Multi-Vehicle System Localization by Distributed Moving Horizon Estimation over a Sensor Network with Sporadic Measurements

Antonello Venturino, Sylvain Bertrand, Cristina Stoica Maniu, Teodoro Alamo, Eduardo Fernandez Camacho

Multi-Vehicle System Localization by Distributed Moving Horizon Estimation over a Sensor Network with Sporadic Measurements

Antonello Venturino, Sylvain Bertrand, Cristina Stoica Maniu, Teodoro Alamo, Eduardo F. Camacho

***Abstract*— This paper proposes a Distributed Moving Horizon Estimator (DMHE) for the Multi-Vehicle system localization problem using Sensor Networks with sporadic measurements. Due to its capability to efficiently exploit environmental information via constraints, the proposed DHME technique is well-suited to better estimate the system state when measurements are available at time instants *a priori* unknown. Indeed, the use of output constraints can contribute to locally improve estimation accuracy, especially when dealing with sporadic measurements and biased sensors data. A realistic case study is proposed within the Robot Operating System framework and Gazebo to localize a Multi-Vehicle System using an inexpensive Sensor Network with low-computation capabilities. A comparative campaign simulation is performed to confirm the effectiveness of the proposed DMHE algorithm in terms of accuracy, computation time, and constraints handling with respect to existing results.**

I. Introduction

Numerous studies have been dedicated to Distributed State Estimation (DSE) over Sensor Network (SN) [1]–[3] during the last few years since these schemes are suitable for diverse applications and contexts. Some examples focus on detecting and mitigating cyber-attacks [4], tracking intruders in a safe area [1], [5], estimating the state of large-scale systems [6], [7], mobile robot localization [8], [9], etc. Some of these works have conducted only theoretical developments or have exclusively numerically shown the effectiveness of the considered techniques. Indeed, there is still a judicious need for deep insights such as applying algorithms in real experiments and applications. For example, communication delays and losses [10], computation time [5], the time-varying topology of the network [9], sporadic measurements [11], [12] are still open problems to cope with in theory and much more in practice.

Distributed State Estimation with a Moving Horizon Estimator (MHE) is one particular case for which the computation time of the state estimation algorithm matters since it involves a constrained optimization problem to be solved at each time instant [13]. The MHE paradigm consists in using an estimation window of fixed size, which moves forward in time hence the problem remains computationally tractable since only the most recent information is processed and the past evolution of the system is summarized in its so-called *arrival cost* [14].

The current paper focuses on Multi-Vehicle System (MVS) localization using Distributed MHE (DMHE) algorithms. Similar works have been conducted by the authors of [8] and [9]. In [8], the DMHE problem has been addressed by focusing on the non linearity of the model and on the possible local observability issues at the sensor level. In [9], the authors accounted for mobile nodes in the Sensor Network that led to deal with a dynamic topology. Indeed, using a flocking algorithm for the motion control, the mobile sensors attempt to move in a specific way in order to get the best positions to observe the target and to avoid collisions between neighboring agents. In the current paper, we focus on the computation time aspect, which is a key factor for real-time implementation.

In previous research works by the authors [5], [15], [16], DMHE algorithms with pre-estimation have been proposed in order to reduce the computation time while preserving or improving the accuracy of the state estimation. To this aim, the input sequence of noise to be estimated has been replaced by a Luenberger observer leading to fewer optimization parameters to be accounted. Furthermore, an observability rank-based weights technique is used to enhance the accuracy. The contribution of the current paper is two-fold. First, in addition to a reduced computation time and an improved accuracy due to the pre-estimation, the proposed DMHE technique is designed for realistic large-scale systems scenarios involving sporadic measurements (i.e. available at time instants *a priori* unknown). To this aim, constraints on measurements (coming from the knowledge of the environment where the Multi-Vehicle System is evolving) are embodied using binary parameters in this novel Distributed Moving Horizon Estimation formulation. Thus, the environment information is exploited to better estimate the system state. Second, the current paper aims at evaluating the performance of the proposed DMHE approach (in terms of accuracy and computation time) on a realistic case study, i.e. the distributed localization of a Multi-Vehicle System by a static sensor network, developed within the Robot Operating System (ROS) framework and Gazebo environment. This realistic distributed implementation within ROS and Gazebo would enable the deployment on a hardware setup. To confirm its efficiency, the proposed DMHE constrained formulation is compared with the notable DMHE algorithm [2].

The paper is structured as follows. Section II describes

A. Venturino and C. Stoica Maniu are with Université Paris-Saclay, CNRS, CentraleSupélec, Laboratoire des signaux et systèmes, 91190, Gif-sur-Yvette, France (e-mail: {antonello.venturino; cristina.stoica}@l2s.centralesupelec.fr).

A. Venturino and S. Bertrand are with Université Paris-Saclay, ONERA, Traitement de l'information et systèmes, 91123, Palaiseau, France (e-mail: {antonello.venturino; sylvain.bertrand}@onera.fr).

T. Alamo and E.F. Camacho are with Department of Ingeniería de Sistemas y Automática, Universidad de Sevilla, Camino de los Descubrimientos, 41092 Sevilla, Spain (e-mail: talamo@us.es, eduardo@esi.us.es).

the problem under investigation and introduces the main ingredients for the algorithm. The proposed DMHE algorithm is presented in Section III. Before concluding remarks (Section V), a realistic simulation scenario within the ROS and Gazebo environments is investigated in Section IV.

II. Distributed State Estimation over Sensor Network with Sporadic Measurements

This section describes the problem of Distributed State Estimation (DSE) of a Multi-Vehicle System (MVS) over a Sensor Network (SN) with sporadic measurements, the considered model, and the Sensor Network with its characteristics.

A. Problem description

The Distributed State Estimation algorithm uses a Sensor Network to estimate the states of the Multi-Vehicle System. In this context, we assume that the Sensor Network is composed of possibly n_S heterogeneous sensors with sporadic measurements, i.e. the measurements may not be (partially or entirely) available all the time to each sensor. For instance, a mobile vehicle is detected by a camera only when it is within its field of view, or by a beacon when it is within its range, etc. A (formation of) vehicle(s) moving in an unknown direction can thus be detected by a given sensor belonging to a Sensor Network at time instants *a priori* unknown.

The system under observation is composed of n_V interconnected ground vehicles which are limited to moving in specific areas, e.g. in an urban area, they can only move on the road (see Fig. 1). We further exploit this information as position constraints in the Distributed State Estimation optimization problem.

B. Considered model

The dynamical model of the ν-th vehicle (denoted by the left superscript) is described as a linear time-invariant (LTI) system

$$ {}^{\nu}x_{t+1} = {}^{\nu}A \ {}^{\nu}x_t + {}^{\nu}w_t, \qquad \nu = 1, \dots, n_V \tag{1} $$

where ${}^{\nu}x_t \in {}^{\nu}\mathcal{X} \subseteq \mathbb{R}^{{}^{\nu}n_x}$ is the state and ${}^{\nu}w_t \in {}^{\nu}\mathcal{W} \subseteq \mathbb{R}^{{}^{\nu}n_x}$ is an exogenous input (e.g. an unknown control input, state perturbation, etc.), with ${}^{\nu}\mathcal{X}$ and ${}^{\nu}\mathcal{W}$ convex sets.

Remark 1: Notice that, when referring to the global Multi-Vehicle System, the left superscript ν is omitted, e.g. $x_t = [{}^{1}x_t^{\top}, \dots, {}^{n_V}x_t^{\top}]^{\top}$ or $A = \text{diag}({}^{1}A, \dots, {}^{n_V}A)$.

Since each sensor i (right superscript) can provide measurements on each vehicle individually. The following mathematical expression[1] refers to the output carried out by sensor i with respect to the ν-th vehicle

$$ {}^{\nu}y_t^i = {}^{\nu}C^i \ {}^{\nu}x_t + {}^{\nu}v_t^i, \qquad i = 1, \dots, n_S \tag{2} $$

where ${}^{\nu}y_t^i \in \mathbb{R}^{n_y^i}$ is the measurement vector and ${}^{\nu}v_t^i \in \mathbb{R}^{n_y^i}$ the measurement noise.

Remark 2: Notice that in (2), the right superscript i refers to the i-th sensor and the left superscript ν to the ν-th vehicle.

[1]Notice that a linear equation is used for the output vector.

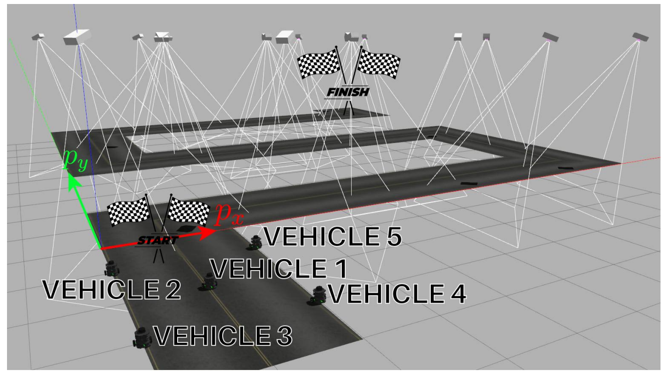

Fig. 1: Scenario illustrated in Gazebo: MVS with 5 vehicles in the starting place.

In this respect, ${}^{\nu}C^i$ is the output matrix specifying that the sensor i is measuring the vehicle ν position. The notation C^i without the left superscript ν refers to the output matrix of the global Multi-Vehicle System $C = \text{diag}({}^{1}C^i, \dots, {}^{n_V}C^i)$, similar to Remark 1.

For the sake of simplicity in exposing the rest of the paper and without losing generality, we assume that each sensor can measure only the position of the vehicles.

The following notation is necessary to denote the collective output matrix of the global system allowing to aggregate both the measuring and non-measuring situations of each sensor

$$ C^i_{\alpha_t} = D^i_{\alpha_t} C^i \tag{3} $$

where $D^i_{\alpha_t}$ is a diagonal matrix having ${}^{\nu}\alpha_t^i \in \{0, 1\}$ as components, with $\nu = 1, \dots, n_V$, leading to

$$ D^i_{\alpha_t} = \text{diag}({}^{1}\alpha_t^i \ I_{{}^{1}n_y}, \dots, {}^{n_V}\alpha_t^i \ I_{{}^{n_V}n_y}) \tag{4} $$

with $I_{{}^{\nu}n_y}$ the identity matrix of size ${}^{\nu}n_y$.

Remark 3: Notice that ${}^{\nu}\alpha_t^i$ is a time-dependent binary parameter indicating if the sensor i is able to measure the ν-th vehicle at time t (i.e. ${}^{\nu}\alpha_t^i = 1$) or not (i.e. ${}^{\nu}\alpha_t^i = 0$).

C. Constraints

This subsection defines measurement constraints exploiting the *a priori* knowledge of the environment and the Sensor Network. First, denote by $\mathcal{R}$ the subset of positions corresponding to the road (assumed to be non-convex) on which the vehicles can move. Then, denote by $\mathcal{F}^i$ the set of the points forming the sensor i field of view. The convex hull of the intersection of these two sets denoted by

$$ \mathcal{S}^i = \text{Co}(\mathcal{R} \cap \mathcal{F}^i) \tag{5} $$

is further used to constrain the position of the vehicle when the mobile vehicle is within the field of view of the sensor i (see Fig. 2 for a graphic representation).

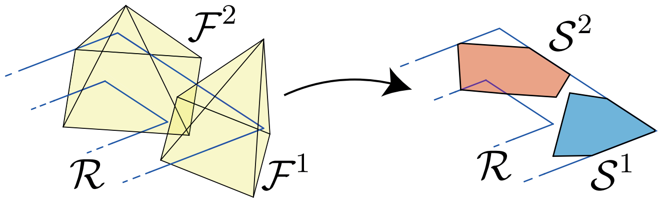

Fig. 2: Road $\mathcal{R}$ (blue line), fields of view $\mathcal{F}^1$ and $\mathcal{F}^2$ (yellow), and convexified constraints $\mathcal{S}^1$ (red polytope) and $\mathcal{S}^2$ (blue polytope).

D. Sensor network

In Distributed State Estimation schemes, the nearby sensors share data among each other. The Sensor Network is described by a directed graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$, where $\mathcal{N} = \{1, 2, \ldots, n_S\}$ is the set of all nodes (sensors) and $\mathcal{E} \subseteq \mathcal{N} \times \mathcal{N}$ is the set of all edges (communication links). Therefore, the pair $(i, j) \in \mathcal{E}$ exists if and only if the sensor j can receive information from the sensor i. The *neighborhood* $\mathcal{N}^i$ of the sensor i is defined as $\mathcal{N}^i = \{j \in \mathcal{N} : (i, j) \in \mathcal{E}\}$ and its cardinality $n_S^i = \text{card}(\mathcal{N}^i)$. In this paper, we consider that the topology of the Sensor Network is fixed.

We distinguish *local* information, i.e. referring only to the local sensor i, and *regional* information, i.e. referring to the entire neighborhood $\mathcal{N}^i$. A general bar notation $(\bar{\cdot})$ is used to denote the regional information, e.g. the regional output of sensor i at time t is $\bar{y}_t^i = [(y_t^i)^\top, (y_t^{j_1})^\top, \ldots, (y_t^{j_{n_S^i}})^\top]^\top$, $\{j_1, \ldots, j_{n_S^i}\} \in \mathcal{N}^i$.

The edges of the graph $\mathcal{G}$ are weighted by the components of a stochastic matrix K, those values are given as follows

$$k_{ij} > 0 \quad \text{if } (j, i) \in \mathcal{E}, \tag{6a}$$

$$k_{ij} = 0 \quad \text{otherwise}, \tag{6b}$$

$$\sum_{j=1}^{n_S} k_{ij} = 1, \quad \forall i = 1, \ldots, n_S. \tag{6c}$$

The values of k_{ij} can be chosen according to some criteria. For instance, in [16] the authors proposed a rank-based method that leads to a better accuracy for the estimations since K will be used in the DMHE algorithm to compute the consensus terms, as described further on in Section III.

E. Problem statement

Consider the discrete-time LTI system (1) and the sensor network $\mathcal{G}$ with the linear measurement equation (2), under the assumption that the graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$ is strongly connected, i.e. every node is reachable from every other node. The role of each sensor $i \in \mathcal{N}$, at each time t, is to (possibly) get measurement on (part of) the Multi-Vehicle System, to exchange information among neighbor nodes of $\mathcal{N}^i$ and to process locally available information in order to determine a local estimate $\hat{x}_t^i$ of the real state x_t of the Multi-Vehicle System.

III. Constrained DMHE with sporadic measurements for Multi-Vehicle Systems

This section recalls the Distributed Moving Horizon Estimation approach with pre-estimation and observability rank-based weights proposed in [16] and presents its novel formulation to handle the Multi-Vehicle localization application considered in the current paper. Thanks to the pre-estimation, the proposed DMHE technique reduces the computation time needed by the sensors (see [16]) to estimate the state of the system, compared to classical DMHE (see [2]). In fact, by replacing the input sequence of the noise to be estimated, the pre-estimation observer reduces the computation time required to solve the optimization problem due to a reduced number of optimization variables. Moreover, thanks to the observability rank-based weights, the accuracy of the estimates is improved and thus it enables the use of classic DMHEs for sporadic measurements.

A. Local optimization problem

At each time t, given an estimation horizon length $N \geqslant 1$, each sensor $i \in \mathcal{N}$ determines the estimate $\hat{x}_{t|t}^i$ of the state $x_{t|t}$ by solving the following constrained minimization problem with pre-estimation:

$$\hat{x}_{t-N|t}^i = \arg \min_{\hat{x}_{t-N}^i} \quad J_{\alpha_t}^i(\cdot) \tag{7}$$

$$\text{s.t.} \quad \hat{x}_{k+1}^i = A\, \hat{x}_k^i + L_{\alpha_k}^i\, \hat{v}_k^i, \tag{8}$$

$$\bar{y}_k^i = \bar{C}^i\, \hat{x}_k^i + \hat{\bar{v}}_k^i, \tag{9}$$

$$\hat{x}_k^i \in \mathcal{X} \cap \mathcal{S}_{\alpha_k}^i, \tag{10}$$

$$\forall k = t - N, \ldots, t.$$

Notice that in (8), the estimate of the measurement noise $\hat{v}_k^i$ is local, while in (9) $\hat{\bar{v}}_k^i$ is regional. The A matrix in (8) refers to the global Multi-Vehicle System. The sequence of state estimates $\hat{x}_{t-N+1|t}^i, \ldots, \hat{x}_{t|t}^i$ is obtained from the optimal solution $\hat{x}_{t-N|t}^i$ and using the dynamics (8). The main novelty w.r.t. [16] concerns the use of the binary parameter ${}^{\nu}\alpha_t^i$ in (7) and (8), as it is detailed in the next paragraph. In addition, the constraints (10) are incorporated within the optimization problem as explained in section II-C.

The parameter ${}^{\nu}\alpha_t^i$ allows to deal with the sporadic measurements. First, it is useful to discern when the constraints $\mathcal{S}^i$ are used by sensor i and when not. In particular, considering ${}^{\nu}\mathcal{S}_{\alpha_t}^i$, i.e. the projection of $\mathcal{S}_{\alpha_t}^i$ into the subspace related to the ν-th vehicle leads to

$$\begin{cases} {}^{\nu}\mathcal{S}_{\alpha_t}^i = & {}^{\nu}\mathcal{S}^i & \text{if } {}^{\nu}\alpha_t^i = 1 \\ {}^{\nu}\mathcal{S}_{\alpha_t}^i = & \mathbb{R}^{{}^{\nu}n_x} & \text{if } {}^{\nu}\alpha_t^i = 0 \end{cases}$$

The Luenberger gain L^i is computed such that $\Phi^i = A - L^i C^i$ is Schur stable when the Multi-Vehicle System is observable by sensor i. One may compute the gain associated to the global MVS or separately, since $L^i = [{}^{1}L^i, \ldots, {}^{n_V}L^i]$. In addition, the dependence on ${}^{\nu}\alpha_t^i$ is formulated via $L_{\alpha_t}^i = L^i D_{\alpha_t}^i$, with $D_{\alpha_t}^i$ defined by (4).

The binary parameter ${}^{\nu}\alpha_t^i$ appears also in the objective function $J_{\alpha_t}^i$ defined as

$$J_{\alpha_t}^i(\cdot) = \frac{1}{2}\sum_{k=t-N}^{t} \left\| \bar{y}_k^i - \bar{C}^i\,\hat{x}_k^i \right\|_{\bar{R}_{\alpha_t}^i}^2 + \Gamma_{t-N}^i(\cdot), \qquad (11)$$

where the weight matrix $\bar{R}_{\alpha_t}^i = (\bar{R}^i)^{-1} D_{\alpha_t}^i$ can be chosen as the product between the covariance matrix of the measurement noise, and the diagonal matrix $D_{\alpha_t}^i$ of appropriate dimensions. The last term of (11) is the *initial penalty* function $\Gamma_t^i(\cdot)$, known in the MHE paradigm as *arrival cost*. This terms is non negative and it summarizes the effect of the past measurements, before time $t-N$. The initial penalty function $\Gamma_t^i(\cdot)$ in (11) defined as follows:

$$\Gamma_t^i(\cdot) = \frac{1}{2}\left\| \hat{x}_{t-N}^i - \hat{\bar{x}}_{t-N|t-1}^i \right\|_{(\bar{\Pi}_{t-N|t-1}^i)^{-1}}^2, \qquad (12)$$

involves two consensus terms described below.

We denote by $\hat{\bar{x}}_{t-N|t-1}^i$ the weighted average state estimation computed by the neighborhood $\mathcal{N}^i$ as follows:

$$\hat{\bar{x}}_{t-N|t-1}^i = \sum_{j\in\mathcal{N}^i} k_{ij}\hat{x}_{t-N|t-1}^j, \qquad (13)$$

where $\hat{x}_{t-N|t-1}^j$ is the second estimated state in the sequence computed at the previous time by sensor j. Notice that the penalty function Γ_{t-N}^i includes a *consensus-on-estimates* term, in the sense that it penalizes deviations of $\hat{x}_{t-N}^i$ from $\hat{\bar{x}}_{t-N|t-1}^i$. It helps to improve the accuracy of the local estimates and it is necessary to guarantee convergence of the state estimates to the state of the observed system even if it lacks of regional observability [2].

The positive definite matrix $\bar{\Pi}_{t-N|t-1}^i$ is computed as in [2]. For the sake of completeness, we recall here the procedure to compute it by:

$$\bar{\Pi}_{t-N|t-1}^i = \sum_{j\in\mathcal{N}^i} n_S^j k_{ij}^2 \Pi_{t-N|t-1}^j, \qquad (14)$$

where the update of $\Pi_{t-N|t-1}^i$ is performed by the sensor i on the basis of regionally available information. In particular, the matrix $\Pi_{t-N|t-1}^i$, with $i\in\mathcal{N}$, is given by one iteration of the difference Riccati equation associated to a Kalman filter for the system:

$$\begin{cases} x_{t-N} = Ax_{t-N-1} + w_{t-N-1} \\ \bar{z}_{t-N}^i = \bar{\mathcal{O}}_N^i x_{t-N} + \bar{V}_{t-N}^i \end{cases}$$

where $\bar{V}_{t-N}^i$ represents the measurement noise and $\bar{\mathcal{O}}_N^i$ defines the i-th sensor regional observability matrix:

$$\bar{\mathcal{O}}_N^i = \begin{bmatrix} (\bar{C}^i)^\top & (\bar{C}^i A)^\top & \cdots & (\bar{C}^i A^{N-1})^\top \end{bmatrix}^\top. \qquad (15)$$

Then defining:

$$\mathcal{S}_N^i = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ \bar{C}^i & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \bar{C}^i A^{N-2} & \bar{C}^i A^{N-3} & \cdots & \bar{C}^i \end{bmatrix} \in \mathbb{R}^{\bar{p}_i N\times n(N-1)}, \qquad (16)$$

$$\bar{R}_N^i = \mathrm{diag}(\bar{R}^i,\ldots,\bar{R}^i) \in \mathbb{R}^{\bar{p}_i N\times \bar{p}_i N}, \qquad (17)$$

$$Q_{N-1} = \mathrm{diag}(Q,\ldots,Q) \in \mathbb{R}^{n(N-1)\times n(N-1)}, \qquad (18)$$

$$\mathrm{Cov}[\bar{V}_t^i] = \bar{R}_N^{*i} = \bar{R}_N^i + \mathcal{S}_N^i Q_{N-1} (\mathcal{S}_N^i)^\top, \qquad (19)$$

and setting the covariance of the estimate $\hat{x}_{t-N-1}^i$ as:

$$\Pi_{t-N-1|t-2}^{*i} = \left(\left(\bar{\Pi}_{t-N-1|t-2}^i\right)^{-1} + (\bar{C}^i)^\top (\bar{R}^i)^{-1}\bar{C}^i \right)^{-1}, \qquad (20)$$

the resulting Riccati recursive equation is given by:

$$\begin{aligned} \Pi_{t-N|t-1}^i = {} & A\Pi_{t-N-1|t-2}^{*i}A^\top + Q - A\Pi_{t-N-1|t-2}^{*i}\left(\bar{\mathcal{O}}_N^i\right)^\top \\ & \cdot \left(\bar{\mathcal{O}}_N^i \Pi_{t-N-1|t-2}^{*i}\left(\bar{\mathcal{O}}_N^i\right)^\top + \bar{R}_N^{*i} \right)^{-1} \\ & \cdot \bar{\mathcal{O}}_N^i \Pi_{t-N-1|t-2}^{*i} A^\top. \end{aligned} \qquad (21)$$

Since the communication network topology is assumed to be time-invariant, these equations can be computed off-line. However, once the matrices $\Pi_{t-N|t-1}^i$ have been computed, we perform a *consensus weights update* in order to compute the matrices $\bar{\Pi}_{t-N|t-1}^i$ according to (14).

B. Observability rank-based weights technique

Here, we briefly recall the weights tuning technique in [16], for the stochastic matrix K associated to the graph $\mathcal{G}$ and highlight its adjustment for the considered Multi-Vehicle localization problem by DMHE over a Sensor Network with sporadic measurements.

This method needs only *local* information available by each sensor to compute its own component of K. Thus, it is suitable for a distributed scheme, and more important for this application context, where the measurements are sporadic. Indeed, this technique enhances the accuracy of the estimates by means of exploiting observability properties of the neighborhoods. Since these properties are changing over time for this Sensor Network, the observability rank-based weights technique is an appropriate method to improve even more the accuracy and the convergence of the algorithm.

Consider a sensor i at time t. Its regional observability matrix

$$\bar{\mathcal{O}}_n^i = \begin{bmatrix} (\bar{C}_{\alpha_t}^i)^\top & (\bar{C}_{\alpha_t}^i A)^\top & \cdots & (\bar{C}_{\alpha_t}^i A^{n-1})^\top \end{bmatrix}^\top \qquad (22)$$

is of full rank if and only if the the pair $(A, \bar{C}_{\alpha_t}^i)$ is completely observable, i.e. $\mathrm{rank}(\bar{\mathcal{O}}_n^i) = n$. For the sake of simplicity, we denote by $\rho_{\mathcal{O}}^i = \mathrm{rank}(\bar{\mathcal{O}}_n^i)$. This information could be used as reliability of sensor i when choosing the weights, which according to (6) must be averaged among the neighbors. Moreover, since at some time instants *a priori* unknown, the entire neighborhoods could not have sensing capabilities at all, i.e. $\rho_{\mathcal{O}}^i = 0$. To avoid division by zero a lower bound smaller than 1 (0.5 in (23)) is chosen for k_{ij}, which results in

$$k_{ij} = \begin{cases} 0.5 & \text{if } \sum_{j\in N^i} \rho_{\mathcal{O}}^j = 0 \\ \frac{\rho_{\mathcal{O}}^j}{\sum_{j\in\mathcal{N}^i} \rho_{\mathcal{O}}^j} & \text{otherwise} \end{cases} \tag{23}$$

C. DMHE modus operandi

Finally, the procedure of the proposed distributed scheme is described in Algorithm 1.

Algorithm 1 DMHE procedure

1: **Off-line:** $\forall i \in \mathcal{N}$
2: **receive** from the nodes $j \in \mathcal{N}^i$: L^j, C^j, R^j
3: **compute** the pre-observer gain L^i
4: **store** the *a priori* initial estimation $\hat{x}_{0|0}^i = \hat{x}_0$ of x_0, where $\hat{x}_0$ is given, and the covariance matrix Π_0 of x_0
5: **Initialization:** $\forall i \in \mathcal{N}$, at the first time step $t = 0$
6: **collect** a first local measurement y_0^i
7: **receive** from the neighborhood $j \in \mathcal{N}^i$ their measurements y_0^j
8: **Online:** $\forall i \in \mathcal{N}$, $\forall t > 0$
9: **collect** the local measurement y_t^i
10: **receive** from the neighbors $j \in \mathcal{N}^i$ the collected data in the step 9
11: **compute** the $D_{\alpha_t}^i$ matrix according to (4)
12: **compute** the k_{ij} components according to (23)
13: **if** $1 \leqslant t \leqslant N$ **then**
14: **set** the horizon length $N = t$, the covariance matrix $\bar{\Pi}_{t-N|t-1}^i = \bar{\Pi}_{0|t-1}^i = \Pi_0$ and the *a priori* initial estimation state $\hat{x}_{t-N|t-1}^i = \hat{x}_{0|t-1}^i$
15: **else**
16: **compute** $\Pi_{t-N|t-1}^i$ according to (19), (20) and (21)
17: **receive** $\Pi_{t-N|t-1}^j$ from the nodes $j \in \mathcal{N}^i$
18: **compute** $\bar{\Pi}_{t-N|t-1}^i$ according to (14)
19: **solve** the local optimization problem of DMHE, minimizing J^i as in (11) and (12) subject to the constraints (8)-(10)
20: **store** the solution $\hat{x}_{t-N|t}^i$ and the corresponding estimate $\hat{x}_{t|t}^i$
21: **receive** from the neighbors $j \in \mathcal{N}^i$ their estimates $\hat{x}_{t-N+1|t}^j$

The sporadic measurements constraints are integrated at step 19, with $D_{\alpha_t}^i$ computed at step 11. Notice that the steps 10, 18 and 21 in the procedure regarding the exchanging information could be rearranged to include only one synchronization. However, the current formulation has been chosen for clarity reasons w.r.t. calculation details.

IV. Realistic simulations

A. Scenario and simulation setup

In this section, the proposed DMHE is applied to estimate the positions of a team of $n_V = 5$ ground vehicles moving together. To evaluate its performance a realistic implementation in the ROS framework and in the Gazebo environment is proposed, see Fig. 1 and the associated video available at `https://youtu.be/KRvlQgvHGEo`. For the estimation models used in the DMHE optimization problem, each vehicle is modeled as single integrator, with a 2-dimensional state vector representing its Cartesian positions in the plane. The control input vector, i.e. Cartesian linear velocities, of each vehicle is assumed to be unknown, and it is further considered as an exogenous input, i.e. ${}^{\nu}w_t \in \mathbb{R}^2$ a uniformly distributed noise with covariance matrix $Q = I_2$.

To simulate a realistic system, each vehicle is modeled in Gazebo as a differential drive robot (TurtleBot3). The Multi-Vehicle System goes from the starting point $(1, -2)$ m towards the final point $(11, 11)$ m driving within the road and controlled by a leader-follower formation control strategy.

The Sensor Network (SN) is composed of $n_S = 17$ cameras measuring the Cartesian positions of the vehicles, and connected as in the Fig. 3 (see the red edges representing the communication links between the nodes depicting the cameras). Fig. 3 also shows the projection on the ground of the field of view of each camera (yellow rectangles), and the road (blue solid line). The start and finish position of the Multi-Vehicle System are clearly indicated in Figs. 1 and 3.

Notice that the graph associated with the Sensor Network is not a complete graph, i.e. a graph in which every pair of distinct vertices is connected by a unique edge. The measurements refer to the reference frame associated with each camera. Thus, to have them in the absolute reference frame it is necessary to translate and rotate them with a transformation matrix. To make the scenario more realistic, we added different biases for each camera (via the measurement equation (2) for each sensor) on these translations and rotations, allowing to model uncertainties related to the cameras' poses.

Assuming that these biases can not be easily estimated and compensated in the considered scenario, the purpose is to investigate the robustness of the proposed Distributed Moving Horizon Estimation to this additional source of uncertainty (i.e. sensor biases) and to validate the usefulness of *a priori* known environment constraints considered in the DMHE optimization problem.

A Monte Carlo simulation with 100 runs with different measurements noises (per run) normally distributed, i.e. ${}^{\nu}v_t^i$ is a white noise with zero mean and covariance matrix $R^i = 0.5I_2$ was performed. The estimators runs with a sampling time $T_s = 0.5$ s and a horizon length $N = 4$. The initial values of the algorithms have been set as ${}^{\nu}\hat{x}_0 = \begin{bmatrix} 0 & 0 \end{bmatrix}^\top$, $\Pi_0 = 10^5 I_2$. The optimization problem was implemented by using the quadratic programming solver from [17] implemented in Python. The considered performance indexes are the computation time τ needed by the solver to estimate the positions of the Multi-Vehicle System, and the Root Mean Square Error (RMSE) computed as follows

$$\text{RMSE}_t = \frac{1}{100 \cdot n_S} \sum_{z=1}^{100} \sum_{i\in\mathcal{N}} \left\| x_t(z) - \hat{x}_{t|t}^i(z) \right\|,$$

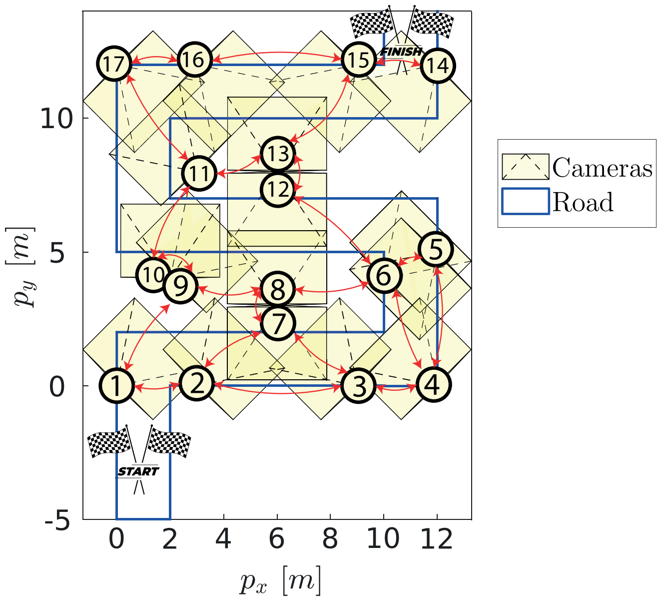

Fig. 3: Simulation scenario with road (blue line), cameras (numbered nodes), field of view of each camera (yellow rectangles), and communication link between the sensors (red arrows).

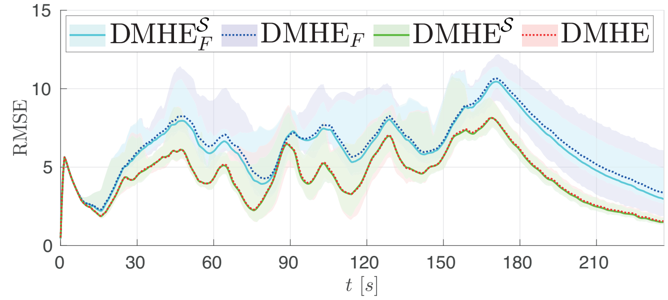

Fig. 4: Averaged RMSE among all the sensors and all the trials.

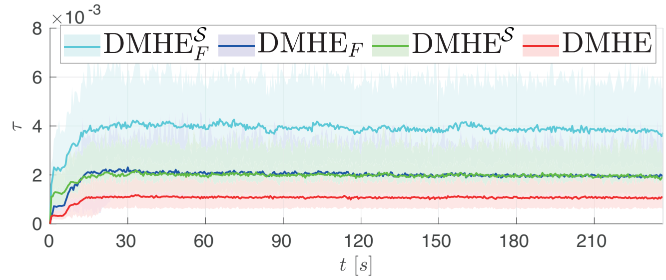

Fig. 5: Averaged computation time τ among all the sensors and all the trials.

both averaged among the trials and the sensors, where $x_t(z)$ and $\hat{x}^i_{t|t}(z)$ are, respectively, the realization of the real state of the system and the estimated one, by sensor i, for the trial z. The simulation is carried out by a PC Linux Ubuntu 20.04 equipped with an Intel i9-11950H processor.

We compare the proposed Distributed Moving Horizon Estimation without constraints (denoted by DMHE) and with constraints[2] (denoted by DMHE$^{\mathcal{S}}$). We also compare the results with the algorithm in [2], denoted hereafter by DMHE$_F$ for the unconstrained case. We also added the constraints (5) to this approach, denoted hereafter by DMHE$_F^{\mathcal{S}}$.

B. Results' analysis

Figure 4 illustrates the averaged RMSE among all the sensors and all the 100 trials. It shows that the proposed DMHE (red dotted curve) and DMHE$^{\mathcal{S}}$ (solid green curve) have better accuracy w.r.t. to the approach in [2], with constraints (solid cyan curve) or without constraints (dark blue dotted curve). Indeed, the RMSE obtained with the proposed estimation approaches (both DMHE and DMHE$^{\mathcal{S}}$) are improved by a factor close to 30% w.r.t. the RMSE of DMHE$_F$ and DMHE$_F^{\mathcal{S}}$. This figure shows also the bounds (shaded colors) representing the minimum and the maximum RMSE of each trial and for each individual local observer.

The same trend can be seen in Fig. 5 showing the computation time τ averaged among all the sensors and trials. Accounting for constraints is done at the cost of an increase of the computation time (close to a factor 2). The proposed pre-estimation mechanism enables to compensate that by drastically reducing the computation time. Here the bounds, representing the minimum and the maximum τ of each trial and for each individual local observer of the proposed DMHEs are tighter than the bounds obtained with the DMHE$_F$ (see [2]) and DMHE$_F^{\mathcal{S}}$.

The constraints $\mathcal{S}^i$ are used in the local optimization problem only when the camera is actually sensing a vehicle. In order to show the effects of these constraints $\mathcal{S}^i$ (5), we picked up, for one random trial, the estimations of the position along the x-axis for the fourth vehicle. In particular, we consider to plot only the estimations during the time periods when their sensors are active, i.e. when the vehicle belongs to its field of view. Thus, Fig. 6 illustrates the real position ${}^4p_{x,t}$ (in green solid line) coming from the true localization on the ground provided by the Gazebo simulator, the measurements of the cameras (cyan dots), the estimations using DMHE (red dotted curves) and DMHE$^{\mathcal{S}}$ (dark blue dotted curves). The zoomed parts also show the local constraints (black dashed lines). Due to the measurement noise and bias, some measurements (cyan dots) could not correspond to possible positions of the vehicle which are constrained to be within the road boundaries. Accounting explicitly for constraints in the estimation helps to improve the accuracy. Figure 6 illustrates that the estimations with the DMHE$^{\mathcal{S}}$ (dark blue dotted curves) method respect the constraints represented by black dashed lines in the zoomed parts.

V. Conclusion and perspectives

In this paper, we proposed a Distributed Moving Horizon Estimation (DMHE) algorithm for localizing a Multi-Vehicle System (MVS) over a static sensor camera network with sporadic measurements, i.e. available at time instants *a*

[2]The constraints are added as in (5).

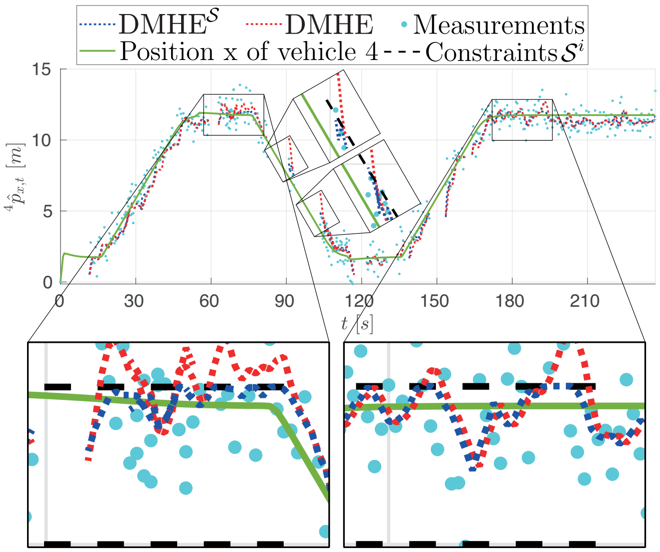

Fig. 6: Estimation of the position along the x-axis of the fourth vehicle $^4\hat{p}_{x,t}$ by all the active sensors.

priori unknown. The proposed approach, which takes into account measurement constraints, has been implemented in a realistic distributed way in the Robotic Operator System (ROS) middleware with a Gazebo simulation environment. Indeed, thanks to the pre-estimation observer in the optimization problem, the proposed DMHE has shown half of the computation time needed by [2], since it replaces the input sequence of the noise to be estimated, thus leading to fewer optimization parameters. Moreover, for the reason that an optimization problem is used in the MHE paradigm, the proposed DMHE is prone to exploit *a priori* information as constraints to better estimate the state of the system. The proposed DMHE formulation is able to deal with sporadic measurements due to binary parameters inserted into the algorithm. In addition, it increases the accuracy of the estimation as a consequence of using the observability rank-based method to tune the components of the consensus matrix associated with the graph of the Sensor Network. This particular aspect is suitable in conditions of sporadic measurements.

Current research work focuses on implementing the proposed method on hardware Multi-Vehicle System, involving TurtleBot3 mobile vehicles, and an inexpensive sensor camera network.

VI. Acknowledgment

This work was supported by the Agencia Estatal de Investigación (AEI)-Spain under Grant PID2019-106212RB-C41/AEI/10.13039/501100011033 and the European Research Council under the advanced grant OCONTSOLAR Grant agreement ID: 789051.

References

[1] G. Battistelli, “Distributed moving-horizon estimation with arrival-cost consensus,” *IEEE Transactions on Automatic Control*, vol. 64, no. 8, pp. 3316–3323, 2018.

[2] M. Farina, G. Ferrari-Trecate, and R. Scattolini, “Distributed moving horizon estimation for linear constrained systems,” *IEEE Transactions on Automatic Control*, vol. 55, no. 11, pp. 2462–2475, 2010.

[3] M. Farina, G. Ferrari-Trecate, and R. Scattolini, “Distributed moving horizon estimation for nonlinear constrained systems,” *International Journal of Robust and Nonlinear Control*, vol. 22, no. 2, pp. 123–143, 2012.

[4] K. Gheitasi, M. Ghaderi, and W. Lucia, “A novel networked control scheme with safety guarantees for detection and mitigation of cyber-attacks,” in *18th European Control Conference*, 2019, pp. 1449–1454.

[5] A. Venturino, S. Bertrand, C. Stoica Maniu, T. Alamo, and E. F. Camacho, “A new ℓ-step neighbourhood distributed moving horizon estimator,” in *60th IEEE Conference on Decision and Control*, 2021, pp. 508–513.

[6] A. Haber and M. Verhaegen, “Moving horizon estimation for large-scale interconnected systems,” *IEEE Transactions on Automatic Control*, vol. 58, no. 11, pp. 2834–2847, 2013.

[7] P. Segovia, V. Puig, and E. Duviella, “Set-membership-based distributed moving horizon estimation of large-scale systems,” *ISA transactions*, 2021.

[8] A. Simonetto, D. Balzaretti, and T. Keviczky, “A distributed moving horizon estimator for mobile robot localization problems,” *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 8902–8907, 2011.

[9] Z. R. Yousefi and M. B. Menhaj, “Constrained distributed estimation based on consensus algorithm for mobile robots tracking,” in *IEEE International Conference on Control System, Computing and Engineering*, 2014, pp. 124–129.

[10] J. Zeng and J. Liu, “Distributed moving horizon estimation subject to communication delays and losses,” in *American Control Conference*, 2015, pp. 5533–5538.

[11] R. Postoyan and D. Nešić, “A framework for the observer design for networked control systems,” *IEEE Transactions on Automatic Control*, vol. 57, no. 5, pp. 1309–1314, 2011.

[12] F. Ferrante, F. Gouaisbaut, R. G. Sanfelice, and S. Tarbouriech, “State estimation of linear systems in the presence of sporadic measurements,” *Automatica*, vol. 73, pp. 101–109, 2016.

[13] K. R. Muske, J. B. Rawlings, and J. H. Lee, “Receding horizon recursive state estimation,” in *American Control Conference*, 1993, pp. 900–904.

[14] C. V. Rao, J. B. Rawlings, and J. H. Lee, “Constrained linear state estimation — a moving horizon approach,” *Automatica*, vol. 37, no. 10, pp. 1619–1628, 2001.

[15] A. Venturino, S. Bertrand, C. Stoica Maniu, T. Alamo, and E. F. Camacho, “Distributed moving horizon estimation with pre-estimating observer,” in *24th International Conference on System Theory, Control and Computing*, 2020, pp. 174–179.

[16] A. Venturino, C. Stoica Maniu, S. Bertrand, T. Alamo, and E. F. Camacho, “Distributed moving horizon state estimation for sensor networks with low computation capabilities,” *System Theory, Control and Computing Journal*, vol. 1, no. 1, pp. 81–87, 2021.

[17] D. Goldfarb and A. Idnani, “A numerically stable dual method for solving strictly convex quadratic programs,” *Mathematical programming*, vol. 27, no. 1, pp. 1–33, 1983.