Multi-vehicle localization by distributed MHE over a sensor network with sporadic measurements: further developments and experimental results

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

Multi-vehicle localization by distributed MHE over a sensor network with sporadic measurements: further developments and experimental results*

Antonello Venturino[a,b], Cristina Stoica Maniu[a], Sylvain Bertrand[b], Teodoro Alamo[c], Eduardo F. Camacho[c]

[a] *Université Paris-Saclay, CNRS, CentraleSupélec, Laboratoire des signaux et systèmes, 91190, Gif-sur-Yvette, France*
[b] *Université Paris-Saclay, ONERA, Traitement de l'information et systèmes, 91123, Palaiseau, France*
[c] *Department of Ingeniería de Sistemas y Automática, Universidad de Sevilla, Camino de los Descubrimientos, 41092 Sevilla, Spain*

Abstract

This paper proposes a Distributed Moving Horizon Estimation (DMHE) approach performed by an external static Sensor Network (SN) composed of surveillance cameras and their associated low-cost computers. This approach allows to localize a non-cooperative Multi-Vehicle System (i.e. intruder vehicles which do not communicate with the SN) from sporadic measurements. In this context, measurements are available at time instants *a priori* unknown and the proposed DHME technique is designed to face this issue by resorting to time-dependent parameters in the problem formulation. Moreover, this technique is well-suited to better estimate the state of the intruder vehicles thanks to its capability to efficiently exploit environmental information via constraints. In fact, when dealing with sporadic measurements and biased noisy sensors data, the use of output constraints can contribute to locally enhance the estimation accuracy. In order to confirm its effectiveness, the proposed method is validated on an experimental setup (video presentation available at `https://youtu.be/1CkSba2wVuI`) within an indoor arena equipped with a motion capture system. Three scenarios are considered for the localization of a non-cooperative Multi-Vehicle System composed of five robots, where the proposed DMHE technique is performed using sporadic position measurements provided by an external static Sensor Network with low-cost cameras (webcams) and computers (Raspberry PI) connected to them. Rigorous comparisons in terms of computation time and accuracy of the estimates highlight the efficacy of the proposed approach.

Keywords:
Distributed Moving Horizon Estimation, constrained state estimation, Sensor Networks, sporadic measurements, Multi-Vehicle Systems' localization

1. Introduction

During the past years, several research works have been dedicated to model, estimate and control distributed Multi-Agent Systems (MAS), see Negenborn and Maestre (2014), Vadigepalli and Doyle III (2003), Millán et al. (2013), Rego et al. (2019), etc. Considerable studies help to better cope with Multi-Agent Systems by developing suitable distributed algorithms to deal with state estimation of large-scale systems (Segovia et al. (2021)), with applications on Multi-Vehicle System (MVS) (Halsted et al. (2021)), also used to localize such systems. Two different main classes of problems can be considered, depending whether the localization is performed by the vehicles themselves and/or accounting for information from them (cooperative localization, e.g. Shorinwa et al. (2020), Viegas et al. (2018)), or by an exogenous system without any exchange of information with the vehicles (non-cooperative localization). This second class of problems can be related to area monitoring or surveillance applications, where the localization of intruders (e.g. non-cooperative vehicles) has to be estimated by a Sensor Network (SN) (Sharma and Chauhan (2020); Foresti and Snidaro (2002)), e.g. cameras. This is the class of problems and applications considered in the current paper.

When dealing with a network of several sensors, different estimation algorithms can be developed depending whether the information provided by each sensor is processed globally by a single entity (centralized estimation) or locally by computing nodes attached to the sensors (distributed estimation). In recent decades, the interest in distributed state estimation has increased tremendously. For instance, in He et al. (2020), the authors reviewed several outcomes of distributed state estimation over a low-cost Sensor Network (SN), pointing out their characteristics, benefits, and challenging issues. One critical point for such Sensor Networks comes from the limited computation and communication resources associated to local sensors, which strongly motivates the current paper.

Consensus techniques have often been used for designing distributed state estimation algorithms for Sensor Networks. A consensus problem in which the agreement value is a dis-

*The material in this paper was partially presented at the 2022 Conference on Control Technology and Applications, Trieste, Italy, August 23-25, 2022.

Email addresses:
`antonello.venturino@l2s.centralesupelec.fr`, `antonello.venturino@onera.fr` (Antonello Venturino), `cristina.stoica@l2s.centralesupelec.fr` (Cristina Stoica Maniu), `sylvain.bertrand@onera.fr` (Sylvain Bertrand), `talamo@us.es` (Teodoro Alamo), `eduardo@esi.us.es` (Eduardo F. Camacho)

tributed estimation of some non-constant quantity of interest is referred to as a dynamic consensus, see Manfredi (2013). Recently, such algorithms have pervaded the control engineering literature on numerous topics, e.g. state estimation (see Duan et al. (2022), Postoyan and Nešić (2011), Olfati-Saber (2007)), detection and mitigation of cyber-attacks (Gheitasi et al. (2019)), combined control and state estimation (Millán et al. (2013), Vadigepalli and Doyle III (2003)), etc.

Among all state estimation algorithms, Moving Horizon Estimator (MHE) techniques have been studied over the past decades due to their capacity to consider constraints within their formulation, which is based on a finite horizon "least-square" optimization problem (Muske et al. (1993)). These techniques use an estimation window of specified size, which moves forward at each instant. Therefore, the problem remains computationally tractable since only the latest measurements are processed, while previous information is condensed in the so-called *arrival cost* (Rao et al. (2001)). The constrained optimization problem brings with it a certain computational load, which may become problematic for large-scale systems (Haber and Verhaegen (2013), Vukov et al. (2015)), or when using MHEs techniques for computationally demanding purposes (see e.g. Famularo et al. (2022) where the authors proposed a MHE scheme using Linear Matrix Inequalities for fault detection and isolation). In addition to specific optimization methods, several techniques have been developed to reduce the computational load in Moving Horizon Estimation. Recent approaches on learning-based MHE rely on existing machine learning frameworks to computationally improve the estimator (Muntwiler et al. (2022), Karg and Lucia (2021)). Another method, *inter alia*, is to reduce the number of optimization parameters by replacing the sequence of unknown inputs (or state noise) to be estimated in the dynamical model of the system by a Luenberger *pre-estimation* observer, which leads to less computation time, while preserving the accuracy of the estimates, as introduced in Sui and Johansen (2014) for linear systems, in Suwantong et al. (2014) for nonlinear systems, and more recently in Venturino et al. (2020) for distributed state estimation of linear systems.

In the context of MAS and Sensor Networks, Distributed Moving Horizon Estimation (DMHE) has indeed received increased attention in recent years, starting from the estimation for linear systems by Farina et al. (2010), where the authors proved the convergence of the estimation error even under weak observability conditions. An extension of DMHE to nonlinear systems subject to constraints has been further proposed by the same authors in Farina et al. (2012). A Distributed MHE (DMHE) scheme for a class of nonlinear systems with bounded output measurement noise and process disturbances is designed in Zhang and Liu (2013), while J. Zeng and Liu (2015) considered Distributed Moving Horizon Estimation of nonlinear systems subject to communication delays and data losses. The authors of Yin and Liu (2017) developed a DMHE for a class of two-time-scale nonlinear systems described in the framework of singularly perturbed systems. In addition, Battistelli (2018) developed a DMHE with fused arrival cost suitable for a fully distributed implementation. An extension of this approach with pre-estimation (Venturino et al. (2020)) has been proposed in Venturino et al. (2021b) by also adding an observability rank-based feature to compute the consensus terms (while only using local available information), leading to better estimation accuracy. Additionally, the authors proposed in Venturino et al. (2021a) a Distributed Moving Horizon Estimation with pre-estimation and diffusion information mechanism, in which the information is spread out through the Sensor Network by exploiting the moving horizon paradigm. In the context of large-scale systems, Segovia et al. (2021) designed a two-step distributed state estimation scheme in the presence of unknown-but-bounded disturbances and noises, which involves a set-membership-based MHE. A Distributed Moving Horizon Estimation via operator splitting for automated robust power system state estimation has been proposed in Kim et al. (2021). More recently, in Yin and Huang (2022) an event-triggered DMHE is proposed for general linear systems that comprise several subsystems. A consensus variational Bayesian MHE for distributed Sensor Networks with unknown noise covariances has been proposed in Dong et al. (2022), where three consensus tasks are performed in parallel at each time instant.

Various distributed state estimation algorithms have been developed to address the localization problem of Multi-Vehicle Systems. In the more specific context of non-cooperative localization of MVS using Sensor Networks, distributed estimation algorithms relying on these consensus and DMHE paradigms have been developed, see e.g. Petitti et al. (2011), Simonetto et al. (2011) or Yousefi and Menhaj (2014). In Simonetto et al. (2011), 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 Yousefi and Menhaj (2014), the authors accounted for mobile nodes in the Sensor Network that led to 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 this context, distributed state estimation over not fully reliable Sensor Network could lead to *sporadic measurements*, i.e. available at time instants *a priori* unknown, as considered in Ferrante et al. (2016), Postoyan and Nešić (2011). In the case of sensors with limited field of views (e.g. cameras), sporadic measurements are prone to be even more frequent.

This paper addresses the problem of Distributed Moving Horizon Estimation for non-cooperative localization of a Multi-Vehicle System via a Sensor Network of static cameras which provides sporadic measurements. The contributions are the following.

First, since the proposed DMHE algorithm is based on the work of Venturino et al. (2020), it inherits the reduced computation time and the enhanced accuracy due to the pre-estimation observer. In addition, this new method is designed for realistic Multi-Vehicle Systems scenarios involving sporadic measurements. For this objective, constraints on measurements (given by the knowledge of the environment where the Multi-Vehicle System is evolving) are integrated using binary parameters in this novel Distributed Moving Horizon Estimation formulation. Therefore, the environmental information (such as the field of

view of the cameras or road constraints) is exploited to better estimate the system state, allowing to localize the MVS. Second, the current paper aims at evaluating the performance of the proposed DMHE approach (in terms of accuracy and computation time) on three real experiments[1] using different numbers of sensors, different communication network topologies and coverages of the cameras' fields of view. Indeed, one of the main contributions of this paper consists in the experimental validation of the proposed DMHE localization technique of a MVS. In the developed experiment setup, the static Sensor Network is composed of low-cost cameras which provide measurements on the positions of the vehicles. Each camera is attached to a Raspberry PI (RPI) for computational and network communication capabilities. The proposed DMHE algorithm has been implemented within the Robot Operating System (ROS) framework to run in a distributed way on each RPI. The non-cooperative Multi-Vehicle System is composed of five Turtlebot3 robots performing formation motion within a road-like area. This corresponds to a reduced-scale mockup of a surveillance scenario of an urban area where a sensor network would be used to localize intruders.

The area used for experiments is located in an indoor arena equipped with a motion capture system that will be used to evaluate the estimation accuracy of the proposed algorithm. To corroborate its efficiency, the proposed DMHE constrained formulation is compared with the notable DMHE algorithm Farina et al. (2010).

The paper is structured as follows. Section 2 describes the problem under investigation and introduces the main theoretical elements. The proposed DMHE algorithm is detailed in Section 3. Section 4 focuses on the hardware setup, the analysis and the comparison of the experimentation results obtained by using a Sensor Network with different communication topologies. Concluding remarks and further developments are drawn in Section 5.

Notation. The symbol $I_n \in \mathbb{R}^{n\times n}$ defines an n-by-n identity matrix. The transpose of a matrix M is denoted by $M^\top$. For a vector $x \in \mathbb{R}^n$, and a positive definite matrix $R \in \mathbb{R}^{n\times n}$, a weighted norm is denoted by $\|x\|_R = \sqrt{x^\top R x}$. Here, diag$(\cdot)$ denotes a block diagonal matrix. Given a set $\mathcal{S}$, then Co$(\mathcal{S})$ denotes its convex hull. Within a Multi-Vehicle System, the left superscript ν from $({}^\nu\cdot)$ refers to the ν-th vehicle. To soften the notation, when referring to the whole Multi-Vehicle System, the left superscript ν is omitted. In $(\cdot^i)$, the right superscript i refers to the i-th sensor. We distinguish *local* information, i.e. referring only to the sensor i, and *regional* information, i.e. referring to sensor i and its neighbors. A general bar notation $(\bar{\cdot})$ is used to denote the regional information.

2. Distributed State Estimation over static Sensor Network with Sporadic Measurements

This section describes the problem of Distributed State Estimation (DSE) of a non-cooperative Multi-Vehicle System over a Sensor Network with sporadic measurements. The considered models, and the characteristics of the Sensor Network are presented.

2.1. Problem description

Consider the problem of Distributed State Estimation of the state (in this paper the 2D position) of a non-cooperative Multi-Vehicle System by a Sensor Network. In this setting, we assume that the Sensor Network is composed of n_S different sensors performing sporadic measurements, i.e. the measurements are not obtainable at all times by each sensor. For example, a moving vehicle can be detected by a camera only when it is within its field of view. A (formation[2] of) vehicle(s) moving in unknown directions can thus be detected by a given sensor of the network at time instants *a priori* unknown.

The Multi Vehicle System under observation consists of n_V ground vehicles which are restricted to move in specific locations, e.g. on roads in urban environments (delimited by the yellow borders in Fig. 1). We additionally make use of this environment knowledge as position constraints in the Distributed State Estimation optimization problem.

2.2. Vehicle dynamical model for state estimation

Consider n_V vehicles. The ν-th vehicle dynamical model is represented as a discrete-time 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 vector 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.

In the application studied in this paper, single-integrator dynamics are considered for the vehicles. The state vector ${}^\nu x = [{}^\nu p_x \ {}^\nu p_y]^\top \in \mathbb{R}^2$ consists of planar position coordinates. The control input vector (i.e. the velocity components) is assumed unknown by the external sensor network for distributed state estimation (in the context of non-cooperative vehicles considered as intruders) and it is therefore considered as the unknown exogenous input ${}^\nu w_t \in \mathbb{R}^2$.

Remark 1. *Notice that, according to the adopted notation, the state of the* global *Multi-Vehicle System is denoted by* $x_t = [{}^1x_t^\top \ \dots \ {}^{n_V}x_t^\top]^\top$, *where its* global *dynamics is described by* $A = diag({}^1A, \dots, {}^{n_V}A)$. *The sensors of the SN have to estimate this state vector* x_t *in a distributed way. Notice that explicit estimation of the unknown inputs* ${}^\nu w_t$ *is not considered in this paper.*

2.3. Sensor measurement model

Given that each vehicle can be detected individually by each sensor i (right superscript), the following mathematical expression models the measurement provided by sensor i with respect to the ν-th vehicle

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

[1] A video presentation of the first experiments is available at: `https://youtu.be/1CkSba2wVuI`.

[2] This paper also deals with the case when only a part of a formation of vehicles can be within the field of view of a camera.

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

In the application considered in this paper, ${}^{\nu}y^i$ consists in a noisy measurement of the position coordinates ${}^{\nu}x$ of vehicle ν, i.e. ${}^{\nu}C^i = I_2$. More details are provided in Section 4.1 explaining how these measurements are obtained in practice in the experimental setup.

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. 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 $C^i = diag({}^1C^i, \ldots, {}^{n_V}C^i)$.*

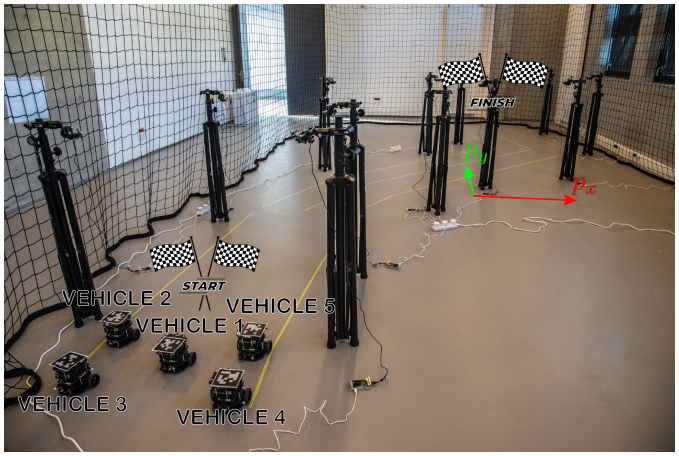


Figure 1: Experiment scenario setup: MVS with 5 vehicles in the starting place and Sensor Network composed of 12 cameras and Raspberry PI computers.

In this paper, we consider that all the measurements are sporadic. This way, we avoid abstruse notation to discern sporadic and non-sporadic measurements.

The following notation is necessary to denote the global system's collective output matrix which aggregates both the measuring and non-measuring situations of each sensor i

$$C_{\alpha_t}^i = D_{\alpha_t}^i C^i, \tag{3}$$

where $D_{\alpha_t}^i$ is a squared diagonal matrix of size $\sum_{\nu=1}^{n_V} {}^{\nu}n_y$ with ${}^{\nu}\alpha_t^i \in \{0, 1\}$ as components, for $\nu = 1, \ldots, n_V$, defined as

$$D_{\alpha_t}^i = \text{diag}({}^1\alpha_t^i I_{{}^1n_y}, \ldots, {}^{n_V}\alpha_t^i I_{{}^{n_V}n_y}), \tag{4}$$

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

Remark 3. *Notice that ${}^{\nu}\alpha_t^i$ is a time-dependent binary parameter marking if the sensor i can detect the ν-th vehicle at time t (i.e. ${}^{\nu}\alpha_t^i = 1$) or not (i.e. ${}^{\nu}\alpha_t^i = 0$). Hereafter, the sensor i is called* active *sensor at time t when ${}^{\nu}\alpha_t^i = 1$.*

2.4. Constraints

This subsection defines measurement constraints exploiting the *a priori* knowledge of the environment and the cameras composing the Sensor Network. First, denote by $\mathcal{R}$ the subset of planar coordinates corresponding to the road (assumed to be non-convex and marked by the blue lines in Fig. 2) on which the vehicles can drive. Further on, denote by $\mathcal{F}^i$ the set of the points forming the sensor i field of view (in yellow in Fig. 2). 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 in the state estimation process when the mobile vehicle is within the field of view of the sensor i, i.e. when this sensor detects the vehicle (see Fig. 2 for a graphic illustration).

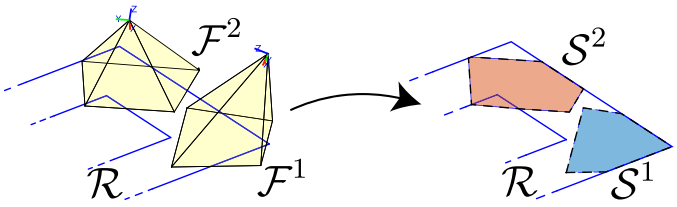


Figure 2: Road $\mathcal{R}$ (blue line), fields of view $\mathcal{F}^1$ and $\mathcal{F}^2$ (yellow), and convexified constraints $\mathcal{S}^1$ (red polygone) and $\mathcal{S}^2$ (blue polygone).

2.5. Sensor Network

In Distributed State Estimation schemes, the nearby sensors share data among each other. The Sensor Network is described, as in Farina et al. (2010), 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 sensor i, and *regional* information, i.e. referring to its entire neighborhood $\mathcal{N}^i$. Based on aforementioned notation, the regional output of sensor i at time t is

$$\bar{y}_t^i = [(y_t^i)^\top \ (y_t^{j_1})^\top \ (y_t^{j_2})^\top \ \ldots \ (y_t^{j_{n_S^i}})^\top]^\top, \{j_1, \ldots, j_{n_S^i}\} \in \mathcal{N}^i \tag{6}$$

with $\bar{y}_t^i \in \mathbb{R}^{\bar{n}_y^i}$.

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

$$k_{ij} > 0 \ \text{ if } (j, i) \in \mathcal{E}, \tag{7a}$$

$$k_{ij} = 0 \ \text{ otherwise}, \tag{7b}$$

$$\sum_{j=1}^{n_S} k_{ij} = 1, \ \ \forall i = 1, \ldots, n_S. \tag{7c}$$

The k_{ij} values can be selected according to some criteria. For instance, in Venturino et al. (2021b) the authors proposed an observability rank-based method that leads to a better accuracy for the estimations since the K matrix will be used in the DMHE algorithm to compute some consensus terms, as described further on in Section 3.

2.6. 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.

3. 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 Venturino et al. (2021b) and presents its novel formulation to handle the Multi-Vehicle localization application considered in the current paper. This novel formulation extends and validates the preliminary results presented in Venturino et al. (2022).

Moreover, thanks to the observability rank-based weights, the accuracy of the estimates is improved and also it enables the use of classic DMHEs for sporadic measurements.

In the DMHE approach, each sensor has to compute an estimate by online solving an optimization problem involving the use of a model of the system dynamics, a cost function and constraints. Each of these elements are presented hereafter and the optimization problem is further formulated in Section 3.6.

3.1. Dynamic model with pre-estimation

The optimization problem to be solved in DMHE to compute a state estimate involves the use of a model of the system dynamics. Regarding system (1), a straightforward choice would be to consider for sensor i the following model to represent the global MVS dynamics

$$\hat{x}_{t+1}^i = A\,\hat{x}_t^i + \hat{w}_t^i. \tag{8}$$

This choice is mainly used in classical MHE formulations which have consequently to estimate the sequence of $\hat{w}$ over the measurement horizon, since they are unknown. However, this leads to a possibly large number of optimization parameters depending on the horizon length N and the dimension of w.

In the current paper, following Sui and Johansen (2014) and Venturino et al. (2020), a model of the system dynamics without w is chosen to drastically reduce the number of optimization parameters and computation time. Not estimating and neglecting w would possibly lead to large model errors and would result in poor performance of the estimator. Therefore, model mismatch is accounted for through feedback information provided by measurements, under the form of a correction term, leading to the following "pre-estimation" Luenberger observer-like model:

$$\hat{x}_{t+1}^i = A\,\hat{x}_t^i + L_{\alpha_t}^i\left(y_t^i - C^i\,\hat{x}_t^i\right). \tag{9}$$

The dependence with ${}^{\nu}\alpha_t^i$ is formulated via $L_{\alpha_t}^i = L^i D_{\alpha_t}^i$, with $D_{\alpha_t}^i$ defined by (4). Moreover, the global Luenberger gain L^i is computed such that $\Phi^i = A - L^i C^i$ is Schur stable when the Multi-Vehicle System is detectable by sensor i, i.e. the pair (A, C^i) is detectable. As *extrema ratio*, one could compute the L^i gain by keeping the spectral radius of Φ^i as small as possible. One may compute the gain related to the global MVS or separately, since $L^i = [{}^1L^i \ \dots \ {}^{n_V}L^i]$.

3.2. Fields of view constraints

The binary parameter ${}^{\nu}\alpha_t^i$ allows to deal with the sporadic measurements. Indeed, it is effective to discern when the constraints $\mathcal{S}^i$ can be used by sensor i and when not, i.e., respectively, when the sensor i can detect a specific vehicle and when not. In particular, considering the vehicle ν detected or not by sensor i at time t, the following constraints ${}^{\nu}\mathcal{S}_{\alpha_t}^i$ are defined

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

3.3. Objective function

The binary parameter ${}^{\nu}\alpha_t^i$ plays a crucial role also in the objective function $J_{\alpha_t}^i$, which is defined as

$$J_{\alpha_t}^i(\cdot) = \frac{1}{2}\sum_{k=t-N}^{t}\left\|\bar{y}_k^i - \bar{C}_{\alpha_t}^i\,\hat{x}_k^i\right\|^2_{(\bar{R}^i)^{-1}} + \Gamma_{t-N}^i(\cdot), \tag{10}$$

where $\bar{R}^i$ is the regional covariance matrix of the measurement noise. Here, we assume that $\bar{R}^i$ is a positive definite matrix. The term $\bar{R}^i$ weights the difference between the predicted outputs and the measurements within the fixed window of size N. In the MHE paradigm the last term $\Gamma_{t-N}^i(\cdot)$ of (10) is known as the *arrival cost*. This non negative term summarizes the effect of the past measurements, before time $t - N$ and is usually approximated by some *initial penalty* function defined as follows:

$$\Gamma_{t-N}^i(\cdot) = \frac{1}{2}\left\|\hat{x}_{t-N}^i - \hat{\bar{x}}_{t-N|t-1}^i\right\|^2_{(\bar{\Pi}_{t-N|t-1}^i)^{-1}}, \tag{11}$$

which involves two consensus terms, $\hat{\bar{x}}_{t-N|t-1}^i$ and $\bar{\Pi}_{t-N|t-1}^i$, described below. These consensus terms help the estimates computed by the neighbor sensors to converge to a common value and, thus, to reach consensus over the sensor network. They also help to cope with local non observability issues.

3.4. Consensus terms

The first term included in the penalty function Γ_{t-N}^i is the *consensus-on-estimates* term, denoted by $\hat{\bar{x}}_{t-N|t-1}^i$. It consists in a weighted average state estimate computed over the neighborhood $\mathcal{N}^i$ as follows:

$$\hat{\bar{x}}_{t-N|t-1}^i = \sum_{j\in\mathcal{N}^i} k_{ij|t}\,\hat{x}_{t-N|t-1}^j, \tag{12}$$

where $\hat{x}_{t-N|t-1}^j$ is the estimated state computed at time $t-1$ by sensor $j \in \mathcal{N}^i$. It is a consensus term in the sense that it penalizes deviations of $\hat{x}_{t-N}^i$ from $\hat{\bar{x}}_{t-N|t-1}^i$.

The second term is the positive definite matrix $\bar{\Pi}^i_{t-N|t-1}$ which is computed similarly as in Farina et al. (2010). For exhaustiveness, we describe here the entire procedure to compute it by:

$$\bar{\Pi}^i_{t-N|t-1} = \sum_{j\in\mathcal{N}^i} k_{ij|t}\Pi^j_{t-N|t-1}, \tag{13}$$

where the iteration of $\Pi^i_{t-N|t-1}$ is calculated by the sensor i based on regionally available data. Specifically, the matrix $\Pi^i_{t-N|t-1}$, with $i \in \mathcal{N}$, is the result of one iteration of the difference Riccati equation associated to a Kalman filter for the following system

$$\begin{cases} x_{t-N} = Ax_{t-N-1} + w_{t-N-1} \\ \bar{z}^i_{t-N} = \bar{O}^i_N x_{t-N} + \bar{V}^i_{t-N}, \end{cases} \tag{14}$$

where $\bar{z}^i_{t-N} \in \mathbb{R}^{N\cdot n^i_y}$is the output and where $\bar{V}^i_{t-N}$ denotes the measurement noise and $\bar{O}^i_N$ represents the i-th sensor regional observability matrix

$$\bar{O}^i_N = \begin{bmatrix}(\bar{C}^i)^\top & (\bar{C}^iA)^\top & \cdots & (\bar{C}^iA^{N-1})^\top\end{bmatrix}^\top. \tag{15}$$

Then, if we define the following matrices

$$\mathcal{S}^i_N = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ \bar{C}^i & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \bar{C}^iA^{N-2} & \bar{C}^iA^{N-3} & \cdots & \bar{C}^i \end{bmatrix} \in \mathbb{R}^{\bar{n}^i_y N\times n_x(N-1)}, \tag{16}$$

$$\bar{R}^i_N = \mathrm{diag}(\bar{R}^i,\dots,\bar{R}^i) \in \mathbb{R}^{\bar{n}^i_y N\times \bar{n}^i_y N}, \tag{17}$$

$$Q_{N-1} = \mathrm{diag}(Q,\dots,Q) \in \mathbb{R}^{n_x(N-1)\times n_x(N-1)}, \tag{18}$$

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

and set the covariance of the estimate $\hat{x}^i_{t-N-1}$ as

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

the resultant Riccati recursive equation is given by

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

If we assume that the communication network topology is time-invariant then these equations can be computed off-line. Once the matrices $\Pi^i_{t-N|t-1}$ have been computed and shared among the neighbors, sensor i can then perform a *consensus weights' update* in order to compute the matrix $\bar{\Pi}^i_{t-N|t-1}$ according to (13).

These two consensus terms help to improve the accuracy of the local estimates and they are necessary to guarantee convergence of the state estimates to the state of the observed system even if it lacks of regional observability (Farina et al. (2010)).

3.5. Observability rank-based weights technique

Here, we adapt the weights' tuning technique proposed in Venturino et al. (2021b) for the stochastic matrix K associated with the graph $\mathcal{G}$ to the considered Multi-Vehicle localization problem by DMHE over a Sensor Network with sporadic measurements.

Thanks to this method, each sensor i computes its components of K based on only *locally* available data. Hence, it is appropriate for a distributed scheme, and furthermore, for the application considered in this paper with sporadic measurements. Indeed, this technique enables to improve the accuracy of the estimates by relying more on the sensors that are currently sensing, in other words, by exploiting the observability properties of the neighborhoods. Since these properties are time-varying for the considered problem, the observability rank-based weights technique is suitable for enhancing the algorithm's accuracy and convergence time.

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

$$\bar{O}^i_{n|t} = \begin{bmatrix}(\bar{C}^i_{\alpha_{t-n+1}})^\top & (\bar{C}^i_{\alpha_{t-n+2}}A)^\top & \cdots & (\bar{C}^i_{\alpha_t}A^{n-1})^\top\end{bmatrix}^\top \tag{22}$$

is of full rank if and only if the the pair $(A, \bar{C}^i_{\alpha_t})$ is completely observable at any instant of time within the interval $[t-n+1,\dots,t]$, i.e. $\mathrm{rank}(\bar{O}^i_{n|t}) = n_x$. For simplicity, we denote by $\rho^i_{O|t} = \mathrm{rank}(\bar{O}^i_{n|t})$. This variable will be considered as an information on the *reliability* of node i, i.e. its sensing capability, and will be used to define the weighting coefficients k_{ij}, which must satisfy constraint (7). Note that, at some time instants *a priori* unknown, the entire neighborhood may not have sensing capabilities at all, i.e. $\rho^i_{O|t} = 0$. To avoid division by zero a lower bound smaller than 1 is chosen for the rank, here 0.5 is chosen, i.e. $\rho^i_{O|t} = \max\left\{\mathrm{rank}(\bar{O}^i_{n|t}), 0.5\right\}$. Then the components $k_{ij|t}$ are computed as follows

$$k_{ij|t} = \frac{\rho^j_{O|t}}{\sum_{j\in\mathcal{N}^i}\rho^j_{O|t}}. \tag{23}$$

3.6. Local optimization problem

Given an estimation horizon length $N \geqslant 1$, at each time t, each sensor $i \in \mathcal{N}$ determines the state estimate $\hat{x}^i_{t|t}$ by solving the following constrained minimization problem:

$$\hat{x}^i_{t-N|t} = \underset{\hat{x}^i_{t-N}}{\arg\min} \quad J^i_{\alpha_t}(\cdot) \tag{24}$$

$$\text{s.t.} \quad \hat{x}^i_{k+1} = A\,\hat{x}^i_k + L^i_{\alpha_k}\left(y^i_k - C^i\,\hat{x}^i_k\right), \tag{25}$$

$$\hat{x}^i_k \in \mathcal{X}, \tag{26}$$

$$\bar{C}^i\,\hat{x}^i_k \in \mathcal{S}^i_{\alpha_k}, \tag{27}$$

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

where the sets $\mathcal{S}^i$ are computed as detailed in Section 2.4. The sequence of state estimates $\hat{x}^i_{t-N+1|t},\dots,\hat{x}^i_{t|t}$ is obtained from the optimal solution $\hat{x}^i_{t-N|t}$ and using the dynamics (25). The A matrix in (25) refers to the global Multi-Vehicle System. Moreover, the positions constraints, as described in Section 2.4, are integrated in the optimization problem in (26).

3.7. 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 neighbor nodes $j \in \mathcal{N}^i$: L^j, C^j, R^j
3: **compute** the pre-estimation Luenberger gain L^i
4: **store** the *a priori* initial estimation $\hat{x}^i_{0|0} = \hat{x}_0$ 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^i_0
7: **receive** from neighbors $j \in \mathcal{N}^i$ their measurements y^j_0
8: **Online:** $\forall i \in \mathcal{N}$, $\forall t > 0$
9: **collect** the local measurement y^i_t
10: **receive** from the neighbors $j \in \mathcal{N}^i$ the collected data in step 9
11: **compute** the matrix $D^i_{\alpha_t}$ according to (4)
12: **compute** the components $k_{ij|t}$ according to (23)
13: **if** $1 \leqslant t \leqslant N$ **then**
14: **set** the horizon length $N = t$, the covariance matrix $\bar{\Pi}^i_{t-N|t-1} = \bar{\Pi}^i_{0|t-1} = \Pi_0$ and the *a priori* initial estimation state $\hat{x}^i_{t-N|t-1} = \hat{x}^i_{0|t-1}$
15: **else**
16: **compute** $\Pi^i_{t-N|t-1}$ according to (19), (20) and (21)
17: **receive** $\Pi^j_{t-N|t-1}$ from the neighbor nodes $j \in \mathcal{N}^i$
18: **compute** $\bar{\Pi}^i_{t-N|t-1}$ according to (13)
19: **solve** the local optimization problem of DMHE, minimizing J^i as in (10) and (11) subject to the constraints (25)-(26)
20: **store** the solution $\hat{x}^i_{t-N|t}$ and the corresponding estimate $\hat{x}^i_{t|t}$
21: **receive** from the neighbors $j \in \mathcal{N}^i$ their estimates $\hat{x}^j_{t-N+1|t}$

The specific steps related to sporadic measurements are integrated at step 19, with $D^i_{\alpha_t}$ 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.

4. Experimental results

This section describes three conducted experiments within the indoor arena equipped with an OptiTrack motion capture system used to provide ground truth localization. This information is compared with the position estimates performed by the DMHE algorithms in order to evaluate the estimation accuracy. The estimation algorithms are using the measurements provided by low-cost cameras (webcams). In the context of intruders' localization, the leading goal is to localize a formation of several mobile robots moving on a road, by performing the proposed Distributed Moving Horizon Estimation over a Sensor Network of low-cost cameras within a given communication topology, performing sporadic measurements. These vehicles are moving in a formation, along the road, controlled by a distributed algorithm using localization from the motion capture system. They are considered as non-cooperative vehicles for the localization problem performed by the Sensor Network.

The experiments are developed within the Robot Operating System (ROS) framework. The distributed state estimation algorithm is deployed on the Raspberry PI (RPI), each one being attached to a low-cost camera (webcam), and in charge of obtaining measurements from the camera, of exchanging information among neighbors and of locally estimating the state of the MVS. The AprilTag library (Wang and Olson (2016)) is used to get position measurements of the vehicles, from each camera. This library uses tags (known in size and pattern) placed on top of the vehicles to robustly and efficiently detect the vehicles and reconstruct their position.

The experimentally collected data are further off-line reprocessed and analyzed by adding artificial Gaussian noise to the measurements and changing the topology of the Sensor Network. We compare online and off-line results w.r.t. the DMHE algorithm of Farina et al. (2010).

4.1. Experiments setup

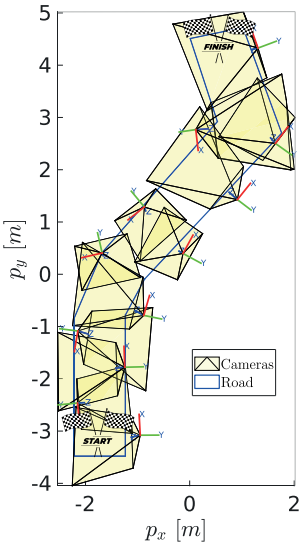
The objective is to track a Multi-Vehicle System (MVS) composed of $n_V = 5$ ground vehicles, one leader in the center of a square and four follower vehicles in the vertices of the square. The MVS goes from the starting point $(-1.75, -3.25)$ m towards the final point $(0.75, 4)$ m driving within the road, clearly indicated in Figs. 3a-5a, controlled by a leader-follower formation distributed control strategy. The control inputs of the intruders' vehicles are assumed to be unknown. The details of this control strategy are beyond the scope of this paper and they are omitted here.

In order to analyze the performance of the proposed Distributed Moving Horizon Estimation, we designed three experimental scenarios, with a different number of sensors involved in the distributed state estimation and different poses for the cameras:

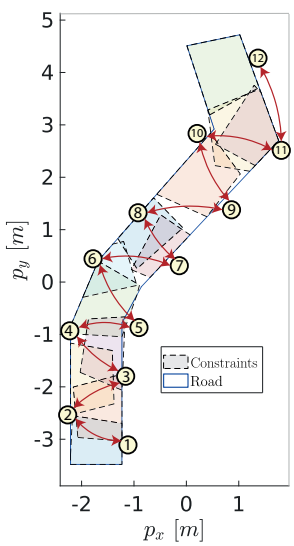
- **Scenario 1** (see Fig. 3) uses 12 cameras for a maximum coverage area by their fields of view (video presentation available at `https://youtu.be/1CkSba2wVuI`);
- **Scenario 2** (see Fig. 4) uses 6 cameras for a maximum coverage area by their fields of view;
- **Scenario 3** (see Fig. 5) uses 12 cameras for a reduced coverage area by their fields of view w.r.t. Scenario 1.

In Fig. 3-Fig. 5, the yellow polyhedra represent the *fields of view of the cameras*[3] (together with their reference frames), the

[3]The fields of view $\mathcal{F}^i$ of the low cost cameras are determined from the technical specifications provided by the manufacturer and from the pose information of the cameras provided by the OptiTrack motion capture system.

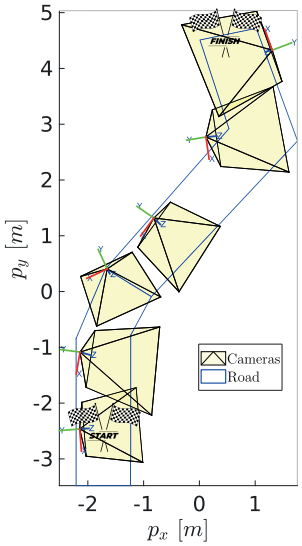


(a) FoVs real experiment

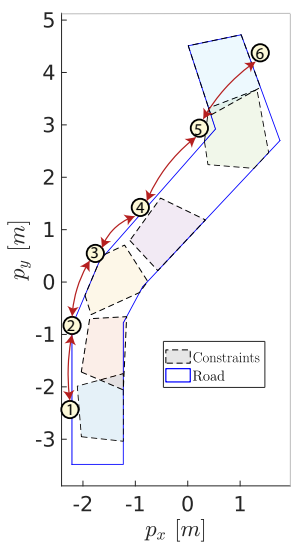


(b) Constraints real experiment

Figure 3: Scenario 1, maximum coverage area with 12 sensors.



(a) FoVs real experiment



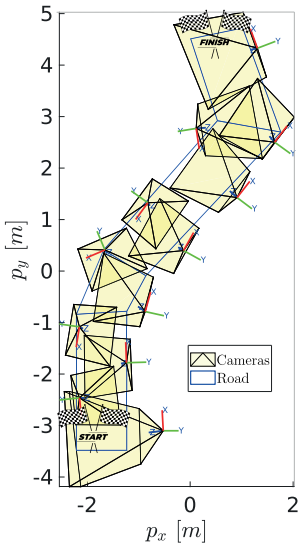
(b) Constraints real experiment

Figure 4: Scenario 2, maximum coverage with area 6 sensors.

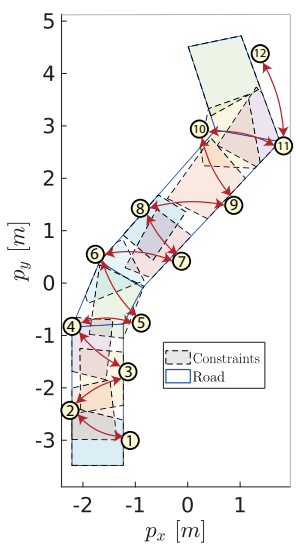
solid blue lines define the *road boundaries*,[4]the dashed colored polygons are the constraints, and finally, the red arrows represent the communication links between the computing nodes (RPIs) associated to the cameras. Notice that the poses of the 12 cameras used in Scenario 3 (except for the 12th sensor) are defined such that each camera can detect a maximum of 3 vehicles, i.e. their fields of view points only at half of the road (see Fig. 5a).

As mentioned in Section 2.2, in the DMHE optimization problem, each vehicle is modeled as single integrator, where the state vector ${}^{\nu}x = [{}^{\nu}p_x \ {}^{\nu}p_y]^\top \in \mathbb{R}^2$ consists of planar position coordinates and the control input vector (i.e. the velocity components) is assumed unknown and considered as an exogenous input ${}^{\nu}w_t \in \mathbb{R}^2$, modeled as an uniformly distributed noise vector with covariance matrix $Q = I_2$.

Each camera provides position measurements of the vehicles in its own reference frame (indicated in Fig. 3a-Fig. 5a). Thus, in order to obtain position measurements in the common absolute reference frame used for the experiments and associated to the motion capture system, it is necessary to translate and rotate the measurements with a transformation matrix. To calculate such a matrix, the knowledge of the poses of the cameras, in a common reference frame, are necessary. However, these poses are not always available, or at least not precisely known, as it is the case in these experiments. Indeed, here we obtained the poses of the low-cost cameras using the OptiTrack motion capture system, which detects 3-4 markers glued on each camera. The precision of these detections was falling on some areas of the arena, e.g. less observed areas such as corners. Such an error in the transform matrix therefore results in some bias in the robots' position measurement translated in the global frame and provided to the estimators. Despite this, the robustness of the proposed DMHE to this additional source of uncertainty (i.e. sensor biases and noise) is further investigated by validating the usefulness of *a priori* known road constraints.



(a) FoVs real experiment



(b) Constraints real experiment

Figure 5: Scenario 3, less coverage area with 12 sensors.

The estimators run with a sampling time $T_s = 0.5$ s and a horizon length $N = 3$. 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 measurements noises ${}^{\nu}v_t^i$ are assumed to be white normally distributed noises, with zero mean and covariance matrix $R^i = I_2$.

The optimization problem was solved by the quadratic programming solver from Goldfarb and Idnani (1983) implemented in Python on twelve RPI, each one associated to a single camera. The considered performance indexes are the com-

[4]The road boundaries are determined by using the OptiTrack motion capture system while preparing the experiments.

putation 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

$$\mathrm{RMSE}_t = \frac{1}{n_S} \sum_{i \in \mathcal{N}} \left\| x_t - \hat{x}^i_{t|t} \right\|,$$

both averaged among all the sensors. The RMSE should remain small for good performance.

We compare the proposed Distributed Moving Horizon Estimation without constraints (denoted by DMHE) and with constraints[5] (denoted by DMHE$^{\mathcal{S}}$). We also provide comparison with the algorithm of Farina et al. (2010), denoted hereafter by DMHE$_F$, for the unconstrained case, and by DMHE$_F^{\mathcal{S}}$ when considering the constraints (5).

n_v	n_S	${}^v n_x^i$	${}^v n_y^i$	T_s	N
5	12 or 6	2	2	0.5 s	3

Table 1: Setting parameters

Table 1 lists the parameters of the developed experiments. Notice that n_S is 12 for scenarios 1 and 3, while n_S is 6 for scenario 2.

4.2. Experimental results

It is essential to highlight that between experiments some parameters are not exactly repeatable (e.g. unpredictable lack of measurements at time instants *a priori* unknown, different initial timing synchronization among sensor neighborhoods, initial positions of the vehicles, etc.) and that a qualitative evaluations may therefore suffer from some bias in the comparison between two experimental runs. This is why all the measurement data have also been recorded to additionally perform offline evaluation from the same data. The video available at `https://youtu.be/1CkSba2wVuI` shows the online experiment of Scenario 1 on using DMHE$^{\mathcal{S}}$ and offers additional details.

Figure 6 illustrates the averaged computation time τ among all the sensors for the three scenarios. The proposed DMHE$^{\mathcal{S}}$ (dotted lines) shows half of the time needed by DMHE$_F^{\mathcal{S}}$ (dashed lines) in all the scenarios. It is a consequence of replacing the system model with unknown input by a Luenberger observer (pre-estimation strategy) in (25). Indeed, the input sequence with a Luenberger pre-estimation involves fewer optimization parameters.

The RMSEs averaged among all the sensors for all scenarios and algorithms with constraints are shown in Fig. 7. Regarding Scenario 1, DMHE$^{\mathcal{S}}$ is better than DMHE$_F^{\mathcal{S}}$ only until *cerca* $t = 70$ s. For Scenarios 2 and 3, DMHE$^{\mathcal{S}}$ has similar or better performance w.r.t. DMHE$_F^{\mathcal{S}}$. As explained before, a rigorous comparison is hard due to unrepeatable conditions, that is why we performed more rigorous comparisons in the next section.

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

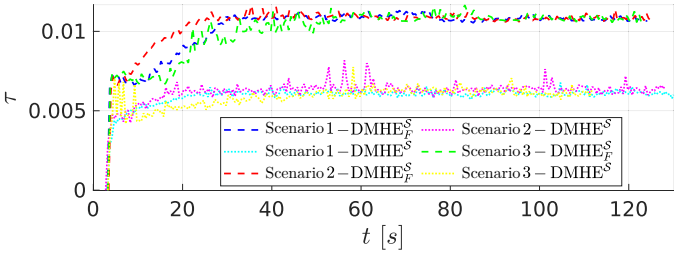


Figure 6: Computation time τ of all algorithms with constraints during the real experiments.

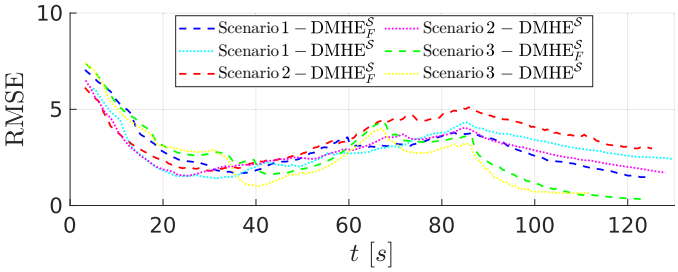
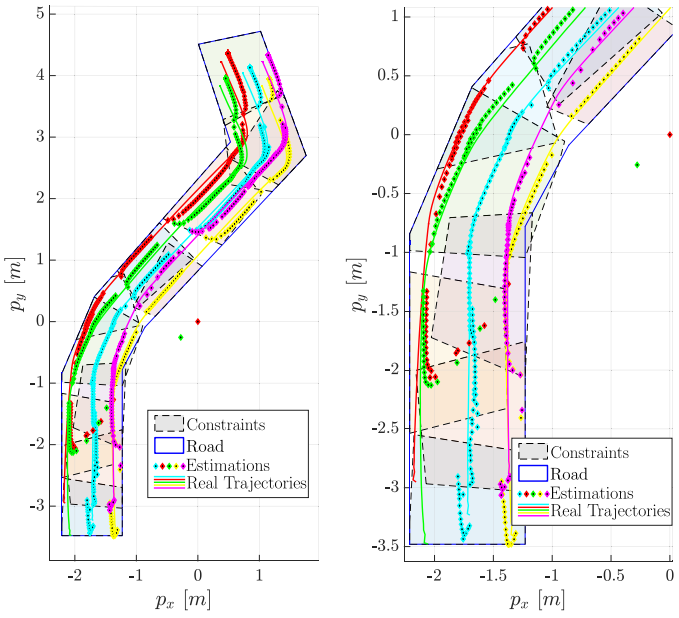


Figure 7: RMSE of all algorithms with constraints during the real experiments.

Figure 8 illustrates the results of the real experiment of Scenario 1. In particular, it shows the trajectories estimation (rhombus) by the active sensors operating with DMHE$^{\mathcal{S}}$ and the actual trajectories of the vehicles (lines) by the motion capture system of Scenario 1 on the left hand side and its zoom on the right hand side. Different colors highlight the vehicles: cyan, red, green, yellow, and magenta refer to vehicles 1, 2, 3, 4, and 5, respectively. Moreover, Fig. 8 shows the road boundaries (blue lines) as well as the constraints (dashed polygones) described as in (5). Notice that starting position is about $p_y = -3.5$ m. As shown in Fig. 8b, at the beginning only vehicles 1 (cyan rhombus), 4 (yellow rhombus), and 5 (magenta rhombus) are detected by the first camera (i.e. they are within the blue dashed polygone, at the bottom). Indeed, camera 1 does not detect vehicles 2 and 3 (i.e. no red or green rhombuses appearing inside this blue dashed polygone). Vehicles 2 and 3 start being detected later on. The rhombus outside the road are due to the initial state estimates considered by the estimators, which are chosen to be at the origin of the plan. Notice that the biased sensors data can result in some bias in the estimations (see the difference between the real trajectories and estimates around the arrival position in the light green polygone on the top in Fig. 8a); however, the magnitude of measurement noise is not perceptible. This is because the AprilTag library (Wang and Olson (2016)) provides accurate position of the tags mounted on the robots, by visual reconstruction using calibration information of the cameras and tags with known sizes and patterns. This is what also motivates us to introduce measurement noise on the experimental data to further validate and analyze the performance of the algorithms.

4.3. Performance evaluation

The ROS framework offers the opportunity to record data for the three scenarios explained above. This data includes time



(a) Entire scenario (b) Zoom

Figure 8: Scenario 1: Planar trajectory of the vehicles and estimations by DMHE^S using 12 cameras (real experiment).

synchronization among measurements and other useful information. Thus, it allows to replay these data in order to replicate the same experiments but with other estimation algorithms and/or changing their parameters. We also added *a posteriori* artificial Gaussian noise in the measurements, with a variance of 0.15 m^2, and in the positions of the cameras (bias), with a variance of 0.1 m^2 to make the scenarios more realistic, e.g. when using a low-cost Sensor Network with video cameras that would run computer vision algorithms for visual detection and position reconstruction of vehicles without tags (e.g. area monitoring scenario).

The first aspect we investigate for Scenario 1 is the topology of the Sensor Network, i.e. how the sensors are connected to each other. For this reason, we define the *radius communication link* as the number of edges reachable in communication by each sensor. Thus, increasing or decreasing this radius can change the topology of the network, i.e. the edges of the graph. To this purpose, let define $d(i, j)$ as the distance, in terms of the number of edges, between nodes i and j. Formally, having a radius ρ, the neighborhood of sensor i is $\mathcal{N}^i = \{j \in \mathcal{N} : d(i, j) \leqslant \rho\}$, i.e. the set of nodes $j \in \mathcal{N}$ for which there exists a path at maximum distant ρ edges from sensor i. For example, in Figs. 3b, 4b and 5b, the radius communication link is $\rho = 1$.

In the next sections, the algorithm performance on the three considered scenarios will be further analyzed by considering noisy measurements and influence of the communication topology.

4.3.1. Scenario 1

The aim of this scenario (see Fig. 3), is to compare the accuracy of the algorithms, with and without constraints, changing the topology of the Sensor Network by varying the radius communication link $\rho = \{1, 3, 6, 9, 12\}$. Notice that when the radius is 12 the graph is complete, i.e. each node is connected to anyone else. Moreover, we show the effects on using or not the constraints as in (5).

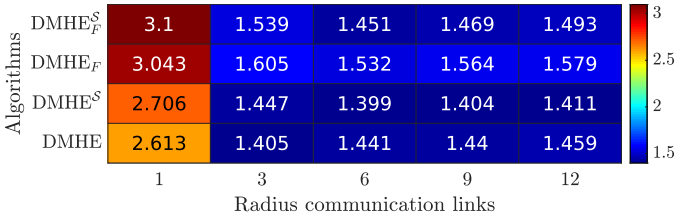


Figure 9: Scenario 1: RMSE averaged among the estimators and time (all sensors).

The first column of Fig. 9 shows the RMSE for the four implemented algorithms with $\rho = 1$. In this case, the smallest value 2.613 (and thus the best accuracy) is obtained with the proposed DHME algorithm without constraints. Figure 9 also shows that starting from $\rho = 3$, the RMSEs are similar to each other. It means that, for this number of sensors, a graph with a radius communication link equal to 3 or higher performs as good as a complete graph. Moreover, the accuracy is always better for the proposed DMHE (lines 3 and 4, respectively) compared with DMHE$_F$ (lines 1 and 2, respectively), for both constrained and unconstrained cases.

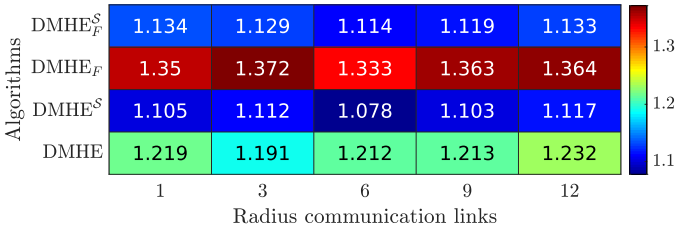


Figure 10: Scenario 1: RMSE averaged among the estimators and time (only active sensors).

To check how the constraints $\mathcal{S}^i$ influence the accuracy of the estimations, we have to look at the RMSE of the active sensors only since the constraints are used in the local optimization problem only when the camera is detecting a vehicle. In Fig. 10, the RMSE of constrained algorithms is always better than their respective unconstrained version. Indeed, it can be seen that the values on line 1 are always lower than the values on line 2, while the values on line 3 are always lower than the values on line 4.

We have seen that the radius ρ has a specific effect on the estimation error. In Fig. 11, we can see the RMSE overtime for the solely DMHE^S, for different values of the radius ρ. It is worth noticing that $\rho > 3$ (yellow, purple and green curves) leads to having better RMSE than $\rho = 3$ (red curve) until the vehicles stop, around $t = 88$ s.

Figure 12 and its zoom (Fig. 13) illustrate the planar trajectories of the vehicles (solid lines) by the motion capture system and their respective estimations (rhombus) by the active sensors. Different colors highlight the vehicles: cyan, red, green,

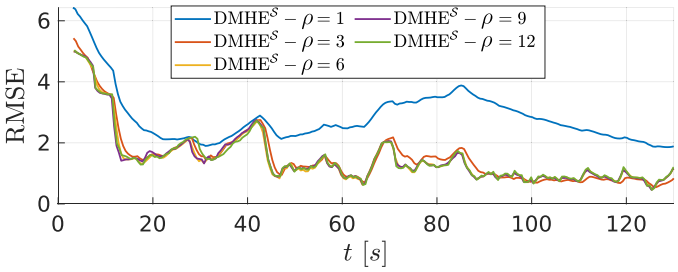
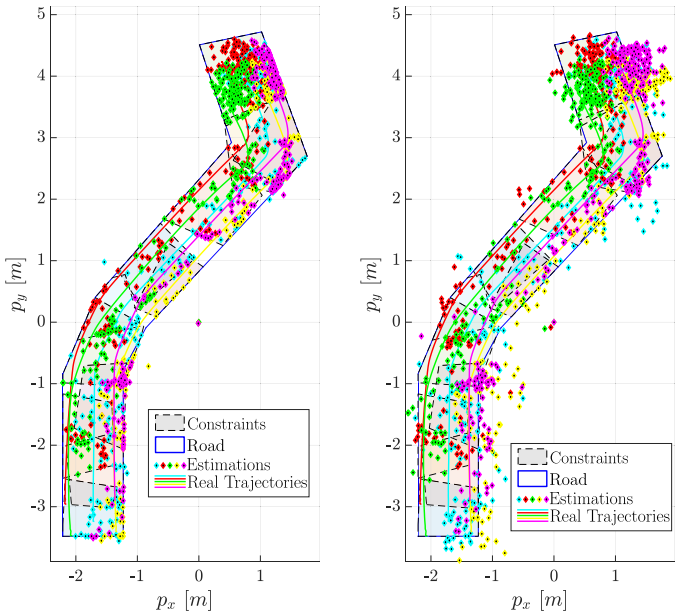


Figure 11: Scenario 1: RMSE of DMHE$^{\mathcal{S}}$ over time for all communication links radius.

yellow, and magenta refer to vehicles 1, 2, 3, 4, and 5, respectively. Moreover, this figure shows the road boundaries (blue line) as well as constraints (dashed polygones), as in (5). In these figures, it is possible to see how the constraints improve the estimation accuracy since the estimates are lying within the constraints and are hence consistent with locations of the vehicles within the road boundaries. This is more evident in zoom proposed in Fig. 13 which shows the final part of the Scenario 1. In particular, Fig. 13a shows the results by using the proposed DMHE with constraints (DMHE$^{\mathcal{S}}$) and Fig. 13b shows the results by using the proposed DMHE without constraints. Notice that in Fig. 13a only two points are outside the constraints (due to mismatch among the considered detection instants by low-cost cameras, Raspberry Pi, ROS), while Fig. 13b shows a lot of estimates outside the constraints.

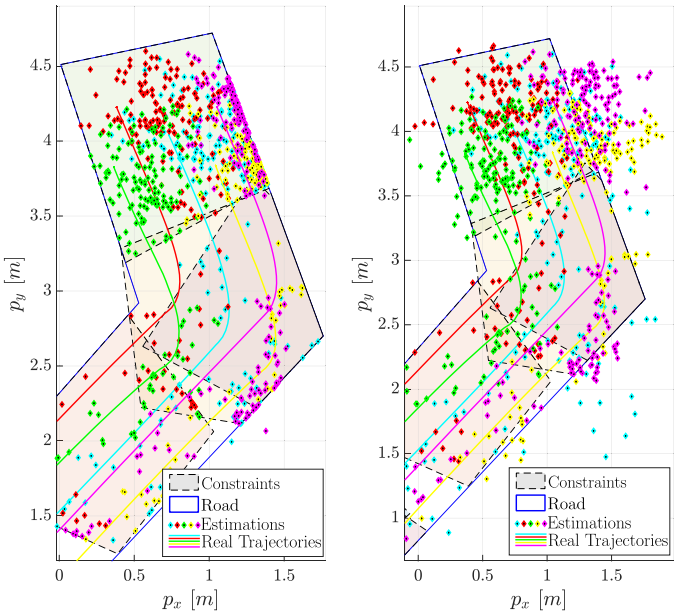


(a) DMHE with constraints (b) DMHE without constraints

Figure 12: Scenario 1: Planar trajectory of the vehicles and estimations with constraints (left) and without (right).

4.3.2. *Scenario 2*

This scenario (see Fig. 4) aims to evaluate the algorithms in terms of accuracy while using fewer sensors and, at the same



(a) DMHE with constraints (DMHE$^{\mathcal{S}}$) (b) DMHE without constraints

Figure 13: Scenario 1: Part of the planar trajectory of the vehicles and estimations with constraints (left) and without (right).

time, to diversify the communication topology. Additionally, it highlights the effects on the convergence of the estimates when using a different number of sensors in such distributed algorithms compared to Scenario 1. In this case, we used half of the sensors, which led to more sporadic measurements since the total covered area by cameras is much smaller (see Fig. 4) compared to Scenario 1.

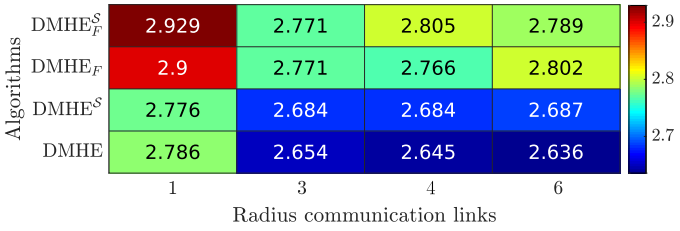


Figure 14: Scenario 2: RMSE averaged among the estimators and time (all sensors).

Figure 14 shows that for different values of the radius communication links, i.e. $\rho = \{1, 3, 4, 6\}$, the RMSEs are not so different from each other, and probably a radius $\rho = 2$ would have been the optimum trade-off between the number of communication links and the accuracy of the estimates. Moreover, comparing the RMSE values in Fig. 9 (Scenario 1) and in Fig. 14 (Scenario 2) it is evident, as expected, that having less covered area by cameras leads to less accuracy, when $\rho > 1$. Although this is not always the case, as illustrated in the first column of these figures (i.e. for $\rho = 1$), the RMSE values are comparable among the same algorithms. Indeed, even though Scenario 1 has a larger covered area, it also has more sensors, which results in more consensus communication steps needed for the convergence of all the estimators, accentuated by the fact that

$\rho = 1$. For this reason, even with fewer communication links than in Scenario 1, the convergence is faster as we can clearly see by comparing the curves in Figs. 11 and 15 around time $t = 88$ s.

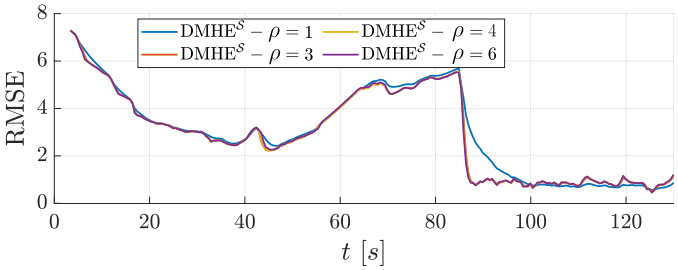


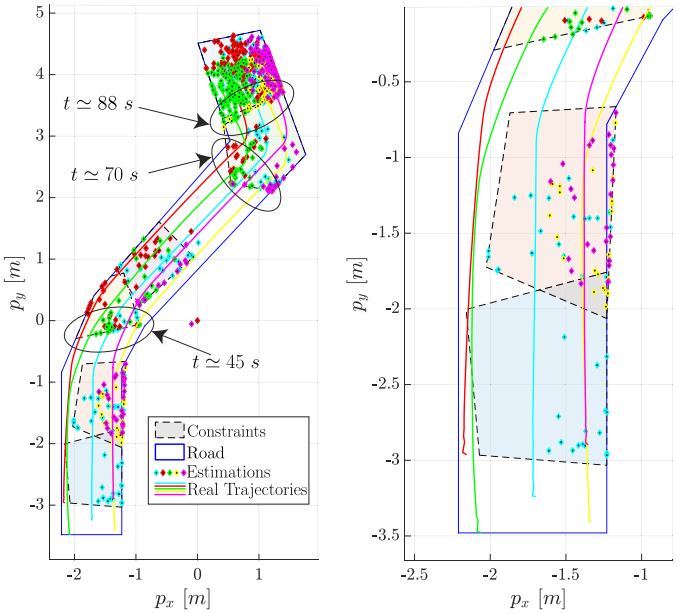
Figure 15: Scenario 2: RMSE of DMHE^S over time with a different communication link radius.

Furthermore, Fig. 15 shows that the RMSE value obtained for $\rho > 1$ (red, yellow and purple curves) is better than the one with $\rho = 1$ (blue curve) only until the vehicles stop, around $t = 88$ s. Thus, in the end, the five vehicles are only detected by sensor 12, which is the only active sensor, thus the only one using constraints as in (5). Moreover, it sends its measurements to its neighbors, which contribute to the global state estimation convergence without using the constraints. Hence, the more neighbors sensor 12 has, the more they contribute to the global estimation without constraints, i.e. emphasizing measurements noise and biased sensor data. To avoid this, a solution could have been to directly consider whether the sensor is active or not in the observability rank-based weights technique (Section 3.5). It might lead to better estimation accuracy.

Figure 16a and its zoom (right hand side of Fig. 16b) show the planar trajectories of the vehicles (solid lines) by the motion capture system and their respective estimates (rhombus) by the active sensors. Different colors highlight the vehicles: cyan, red, green, yellow, and magenta refer to vehicles 1, 2, 3, 4, and 5, respectively. This figure also shows the road boundaries (blue line) as well as the constraints (dashed polygones), as considered in (5). It is worth noticing that in Scenario 2 most of the time the vehicles are not detected (see Fig. 16a). Indeed, vehicles 2 (red rhombus) and 3 (green rhombus) began to be detected around $t \simeq 45$ s (see the zoom Fig. 16b where the green and red rhombuses appear only inside the light orange dashed polygone, on the top). Moreover, after some moments when no vehicle is detected, vehicles 1, 2, 3 and 5 are detected again around $t \simeq 70$ s (see the corresponding rhombuses in Fig. 16a). Only nearby the arrival point all the vehicles are detected at once, *cerca* $t \simeq 88$ s (see Fig. 16a). In addition, Fig. 15 shows that around the time instants $t \simeq 45$ s, $t \simeq 70$ s, and $t \simeq 88$ s when the vehicles are detected again, the RMSE decreases and it drops significantly especially around $t \simeq 88$ s since all the vehicles are detected (thus leading to a very small value for the RMSE).

4.3.3. Scenario 3

The goal of the last scenario (see Fig. 5) is to evaluate the performance of the proposed DMHE when the poses of the cameras are such that a single camera cannot detect all the vehicles at once, except sensor 12. Moreover, 10 trials have been run to show the robustness against different realization of measurement noise and bias. The provided results are averaged among the 10 trials (only for Scenario 3) and compared with the previous scenarios (notice that only one trial is considered for Scenarios 1 and 2). The radius communication link is $\rho = 1$.



(a) DMHE with constraints (b) Zoom

Figure 16: Scenario 2: Planar trajectory of the vehicles and estimates with constraints (left) and zoom (right).

Figure 17 illustrates the RMSEs of all scenarios considering the proposed DMHE algorithm with constraints DMHE^S. The green curve refers to the RMSE of the third scenario which also shows the bounds of the minimum and maximum RMSE of each trial (dashed lines). As expected, Scenario 3 (green curve) offers better accuracy than Scenario 2 (red curve) until the vehicles stop, since Scenario 3 has double of the sensors w.r.t. Scenario 2, thus leading to a larger covered area. Moreover, Scenario 3 (green curve) has always worse accuracy than Scenario 1 (blue curve), since Scenario 3 has the same number of sensors as Scenario 1 but less covered area. It can also be noticed that around $t \simeq 88$ s, the RMSE drops since all the vehicles are detected.

5. Conclusion and perspectives

This paper 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 priori* unknown. The proposed approach, which considers measurement constraints, has been implemented on a real Sensor Network composed of several low-cost cameras, each of them attached to a RaspBerry Pi for distributed implementation of the algorithms

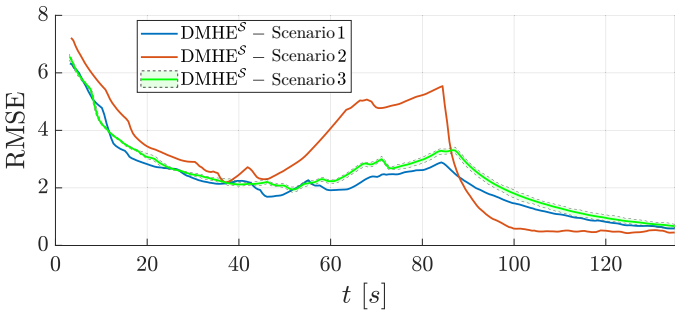


Figure 17: Comparison between the RMSE of the DMHE^S for the 3 scenarios for the $\rho = 1$, with an average among 10 trials for Scenario 3.

(using Robot Operating System (ROS) middleware) and network communication. The objective was to track a fleet of autonomous vehicles (ground robots) moving in an urban-like environment with road constraints. Three different experimental scenarios evaluated distinct aspects of the proposed algorithm, which differ in the number of sensors involved, the topology of the communication network, and the covered area by the cameras' fields of view.

The computation time of the proposed DMHE has been deceased by a factor two w.r.t. the time needed by the one of Farina et al. (2010). This result is obtained thanks to the pre-estimation observer included in the optimization problem that replaces the need to estimate the sequence of unknown inputs of the model over the estimation window and therefore leads to fewer optimization parameters. Taking advantage of constraint handling in online optimization required by MHE, the proposed algorithm exploits *a priori* information as environmental constraints (such as the road boundaries) to better estimate the state of the system. Moreover, the proposed DMHE formulation can deal with sporadic measurements thanks to the time-varying binary parameters embedded into the algorithm. Finally, it improves the accuracy of the estimation by utilizing an observability rank-based method to adjust the components of the consensus matrix associated with the graph of the Sensor Network. This particular aspect is particularly well suited to cases with sporadic measurements, as the one considered in this paper.

Several experiments have been realized, and collected data have been re-executed off-line in order to make rigorous performance analysis and comparison of the proposed DMHE with one of reference DMHE algorithms in literature Farina et al. (2010) and analyzing the effect of changing some properties of the application. Indeed, results have been shown in terms of accuracy by changing the number of sensors composing the network and the communication topology among neighbor nodes.

The proposed algorithm could be further extended by taking into account active and non active sensors when computing the consensus terms. Furthermore, in a context of fault detection, the possible sensor faults can be treated in the same manner as non active sensors, extending the application field of the current paper.

6. Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

7. Acknowledgment

The authors thankfully acknowledge Aarsh Thakker (Research Engineer L2S) for helping with the experiments.

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

Battistelli, G., 2018. Distributed moving-horizon estimation with arrival-cost consensus. IEEE Transactions on Automatic Control 64, 3316–3323.

Dong, X., Battistelli, G., Chisci, L., Cai, Y., 2022. Consensus variational bayesian moving horizon estimation for distributed sensor networks with unknown noise covariances. Signal Processing , 108571.

Duan, P., Qian, J., Wang, Q., Duan, Z., Shi, L., 2022. Distributed state estimation for continuous-time linear systems with correlated measurement noise. IEEE Transactions on Automatic Control , 1–15.

Famularo, D., Franzè, G., Tedesco, F., 2022. Fault detection and isolation for uncertain linear systems: A robust moving horizon estimation scheme using LMIs. Journal of the Franklin Institute 359, 1692–1712.

Farina, M., Ferrari-Trecate, G., Scattolini, R., 2010. Distributed moving horizon estimation for linear constrained systems. IEEE Transactions on Automatic Control 55, 2462–2475.

Farina, M., Ferrari-Trecate, G., Scattolini, R., 2012. Distributed moving horizon estimation for nonlinear constrained systems. International Journal of Robust and Nonlinear Control 22, 123–143.

Ferrante, F., Gouaisbaut, F., Sanfelice, R.G., Tarbouriech, S., 2016. State estimation of linear systems in the presence of sporadic measurements. Automatica 73, 101–109.

Foresti, G.L., Snidaro, L., 2002. A distributed sensor network for video surveillance of outdoor environments, in: International Conference on Image Processing, pp. I–I.

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

Goldfarb, D., Idnani, A., 1983. A numerically stable dual method for solving strictly convex quadratic programs. Mathematical programming 27, 1–33.

Haber, A., Verhaegen, M., 2013. Moving horizon estimation for large-scale interconnected systems. IEEE Transactions on Automatic Control 58, 2834–2847.

Halsted, T., Shorinwa, O., Yu, J., Schwager, M., 2021. A survey of distributed optimization methods for multi-robot systems. arXiv preprint arXiv:2103.12840 , 1–17.

He, S., Shin, H.S., Xu, S., Tsourdos, A., 2020. Distributed estimation over a low-cost sensor network: A review of state-of-the-art. Information Fusion 54, 21–43.

J. Zeng, Liu, J., 2015. Distributed moving horizon estimation subject to communication delays and losses, in: American Control Conference, pp. 5533–5538.

Karg, B., Lucia, S., 2021. Approximate moving horizon estimation and robust nonlinear model predictive control via deep learning. Computers & Chemical Engineering 148, 107266.

Kim, J., Kang, J.H., Bae, J., Lee, W., Kim, K.K.K., 2021. Distributed moving horizon estimation via operator splitting for automated robust power system state estimation. IEEE Access 9, 90428–90440.

Manfredi, S., 2013. Design of a multi-hop dynamic consensus algorithm over wireless sensor networks. Control Engineering Practice 21, 381–394.

Millán, P., Orihuela, L., Vivas, C., Rubio, F., Dimarogonas, D.V., Johansson, K.H., 2013. Sensor-network-based robust distributed control and estimation. Control Engineering Practice 21, 1238–1249.

Muntwiler, S., Wabersich, K.P., Zeilinger, M.N., 2022. Learning-based moving horizon estimation through differentiable convex optimization layers, in: Firoozi, R., Mehr, N., Yel, E., Antonova, R., Bohg, J., Schwager, M., Kochenderfer, M. (Eds.), Proceedings of The 4th Annual Learning for Dynamics and Control Conference, PMLR. pp. 153–165. URL: https://proceedings.mlr.press/v168/muntwiler22a.html.

Muske, K.R., Rawlings, J.B., Lee, J.H., 1993. Receding horizon recursive state estimation, in: American Control Conference, pp. 900–904.

Negenborn, R., Maestre, J., 2014. Distributed Model Predictive Control: An overview and roadmap of future research opportunities. IEEE Control Systems Magazine 34, 87–97.

Olfati-Saber, R., 2007. Distributed kalman filtering for sensor networks, in: 2007 46th IEEE Conference on Decision and Control, IEEE. pp. 5492–5498.

Petitti, A., Di Paola, D., Rizzo, A., Cicirelli, G., 2011. Consensus-based distributed estimation for target tracking in heterogeneous sensor networks, in: 50th IEEE Conference on Decision and Control and European Control Conference, pp. 6648–6653.

Postoyan, R., Nešić, D., 2011. A framework for the observer design for networked control systems. IEEE Transactions on Automatic Control 57, 1309–1314.

Rao, C.V., Rawlings, J.B., Lee, J.H., 2001. Constrained linear state estimation — a moving horizon approach. Automatica 37, 1619–1628.

Rego, F.F., Pascoal, A.M., Aguiar, A.P., Jones, C.N., 2019. Distributed state estimation for discrete-time linear time invariant systems: A survey. Annual Reviews in Control 48, 36–56.

Segovia, P., Puig, V., Duviella, E., 2021. Set-membership-based distributed moving horizon estimation of large-scale systems. ISA Transactions (*in press*) , 1–12.

Sharma, A., Chauhan, S., 2020. Sensor fusion for distributed detection of mobile intruders in surveillance wireless sensor networks. IEEE Sensors Journal 20, 15224–15231.

Shorinwa, O., Yu, J., Halsted, T., Koufos, A., Schwager, M., 2020. Distributed multi-target tracking for autonomous vehicle fleets, in: IEEE International Conference on Robotics and Automation, pp. 3495–3501.

Simonetto, A., Balzaretti, D., Keviczky, T., 2011. A distributed moving horizon estimator for mobile robot localization problems. IFAC Proceedings Volumes 44, 8902–8907.

Sui, D., Johansen, T.A., 2014. Linear constrained moving horizon estimator with pre-estimating observer. Systems & Control Letters 67, 40–45.

Suwantong, R., Bertrand, S., Dumur, D., Beauvois, D., 2014. Stability of a nonlinear moving horizon estimator with pre-estimation, in: American Control Conference, pp. 5688–5693.

Vadigepalli, R., Doyle III, F.J., 2003. Structural analysis of large-scale systems for distributed state estimation and control applications. Control Engineering Practice 11, 895–905.

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

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

Venturino, A., Bertrand, S., Stoica Maniu, C., Alamo, T., Camacho, E.F., 2022. Multi-vehicle system localization by distributed moving horizon estimation over a sensor network with sporadic measurements, in: 6th IEEE Conference on Control Technology and Applications.

Venturino, A., Stoica Maniu, C., Bertrand, S., Alamo, T., Camacho, E.F., 2021b. Distributed moving horizon state estimation for sensor networks with low computation capabilities. System Theory, Control and Computing Journal 1, 81–87.

Viegas, D., Batista, P., Oliveira, P., Silvestre, C., 2018. Discrete-time distributed Kalman filter design for formations of autonomous vehicles. Control Engineering Practice 75, 55–68.

Vukov, M., Gros, S., Horn, G., Frison, G., Geebelen, K., Jørgensen, J.B., Swevers, J., Diehl, M., 2015. Real-time nonlinear MPC and MHE for a large-scale mechatronic application. Control Engineering Practice 45, 64–78.

Wang, J., Olson, E., 2016. AprilTag 2: Efficient and robust fiducial detection, in: 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE. pp. 4193–4198. doi:10.1109/IROS.2016.7759617.

Yin, X., Huang, B., 2022. Event-triggered distributed moving horizon state estimation of linear systems. IEEE Transactions on Systems, Man, and Cybernetics: Systems , 1–13.

Yin, X., Liu, J., 2017. Distributed moving horizon state estimation of two-time-scale nonlinear systems. Automatica 79, 152–161.

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

Zhang, J., Liu, J., 2013. Distributed moving horizon state estimation for nonlinear systems with bounded uncertainties. Journal of Process Control 23, 1281–1295.