least-squares solver [14], in order to solve (17) and (18). We named our centralized CL approach **Graph based CL-Centralized Laplacian** or **GCL-CLapl**. The main steps of **GCL-CLapl** are summarized on **Algorithm 1**.

Algorithm 1: Graph based CL-Centralized Laplacian or GCL-CLapl

Input: N, T
Output: $\mathbf{x}^{(k)}, \mathbf{y}^{(k)}$

```

1 for  $k = 1, 2, \dots, T$  do
2    $\delta^{(k), (x)} = \delta^{(k), (y)} = \mathbf{b}^{(k), (x)} = \mathbf{b}^{(k), (y)} = \mathbf{0}$ ;
3   Movement of vehicles;
4    $\mathbf{L}^{(k)} = \mathbf{D}^{(k)} - \mathbf{A}^{(k)}$ ;
5   Create  $\tilde{\mathbf{L}}^{(k)}$  from  $\mathbf{L}^{(k)}, \mathbf{D}^{(k)}, \mathbf{A}^{(k)}$ ;
6   for  $i = 1, 2, \dots, N$  do
7      $\delta_i^{(k), (x)} = \frac{1}{d_i^{(k)}} \sum_{j \in N(i)} -\tilde{z}_{d,ij}^{(k)} \sin \tilde{z}_{az,ij}^{(k)}$ ;
8      $\delta_i^{(k), (y)} = \frac{1}{d_i^{(k)}} \sum_{j \in N(i)} -\tilde{z}_{d,ij}^{(k)} \cos \tilde{z}_{az,ij}^{(k)}$ ;
9   end
10   $\mathbf{b}^{(k), (x)} = \left[ d_1^{(k)} \delta_1^{(k), (x)} \dots c_\alpha^{(k), (x)} \right]^T$ ;
11   $\mathbf{b}^{(k), (y)} = \left[ d_1^{(k)} \delta_1^{(k), (y)} \dots c_\alpha^{(k), (y)} \right]^T$ ;
12   $\mathbf{x}^{(k)} = \left( \tilde{\mathbf{L}}^{(k)T} \tilde{\mathbf{L}}^{(k)} \right)^{-1} \tilde{\mathbf{L}}^{(k)T} \mathbf{b}^{(k), (x)}$ ;
13   $\mathbf{y}^{(k)} = \left( \tilde{\mathbf{L}}^{(k)T} \tilde{\mathbf{L}}^{(k)} \right)^{-1} \tilde{\mathbf{L}}^{(k)T} \mathbf{b}^{(k), (y)}$ ;
14 end
```

B. Local Laplacian Implementation

In a local Laplacian processing approach, each vehicle receives GPS measurements from neighboring vehicles and solves a local Laplacian problem. This may offer some advantages as compared to the centralized technique, such as: a) no need to define a central node, which is difficult in such a varying topology of VANETs and b) robustness against failure of the central node. In the local Laplacian implementation presented here, instead of using the overall $\tilde{\mathbf{L}}^{(k)}$, we use local $\tilde{\mathbf{L}}_i^{(k)} \in \mathbb{R}^{(N(i)+2) \times (N(i)+1)}$. For example, local $\tilde{\mathbf{L}}_1^{(k)}$ for the vertex $\mathbf{v}_1^{(k)}$ of Fig. 3 is equal to:

$$\tilde{\mathbf{L}}_1^{(k)} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \\ & & \mathbf{e}_3 \end{pmatrix} \quad (21)$$

While, local vectors $\mathbf{b}_1^{(k), (x)} \in \mathbb{R}^{(N(1)+2) \times 1}$ and $\mathbf{b}_1^{(k), (y)} \in \mathbb{R}^{(N(1)+2) \times 1}$ are equal to:

$$\mathbf{b}_1^{(k), (x)} = \begin{bmatrix} \tilde{z}_{p,2}^{(k), (x)} & \tilde{z}_{p,3}^{(k), (x)} & d_1^{(k)} \delta_1^{(k), (x)} & \tilde{z}_{p,1}^{(k), (x)} \end{bmatrix}^T \quad (22)$$

$$\mathbf{b}_1^{(k), (y)} = \begin{bmatrix} \tilde{z}_{p,2}^{(k), (y)} & \tilde{z}_{p,3}^{(k), (y)} & d_1^{(k)} \delta_1^{(k), (y)} & \tilde{z}_{p,1}^{(k), (y)} \end{bmatrix}^T \quad (23)$$

In order to determine the location of $\mathbf{v}_1^{(k)}$, we must solve the two systems:

$$\tilde{\mathbf{L}}_1^{(k)} \mathbf{x}_{N(1)}^{(k)} = \mathbf{b}_1^{(k), (x)}, \quad \tilde{\mathbf{L}}_1^{(k)} \mathbf{y}_{N(1)}^{(k)} = \mathbf{b}_1^{(k), (y)} \quad (24)$$

Vectors $\mathbf{x}_{N(1)}^{(k)} \in \mathbb{R}^{(N(1)+1) \times 1}$ and $\mathbf{y}_{N(1)}^{(k)} \in \mathbb{R}^{(N(1)+1) \times 1}$ can be computed (based on (19) and (20)) as follows:

$$\mathbf{x}_{N(1)}^{(k)} = \left(\tilde{\mathbf{L}}_1^{(k)T} \tilde{\mathbf{L}}_1^{(k)} \right)^{-1} \tilde{\mathbf{L}}_1^{(k)T} \mathbf{b}_1^{(k), (x)} \quad (25)$$

$$\mathbf{y}_{N(1)}^{(k)} = \left(\tilde{\mathbf{L}}_1^{(k)T} \tilde{\mathbf{L}}_1^{(k)} \right)^{-1} \tilde{\mathbf{L}}_1^{(k)T} \mathbf{b}_1^{(k), (y)} \quad (26)$$

The estimated location of $\mathbf{v}_1^{(k)}$ is the last element of vectors $\mathbf{x}_{N(1)}^{(k)}$ and $\mathbf{y}_{N(1)}^{(k)}$ (according to the definition of $\mathbf{b}_1^{(k), (x)}$, $\mathbf{b}_1^{(k), (y)}$ and $\tilde{\mathbf{L}}_1^{(k)}$). Thus, in **Step 1** each vehicle exchanges GPS positions with its neighbors, in **Step 2** it computes its own $\delta_i^{(k)}$ coordinate and in **Step 3** estimates its location using (25) and (26). The presented approach, offers resilience against potential cyber-failures, while as it will be also shown in the following section its accuracy is close enough to that of the global centralized method. As a future step we plan to extend this method to a fully distributed scheme, though the gain margins seems to be limited, compared to the communication and processing overheads that will be potentially required. We named the local (or 'distributed') approach of this Section **Graph based CL-Distributed Laplacian** or **GCL-DLapl**. **GCL-DLapl** is summarized on **Algorithm 2**. Note that the

Algorithm 2: Graph based CL-Distributed Laplacian or GCL-DLapl

Input: N, T
Output: $\mathbf{x}_{N(i)}^{(k)}, \mathbf{y}_{N(i)}^{(k)}$

```

1 for  $k = 1, 2, \dots, T$  do
2   Movement of vehicles;
3   for  $i = 1, 2, \dots, N$  do
4     Create  $\tilde{\mathbf{L}}_i^{(k)}$  using (21);
5      $\delta_i^{(k), (x)} = \frac{1}{d_i^{(k)}} \sum_{j \in N(i)} -\tilde{z}_{d,ij}^{(k)} \sin \tilde{z}_{az,ij}^{(k)}$ ;
6      $\delta_i^{(k), (y)} = \frac{1}{d_i^{(k)}} \sum_{j \in N(i)} -\tilde{z}_{d,ij}^{(k)} \cos \tilde{z}_{az,ij}^{(k)}$ ;
7     Define  $\mathbf{b}_i^{(k), (x)}$  and  $\mathbf{b}_i^{(k), (y)}$  from (22) and (23);
8      $\mathbf{x}_{N(i)}^{(k)} = \left( \tilde{\mathbf{L}}_i^{(k)T} \tilde{\mathbf{L}}_i^{(k)} \right)^{-1} \tilde{\mathbf{L}}_i^{(k)T} \mathbf{b}_i^{(k), (x)}$ ;
9      $\mathbf{y}_{N(i)}^{(k)} = \left( \tilde{\mathbf{L}}_i^{(k)T} \tilde{\mathbf{L}}_i^{(k)} \right)^{-1} \tilde{\mathbf{L}}_i^{(k)T} \mathbf{b}_i^{(k), (y)}$ ;
10    Last element of  $\mathbf{x}_{N(i)}^{(k)}$  and  $\mathbf{y}_{N(i)}^{(k)}$  corresponds to the estimated true location of  $i$ -th vehicle;
11  end
12 end
```

local scheme, could also be useful for the self-localization task, where instead of cooperating vehicles, visually detected road landmarks (e.g. poles, facades) [15] can be used. Distances and angles between the landmark and the vehicle or the relative

position of landmark (determined by LIDAR or Camera), could be exploited.

IV. SIMULATIONS

A. Experimental Setup

The conducted experiments were based on different levels of noise of measurement models and different number of vehicles. The initial locations of vehicles were determined assuming uniform distribution for x and y coordinates. We generate the trajectories of vehicles for 500 time instances according to the vehicle motion model of [5]. For a reduced computational load, active communication links are assumed to exist between the vehicles, only if their distance is lower than 20 m, and the number of connected neighbors is at most 6. Moreover, each vehicle always has a connected neighbor. Based on the aforementioned assumptions we consider three different approaches: i) the Traditional CL of (10) based on MLE or TCL-MLE, ii) the proposed centralized scheme **GCL-CLapl**, and iii) the proposed local ('distributed') scheme **GCL-DLapl**. We apply the three approaches each time instant. We minimize (10) using a cooperative gradient descent algorithm. Each time instant, we define $\sigma_x = 3$ m, $\sigma_y = 2.5$ m in order to have an average GPS error equal to 3.4 m and we calculate the Localization Mean Square and the Maximum Absolute Error of GPS and the 3 aforementioned methods. In Fig. 4, we present the Cumulative Distribution Functions (CDFs) of Localization Maximum Absolute Errors for 10 vehicles with (a) $\sigma_d = 1$ m, $\sigma_{a,az} = 4^\circ$ and (b) $\sigma_d = 3$ m, $\sigma_{a,az} = 5^\circ$. In Fig. 5, we present the CDFs of Localization Mean Square Errors for (a) 5 and (b) 20 vehicles with $\sigma_d = 1$ m, $\sigma_{a,az} = 4^\circ$. The variance of noise of distances and angles is based on [2].

B. Evaluation Study

The performances of the proposed method, as well as that of the method in [2], depend on the error introduced in range measurements. The variance of the range measurements error is usually much smaller than the variance of the GPS error, allowing an accurate estimation of the differential coordinates, which seems to be a requirement for estimating accurately the actual vehicles position. In Fig. 4, it is evident that there is a significant reduction of GPS error. In Fig. 4-(a) the reduction of error is 69% for **GCL-CLapl**, 60% for **GCL-DLapl** and 60% for TCL-MLE. In Fig. 4-(b) the reduction of error is 56% for **GCL-CLapl**, 53% for **GCL-DLapl** and 50% for TCL-MLE. **GCL-CLapl** outperforms TCL-MLE and **GCL-DLapl** in Fig. 4-(a),-(b). The increased noise of range measurements (Fig. 4-(b)) results in deteriorated performance for all three methods. However, our schemes turn out to be more robust to noise. This is apparently due to the fact that the proposed method exploits not only the (noisy) inter-vehicle measurements but it also integrates properly the connectivity representation of VANET graph. Thus, the impact of noise could be further reduced.

We also tested our method on different number of vehicles, namely 5 and 20. In Fig. 5-(a), the reduction of error is 80% for

GCL-CLapl, 78% for **GCL-DLapl** and 71% for TCL-MLE. In Fig. 5-(b), the reduction of error is 87% for **GCL-CLapl**, 81% for **GCL-DLapl** and 82% for TCL-MLE. Once again, **GCL-CLapl** outperforms the 2 others methods. We observe also that the performance of CL methods has increased when the number of vehicles grows larger, i.e 20 vehicles. The explanation is that as the number of vehicles increases, so does the number of neighbors of each vehicle. Thus, vehicles integrate greater amount of information during the location estimation process, either with (10) or our method. We notice also that the distributed scheme **GCL-DLapl** outperforms TCL-MLE in Fig. 5-(a), while in Fig. 5-(b) have similar performance.

We are planning in the future, to apply our schemes to datasets extracted by CARLA autonomous driving simulator [16] but also to investigate the benefits of integrating the detected road landmarks to our location estimation approach.

C. Complexity

To show the computational complexity of the derived algorithms we demonstrate below in TABLE I the average execution time of 500 time instances of the 3 methods, for 10, 20 and 30 vehicles. It is obvious that **GCL-CLapl** and **GCL-DLapl** are much more faster than TCL-MLE. More specifically, **GCL-CLapl** and **GCL-DLapl** are 290 and 63, 453 and 63 and 338 and 66 times faster than TCL-MLE, for 10, 20 and 30 vehicles, respectively. As a result, in cases where **GCL-DLapl** and TCL-MLE have similar performance (Fig. 4-(a), Fig. 5-(b)), **GCL-DLapl** has the advantage of the lower complexity and execution time. Also, we observe that as the number of vehicles increases, so does the execution time, which is reasonable.

Parentheses into the cells of TABLE I refer to how many times faster than **TCL-MLE** the 2 proposed schemes are.

TABLE I
AVERAGE EXECUTION TIME ($\times 10^{-4} sec$)

| CL methods | 10 vehicles | 20 vehicles | 30 vehicles |
|------------------|--------------------|--------------------|--------------------|
| GCL-CLapl | 15 (290 \times) | 19 (453 \times) | 40 (338 \times) |
| GCL-DLapl | 69 (63 \times) | 137 (63 \times) | 204 (66 \times) |
| TCL-MLE | 4357 | 8617 | 13520 |

V. CONCLUSION

In this paper, we modeled the vehicles of a VANET as vertices of an undirected graph and the communication links between them as the edges of the graph. Each vehicle is able to know its absolute position and to measure its relative distance and angle or azimuth angle to its neighboring and connected vehicles. Thus, by creating the overall or local Laplacian matrix and defining the δ coordinates, Sparse Laplacian Localization can significantly improve vehicles' GPS accuracy. Finally, we have proven that the centralized or local ('distributed') scheme of our method, outperforms the method of [2], both in terms of accuracy and execution time. Regarding

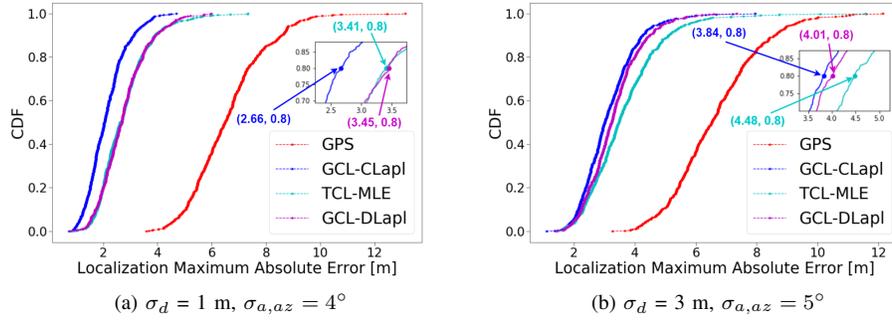


Fig. 4. CDFs of Localization Maximum Absolute Error

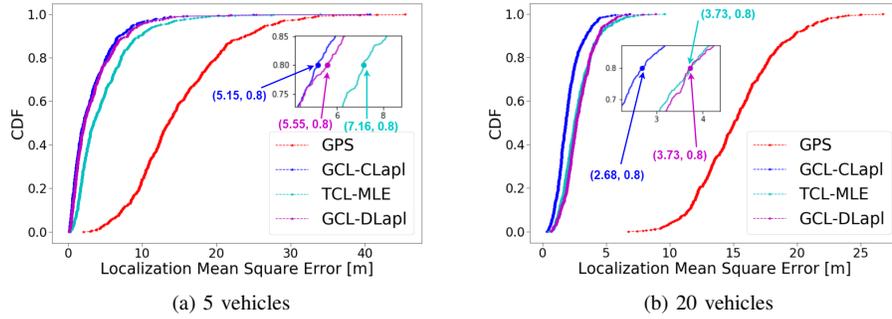


Fig. 5. CDFs of Localization Mean Square Error

the accuracy, the reduction of Mean Square and Maximum Absolute Localization error can reach 87% and 69% with our method (instead of 82% and 60% of [2]), while the gain in execution time is 3 (**GCL-CLapl**) and 2 (**GCL-DLapl**) orders of magnitude.

ACKNOWLEDGMENT

This paper has received funding from the European Union's H2020 research and innovation programme CPSoSaware under grant agreement No 871738.

REFERENCES

- [1] H. Wymeersch, J. Lien and M. Z. Win, "Cooperative Localization in Wireless Networks," in Proceedings of the IEEE, vol. 97, no. 2, pp. 427-450, Feb. 2009.
- [2] H. Kim, S. H. Lee and S. Kim, "Cooperative localization with distributed ADMM over 5G-based VANETs," 2018 IEEE Wireless Communications and Networking Conference (WCNC), Barcelona, 2018, pp. 1-5.
- [3] H. Li and F. Nashashibi, "Cooperative Multi-Vehicle Localization Using Split Covariance Intersection Filter," in IEEE Intelligent Transportation Systems Magazine, vol. 5, no. 2, pp. 33-44, Summer 2013.
- [4] F. Bounini, D. Gingras, H. Pollart and D. Gruyer, "Real time cooperative localization for autonomous vehicles," 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), Rio de Janeiro, 2016, pp. 1186-1191.
- [5] G. Hoang, B. Denis, J. Härrä and D. T. M. Slock, "Robust data fusion for cooperative vehicular localization in tunnels," 2017 IEEE Intelligent Vehicles Symposium (IV), Los Angeles, CA, 2017, pp. 1372-1377.
- [6] Mariam Elazab, Aboelmagd Noureldin, Hossam S. Hassanein, "Integrated cooperative localization for Vehicular networks with partial GPS access in Urban Canyons, Vehicular Communications", Volume 9, 2017, Pages 242-253.
- [7] João Pinto Neto, Lucas Gomes, Fernando Ortiz, Thales Almeida, and Miguel Elias Mitre Campista, "An Accurate Cooperative Positioning System for Vehicular Safety Applications", Computers and Electrical Engineering, Elsevier, 2020.
- [8] G. Soatti, M. Nicoli, N. Garcia, B. Denis, R. Raulefs and H. Wymeersch, "Implicit Cooperative Positioning in Vehicular Networks," in IEEE Transactions on Intelligent Transportation Systems, vol. 19, no. 12, pp. 3964-3980, Dec. 2018.
- [9] M. Rohani, D. Gingras, V. Vigneron and D. Gruyer, "A New Decentralized Bayesian Approach for Cooperative Vehicle Localization Based on Fusion of GPS and VANET Based Inter-Vehicle Distance Measurement," in IEEE Intelligent Transportation Systems Magazine, vol. 7, no. 2, pp. 85-95, Summer 2015.
- [10] S. Kuutti, S. Fallah, K. Katsaros, M. Dianati, F. McCullough and A. Mouzakitis, "A Survey of the State-of-the-Art Localization Techniques and Their Potentials for Autonomous Vehicle Applications," in IEEE Internet of Things Journal, vol. 5, no. 2, pp. 829-846, April 2018.
- [11] N. Alam and A. G. Dempster, "Cooperative Positioning for Vehicular Networks: Facts and Future," in IEEE Transactions on Intelligent Transportation Systems, vol. 14, no. 4, pp. 1708-1717, Dec. 2013.
- [12] Slavisa Tomic, Marko Beko, Luís M. Camarinha-Matos, and Luís Bica Oliveira, "Distributed Localization with Complemented RSS and AOA Measurements: Theory and Methods", in Applied Sciences 10(1):272, December 2019.
- [13] Olga Sorkine, "Laplacian Mesh Processing", Eurographics 2005.
- [14] David Chin-Lung Fong and Michael Saunders, "LSMR: An iterative algorithm for sparse least-squares problems", SIAM J. Sci. Comput., Vol. 33, No. 5, pp. 2950-2971, 2011.
- [15] J. Kümmerle, M. Sons, F. Poggenhans, T. Kühner, M. Lauer and C. Stiller, "Accurate and Efficient Self-Localization on Roads using Basic Geometric Primitives," 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 2019, pp. 5965-5971.
- [16] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An open urban driving simulator. In Conference on Robot Learning (CoRL), 2017.