Distributed decentralized EKF for very large-scale networks with application to satellite mega-constellations navigation

Leonardo Pedroso[1] and Pedro Batista[1]

[1] *Institute for Systems and Robotics, Instituto Superior Técnico, Universidade de Lisboa, Portugal*

Published version[1]

Published on 25 Mar 2023

Cite as

L. Pedroso and P. Batista, 'Distributed decentralized EKF for very large-scale networks with application to satellite mega-constellations navigation', *Control Engineering Practice*, vol. 135, 2023. doi: 10.1016/j.conengprac.2023.105509.

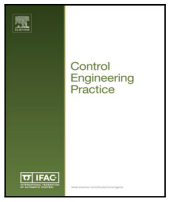
[1] Level of access, as per info available on Sherpa Romeo: `https://v2.sherpa.ac.uk/id/publication/4620` (25 Mar 2023)

Control Engineering Practice 135 (2023) 105509

Contents lists available at ScienceDirect

Control Engineering Practice

journal homepage: www.elsevier.com/locate/conengprac



Distributed decentralized EKF for very large-scale networks with application to satellite mega-constellations navigation

Leonardo Pedroso *, Pedro Batista

Institute for Systems and Robotics, Instituto Superior Técnico, Universidade de Lisboa, Portugal

ARTICLE INFO

Keywords:
Extended Kalman filter
Decentralized filtering
Distributed robot systems
Networked robots
Space robotics

ABSTRACT

The design of a decentralized and distributed filtering solution for large-scale networks of interconnected systems is addressed considering (i) generic nonlinear dynamics and (ii) generic coupled nonlinear outputs in a generic, possibly time-varying, topology. The local filters, which follow the structure of the extended Kalman filter, are implemented in each system, which estimates its own state exclusively. To be suitable for the heavily restricted implementation to very large-scale systems, a novel algorithm is proposed, which: (i) does not rely on instantaneous data transmission; (ii) allows local communication exclusively; and (iii) requires computational, memory, and data transmission resources for each system that do not scale with the dimension of the network. The scalability of the proposed algorithm allows for its application to the cooperative localization problem of very large-scale systems. In particular, it is applied herein to the on-board position estimation problem of LEO mega-constellations using GNSS featuring numerical simulations for the Starlink constellation.

1. Introduction

1.1. Motivation

Over the past decades, decentralized estimation and control have been highly researched topics since they provide a solution to the estimation and control problems of large-scale networks of interconnected systems. In fact, they emerge as an alternative to the use of well-known centralized solutions, which become unfeasible to implement as the dimension of the network increases. The popularity of decentralized solutions is also increasing with the widening of its applications to a broad range of engineering fields. Examples of such applications are unmanned aircraft formation flight (Bereg, Díaz-Báñez, Lopez, Rozario, & Valavanis, 2015; Thien & Kim, 2018), unmanned underwater formations (Viegas, Batista, Oliveira, & Silvestre, 2012; Yuan, Licht, & He, 2017), satellite formation control (Ivanov, Monakhova, & Ovchinnikov, 2019; Russell Carpenter, 2002), and irrigation networks (Li, 2014; Prodan, Lefevre, Genon-Catalot, et al., 2017). Although plenty of work has been carried out in decentralized control and estimation of linear systems, the problem of designing such controllers, which consists in solving an optimization problem subject to a constraint that arises from the distributed nature of the configuration, is extremely difficult (Blondel & Tsitsiklis, 2000) and remains an open problem. Having said that, a large portion of the systems for which control and estimation solutions have to be designed are, in practice, nonlinear. In these cases, it is a common practice to make use of linearization techniques about successive operation points, for which the solution is given by linear control and estimation solutions. For the estimation problem in particular, the well-known extended Kalman filter (EKF) and multiple of its variations have been used for decades, yielding good results in a broad range of applications (Chen, 2011; Crassidis, Markley, & Cheng, 2007). Nevertheless, when it comes to the application of the EKF in a decentralized and distributed framework on a large-scale network of interconnected systems, on top of the aforementioned intricacies of decentralized estimator design for linear systems, additional complications arise. These concern the increased communication demands of the EKF, which are unfeasible to implement in large-scale systems.

This paper addresses the problem of the design of decentralized and distributed filtering solutions to very large-scale networks of interconnected systems considering: (i) generic nonlinear dynamics and (ii) generic coupled nonlinear outputs in a generic, possibly time-varying, topology, following the approach of the EKF. In this framework, each system estimates exclusively its own state with access to local output measurements and local communication. This framework differs significantly from the consensus EKF problem, for which distributed solutions have also been proposed (Li, Jia, & Du, 2017). The implementation of filtering solutions to large-scale interconnected systems is very challenging due to the limited communication between systems, and limited computational power and memory, which are distributed across the systems. There are few satisfactory solutions to this problem detailed in the literature. First, several centralized-equivalent approaches have been proposed. In Dai, Mu, and Wu (2016) the application of a junction-tree-based protocol for the distributed implementation of the centralized EKF is proposed. In Leung, Barfoot,

* Corresponding author.
E-mail address: leonardo.pedroso@tecnico.ulisboa.pt (L. Pedroso).

https://doi.org/10.1016/j.conengprac.2023.105509
Received 26 September 2022; Received in revised form 14 March 2023; Accepted 14 March 2023
Available online xxxx

and Liu (2009) the cooperative localization problem makes use of extensive measurement storage that scales with the size of the network. Notwithstanding the lack of loss of performance in relation to the centralized EKF, the communication and memory burden, respectively, of these approaches is massive, which renders its application to a very large-scale network unfeasible. Second, it is possible to implement an EKF making use of recent decentralized filtering solutions for linear time-varying systems (Pedroso, Batista, Oliveira, & Silvestre, 2021). The fact that the dynamics of the systems have to be successively linearized about different operation points requires all-to-all communication, which renders this approach infeasible to large-scale nonlinear networks. Third, another well-known approach is to approximate the covariance propagation to meet the requirements for a feasible implementation (Madhavan, Fregene, & Parker, 2002, 2004). However, this scheme is prone to become over-confident by underestimating the correlation between measurements, which is designated as double-counting. This effect is exemplified in Panzieri, Pascucci, and Setola (2006). In Martinelli (2007) this approach is also followed to devise an hierarchical EKF dividing the network into groups of M systems. In each group, there is a leader node which processes all the measurements of the group with a computational cost $\mathcal{O}(M^4)$. Covariance intersection and split covariance intersection methods have been proposed to mitigate double-counting (Carrillo-Arce, Nerurkar, Gordillo, & Roumeliotis, 2013; Julier & Uhlmann, 1997b; Li, Nashashibi, & Yang, 2013; Wanasinghe, Mann, & Gosine, 2014). In particular, in Carrillo-Arce et al. (2013), at the expense of overly pessimistic estimates, computational and communication complexities that do not scale with the dimension of the network are achieved. Nevertheless, this result is only attained if a system can predict the state of a neighboring system resorting to its own state estimate and a relative measurement. This is the case in most problems tackled in the cooperative localization state-of-the-art, but it is not the case for networks whose systems have general nonlinear dynamics and general nonlinear coupled outputs, as it considered in this paper. In fact, it is not applicable to the on-board position estimation problem of LEO mega-constellations, which is considered as an application example of the method proposed in this paper. Recently, in Luft, Schubert, Roumeliotis, and Burgard (2016, 2018), Luft et al. proposed a promising recursive decentralized method that computes an approximate correlation distributively between each pair of systems for a network with a generic measurement model and supports asynchronous communication and measurements. This solution achieves good performance and a communication burden that does not scale with the dimension of the network. Although it does not rely on measurement bookkeeping, the memory requirements scale linearly with the dimension of the network. Furthermore, although an approximation error analysis is conducted, the solution is not provably consistent. The emergence of very large-scale networks, with generic nonlinear dynamics, calls for very efficient filtering algorithms, whose communication, memory, and computational requirements do not scale with the size of the network. Although some results in the cooperative localization community satisfy these constraints, they are not valid for generic dynamics and generic output measurement models. Nevertheless, similar insight can be leveraged to solve this problem.

In this paper, the decentralized estimation problem is addressed in a fully decentralized configuration, i.e., each system can only make use of its own measurements. Furthermore, severe requirements are imposed for the successful implementation in the emerging very large-scale networks of interconnected systems. The data transmissions cannot be considered instantaneous, i.e., the filtering solutions must cope with and account for delays in data transmissions. The number of systems with which each system communicates must not scale with the dimension of the network, N, i.e., the communication complexity ought be $\mathcal{O}(1)$. The quantity of transmitted data in each discrete-time instant and the memory requirements of each system must not scale with the dimension of the network. The computational load of the filtering solution must be distributed across all systems, and the computational complexity of each system must not scale with the dimension of the network, i.e., the computational complexity ought to be $\mathcal{O}(1)$. An important distinction is made between the terms distributed and decentralized. Although these are, oftentimes, used interchangeably by many authors, in this work they characterize control algorithms with different characteristics. Herein, a control solution is said to be decentralized if the implementation of the control law in each system can be deployed resorting to local communication exclusively. In contrast, a control solution is said to be distributed if its control law can be synthesized in real-time in a distributed manner across the systems of the network resorting to local communication exclusively. Thus, the inevitable paradigm change towards very large-scale networks calls for distributed and decentralized control algorithms.

The problem is formulated for a linear time-varying system, relying on the EKF framework. Similarly to Luft et al. (2016, 2018), the decoupling of the covariance propagation is achieved by introducing an approximation. In the proposed method, each system $\mathcal{S}_i$ keeps an estimate of the estimation error covariance between every pair of systems with which it is output coupled. These are updated in $\mathcal{S}_i$ considering that the estimation error covariance between systems that are not coupled with a common system with which $\mathcal{S}_i$ is coupled are neglected. In what follows, the approximation is formally presented, its sense in the context of estimation is explored, and its role in decoupling the filter dynamics is thoroughly detailed. To assess the performance of the proposed method and validate its scalability to very large-scale networks of systems, it is implemented to a mega-constellation of satellites in low Earth orbit (LEO). This example is a very large-scale system with nonlinear dynamics that has emerged very recently. Decentralized onboard filtering solutions are of paramount importance for these constellations, on which their economical viability is dependent. A MATLAB implementation of the proposed algorithm and all the source code of the simulations can be found as an example of the DECENTER Toolbox available at https://decenter2021.github.io/examples/DDEKFStarlinkConstellation/.

1.2. Motivation for application in LEO mega-constellations

Inspired by the emergence in various engineering fields, over the past decades, of decentralized solutions as an alternative to the use of these well-known centralized solutions, the management of satellite mega-constellations could also be carried out in this fashion in a decentralized tracking telemetry and command (TT&C) architecture. In a decentralized configuration, low level constellation operations, such as orbit determination and constellation control, are carried out cooperatively and local communication between satellites is assured by inter-satellite links (ISL). The gains in efficiency and cost effectiveness of such a paradigm shift are evident. First, only local communication between satellites in close proximity is required, which is assured by ISL. Second, by employing cooperative algorithms that run in a distributed configuration across the whole network, all-to-all communication is no longer required, which dramatically reduces the quantity of data that is transmitted. Third, as the constellation management is carried out in a distributed manner, the computational load is shared across multiple satellites. Fourth, the ISL are more secure and require less power, given the proximity of the endpoints. For these reasons, the cost effectiveness of LEO mega-constellations would greatly improve with the adoption of a decentralized architecture. However, little research has been undergone on distributed algorithms for systems of very large-scale. As mentioned previously in this section, for the particular problem of decentralized control and estimation, even though there are state-of-the-art algorithms that yield good performance, they either: (i) require all-to-all communication; (ii) extensive measurement bookkeeping; or (iii) are not computationally scalable. Decentralized orbit determination and control has already been given significant attention for small and closely packed formations of satellites (Busse & How, 2002; Ferguson & How, 2003; Mandic, Breger, & How, 2004; Park, Ferguson, Pohlman, & How, 2001; Wang, Qin, Bai, & Cui, 2016).

However, these methods do not meet the aforementioned strict computational, memory, and communication requirements that enable the application to large-scale constellations of satellites. One of the few works to address the decentralized constellation navigation problem is Dai et al. (2016), at the expense of unbearable computational load, since a centralized-equivalent approach is proposed.

The algorithm proposed in this paper is applied to the decentralized navigation problem for LEO mega-constellations. The communication, computational, and memory constraints followed by the proposed method allow for the real-time implementation in the new generation of LEO mega-constellations. The application implemented in this paper makes use of a GNSS system to provide each satellite with inertial absolute position measurements and also relative position measurements between satellites in close proximity, obtained with carrier-phase differential GNSS over long baselines (Blewitt, 1989; Tancredi, Renga, & Grassi, 2014; Wolfe, Speyer, Hwang, Lee, & Lee, 2007; Wu & Bar-Sever, 2006). The fusion of inertial position measurements with relative position measurements, which are significantly more accurate, allows for a significant increase in absolute estimation performance than if only inertial GNSS measurements were used. Although ISL are required for taking the relative measurements and fusing them with the inertial measurements, that is, oftentimes, already a requirement for the operation of the mega-constellation. It is important to remark that the decentralized relative navigation problem for small satellite formations using carrier-phase differential GNSS, which has been extensively studied in the aforementioned works (Busse & How, 2002; Ferguson & How, 2003; Park et al., 2001), is very distinct from the application considered herein. While in these works the goal is to estimate the relative position between satellites, for *rendezvous* or sensing applications, in this paper the aim is to significantly improve inertial position estimates by considering relative measurements. The increase in position estimation accuracy achieved by the proposed method allows for: (i) more densely packed constellations; (ii) increased efficiency in the constellation-keeping control problem, thereby saving fuel; and (iii) increased efficiency in obstacle avoidance, requiring smaller obstacle clearance. Furthermore, it can also enable large-scale remote sensing applications, whose position accuracy requirements are very strict, often calling for meter-level or sub-meter-level accuracy. Examples are altimetry, gravimetry, SAR interferometry, and atmospheric sounding.

1.3. Paper outline

This paper is organized as follows. In Section 2, the decentralized estimation problem is formulated alongside the communication, computational, and memory constraints the estimation solution must follow to allow for its implementation in real-time and for very large-scale systems. In Section 3, the distributed and decentralized filtering EKF filtering algorithm is derived. In Section 4, the decentralized EKF algorithm proposed in this paper is applied to the cooperative navigation of LEO mega-constellations using a GNSS. Section 5 presents the main conclusions of this paper.

1.4. Notation

Throughout this paper, the identity, null, and ones matrices, all of proper dimensions, are denoted by $\mathbf{I}$, $\mathbf{0}$, and $\mathbf{1}$, respectively. Alternatively, $\mathbf{I}_n$, $\mathbf{0}_{n\times m}$, and $\mathbf{1}_{n\times m}$ are also used to represent the $n\times n$ identity matrix and the $n\times m$ null and ones matrices, respectively. The column-wise concatenation of vectors $\mathbf{x}_1,\dots,\mathbf{x}_N$ is denoted by $\mathrm{col}(\mathbf{x}_1,\dots,\mathbf{x}_N)$ and $\mathrm{diag}(\mathbf{A}_1,\dots,\mathbf{A}_N)$ denotes the square block diagonal matrix whose diagonal blocks are given by matrices $\mathbf{A}_1,\dots,\mathbf{A}_N$. Given a symmetric matrix $\mathbf{P}$, $\mathbf{P}\succ\mathbf{0}$ and $\mathbf{P}\succeq\mathbf{0}$ are used to point out that $\mathbf{P}$ is positive definite and positive semidefinite, respectively. The expected value of a random variable X is denoted as $\mathrm{E}[X]$. The cross-product between two vectors $\mathbf{a},\mathbf{b}\in\mathbb{R}^3$ is denoted by $\mathbf{a}\times\mathbf{b}$. The Kronecker product of two matrices $\mathbf{A}$ and $\mathbf{B}$ is denoted by $\mathbf{A}\otimes\mathbf{B}$. The cardinality of a set $\mathcal{A}$ is denoted by $|\mathcal{A}|$. The Cartesian product of two sets $\mathcal{A}$ and $\mathcal{B}$ is denoted by $\mathcal{A}\times\mathcal{B}$.

2. Problem statement

The statement of the novel decentralized extended Kalman filter for very large-scale networks of systems with coupled outputs and time-varying topology is carried out in three steps. First, in Section 2.1, the local models of each of the systems are presented, which are posteriorly concatenated to define a global model for the network. Second, in Section 2.2, the local extended Kalman filters are detailed and the estimation problem is formulated for the global filter. Third, in Section 2.3, communication, computational, and memory constraints are defined. It is important to point out that this problem is stated and addressed for a generic network with nonlinear output couplings between systems, making no assumptions neither on the dynamics of each system, nor on the output function, nor on the topology of the network.

2.1. Model of a network of coupled systems

Consider a network of N interconnected systems, $\mathcal{S}_i$, each associated with one computing unit $\mathcal{T}_i$, with $i=1,\dots,N$. Each system has decoupled nonlinear dynamics and has access to nonlinear sensor outputs, which are coupled with a set of other subsystems. The topology of this network is, thus, defined by the output couplings between systems, and it may be time-varying. Such output coupling topology may be represented by a directed graph, or digraph, $\mathcal{G}:=(\mathcal{V},\mathcal{E})$, composed of a set $\mathcal{V}$ of vertices and a set $\mathcal{E}$ of directed edges. An edge e incident on vertices i and j, directed from j towards i, is denoted by $e=(j,i)$. For a vertex i, its in-degree, ν_i^-, is the number of edges directed towards it, and its in-neighborhood, $\mathcal{D}_i^-$, is the set of indices of the vertices from which such edges originate. Conversely, for a vertex i, its out-degree, ν_i^+, is the number of edges directed from it, and its out-neighborhood, $\mathcal{D}_i^+$, is the set of indices of the vertices towards which such edges are directed. For a more detailed overview of the elements of graph theory used to model this network, see Wallis (2010) and West et al. (1996). In this framework, each system is represented by a vertex, i.e., system $\mathcal{S}_i$ is represented by node i, and if $\mathcal{S}_i$ has access to an output that depends on the state of system $\mathcal{S}_j$, then this coupling is represented by an edge directed from vertex j towards vertex i, i.e., edge $e=(j,i)$. It is important to stress that the direction of the edge matters. Note, for instance, that the fact that $\mathcal{S}_i$ has access to an output that depends on the state of system $\mathcal{S}_j$ does not, necessarily, imply the converse. Also, given that the goal of each system $\mathcal{S}_i$ is to estimate its state, it is assumed henceforth, without loss of generality, that the output of system $\mathcal{S}_i$ depends on its state, i.e., $i\in\mathcal{D}_i^-$. No further assumptions are made on the topology of the directed graph. In particular, it is not assumed to be either cyclic or acyclic.

The dynamics of system $\mathcal{S}_i$ are modeled by the discrete-time nonlinear system

$$\begin{cases}\mathbf{x}_i(k+1)=\mathbf{f_i}(k,\mathbf{x}_i(k),\mathbf{u}_i(k))+\mathbf{w}_i(k)\\ \mathbf{y}_i(k)=\mathbf{g_i}(k;\mathbf{x}_j(k),j\in\mathcal{D}_i^-)+\mathbf{v}_i(k),\end{cases}\tag{1}$$

where $\mathbf{x}_i(k)\in\mathbb{R}^{n_i}$ is the state vector, $\mathbf{u}_i(k)\in\mathbb{R}^{m_i}$ is the input vector, which is assumed to be known, and $\mathbf{y}_i(t)\in\mathbb{R}^{o_i}$ is the output vector, all of system $\mathcal{S}_i$; functions $\mathbf{f_i}:\mathbb{R}^{n_i}\times\mathbb{R}^{m_i}\to\mathbb{R}^{n_i}$ and $\mathbf{g_i}:\prod_{j\in\mathcal{D}_i^-}\mathbb{R}^{n_j}\to\mathbb{R}^{o_i}$ are known multivariate functions that model the dynamics of system $\mathcal{S}_i$ and its output coupling with the other systems in its in-neighborhood; vector $\mathbf{w}_i(k)\in\mathbb{R}^{n_i}$ is the process noise, modeled as a zero-mean, white Gaussian process with associated covariance matrix $\mathbf{Q}_i(k,\mathbf{x}_i(k),\mathbf{u}_i(k))\succeq\mathbf{0}\in\mathbb{R}^{n_i\times n_i}$, vector $\mathbf{v}_i(k)\in\mathbb{R}^{o_i}$ is the observation noise, modeled as a zero-mean, white Gaussian process. It is considered that the observation noise vectors $\mathbf{v}_i(k)$ and $\mathbf{v}_j(k)$ of system $\mathcal{S}_i$ and $\mathcal{S}_j$, respectively, are correlated only if their outputs are coupled with a common system $\mathcal{S}_p$. To illustrate the need to introduce such a correlation consider, as an example, relative measurements between submarines using ultrasonic signals. The relative measurement between a submarine i and a submarine p and the relative measurement between

a submarine j and a submarine p are obtained by processing ultrasonic signals that, depending on the position of the three submarines, may have propagation paths near each other. In that case, the measurement noise induced by propagation delays related with the properties of the propagation medium (for example water temperature or salinity) are correlated. The observation noise process, thus, follows

$$\mathrm{E}[\mathbf{v}_i(k)\mathbf{v}_i^T(k)] = \mathbf{R}_{i,i}(k; \mathbf{x}_j(k), j \in \mathcal{D}_i^-) \succ \mathbf{0} \in \mathbb{R}^{o_i \times o_i} \tag{2}$$

and

$$\mathrm{E}[\mathbf{v}_i(k)\mathbf{v}_j^T(k)] = \begin{cases} \mathbf{R}_{i,j}(k; \mathbf{x}_l(k), l \in \{i,j\} \cup \mathcal{D}_i^+ \cap \mathcal{D}_j^+), & \mathcal{D}_i^+ \cap \mathcal{D}_j^+ \neq \emptyset \\ \mathbf{0}_{\mathbf{o_i} \times \mathbf{o_j}}, & \text{otherwise,} \end{cases} \tag{3}$$

where $\mathbf{R}_{i,j}$ are covariance matrices that depend on time and on a set of states with which the outputs of systems $\mathcal{S}_i$ and $\mathcal{S}_j$ are both coupled.

The global dynamics of the network can, then, modeled as discrete-time nonlinear system of the form

$$\begin{cases} \mathbf{x}(k+1) = \mathbf{f}(k, \mathbf{x}(k), \mathbf{u}(k)) + \mathbf{w}(k) \\ \mathbf{y}(k) = \mathbf{g}(k; \mathbf{x}(k)) + \mathbf{v}(k), \end{cases} \tag{4}$$

where $\mathbf{x}(k) := \mathrm{col}(\mathbf{x}_1(k), \dots, \mathbf{x}_N(k)) \in \mathbb{R}^n$ is the global state vector, $\mathbf{u}(k) := \mathrm{col}(\mathbf{u}_1(k), \dots, \mathbf{u}_N(k)) \in \mathbb{R}^m$ is the global input vector, and $\mathbf{y}(k) := \mathrm{col}(\mathbf{y}_1(k), \dots, \mathbf{y}_N(k)) \in \mathbb{R}^o$ is the global output vector; $\mathbf{w}(k) := \mathrm{col}(\mathbf{w}_1(k), \dots, \mathbf{w}_N(k))$ is the process noise, modeled as a zero-mean, white Gaussian process with associated covariance matrix given by $\mathbf{Q}(k, \mathbf{x}(k), \mathbf{u}(k)) := \mathrm{diag}\left(\mathbf{Q}_1(k, \mathbf{x}_1(k), \mathbf{u}_1(k)), \dots, \mathbf{Q}_N(k, \mathbf{x}_N(k), \mathbf{u}_N(k))\right)$; $\mathbf{v}(k) := \mathrm{col}(\mathbf{v}_1(k), \dots, \mathbf{v}_N(k))$ is the global observation noise, modeled as a zero-mean, white Gaussian process with associated covariance matrix $\mathbf{R}(k, \mathbf{x}(k))$ defined by (2) and (3); and $\mathbf{f} : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ and $\mathbf{g} : \mathbb{R}^n \to \mathbb{R}^o$ model the dynamics and output of the global system, respectively, and are defined as $\mathbf{f}(k, \mathbf{x}(k), \mathbf{u}(k)) := \mathrm{col}\left(\mathbf{f_1}(k, \mathbf{x}_1(k), \mathbf{u}_1(k)), \dots, \mathbf{f_N}(k, \mathbf{x}_N(k), \mathbf{u}_N(k))\right)$ and $\mathbf{g}(k, \mathbf{x}(k)) := \mathrm{col}\left(\mathbf{g_1}(k, \mathbf{x}_1(k)), \dots, \mathbf{g_N}(k, \mathbf{x}_N(k))\right)$. Before proceeding with the statement of the estimation problem, it is worth pointing out that virtually all large-scale networks have sparse couplings. In particular, v_i^-, the number of coupling outputs of a system, is bounded and it is independent of N, the number of systems in the network. The novel solution presented in this paper takes advantage of the sparsity of these couplings to design a fully distributed extended Kalman filter under communication and computational power limitations.

2.2. Formulation of the decentralized estimation problem

In this paper, state estimation is assumed to be achieved by a dynamical filter based on prediction-update steps employed in a Kalman filter. In a centralized configuration, each system has access to the global output, at the expense of all-to-all communication via a central system. In a decentralized configuration, that is not the case: each system $\mathcal{S}_i$ only has access to a subset of the outputs. In this paper, a fully decentralized configuration is considered, in the sense that, at each discrete-time instant k, the only output known to $\mathcal{S}_i$ is $\mathbf{y}_i(k)$. This property is analyzed with more detail in Section 2.3. The prediction and update steps are, thus, given by

$$\begin{cases} \hat{\mathbf{x}}_i(k+1|k) = \mathbf{f_i}(\hat{\mathbf{x}}_i(k|k), \mathbf{u}_i(k)) \\ \hat{\mathbf{x}}_i(k+1|k+1) = \hat{\mathbf{x}}_i(k+1|k) \\ \qquad +\mathbf{K}_i(k+1)\left(\mathbf{y}_i(k+1) - \mathbf{g_i}(k+1; \hat{\mathbf{x}}_j(k+1|k), j \in \mathcal{D}_i^-)\right), \end{cases} \tag{5}$$

where $\hat{\mathbf{x}}_i(k+1|k)$ denotes the predicted state estimate of system $\mathcal{S}_i$ at instant $k+1$, $\hat{\mathbf{x}}_i(k+1|k+1)$ the updated state estimate of system $\mathcal{S}_i$ at instant $k+1$, and $\mathbf{K}_i(k+1)$, $i = 1, \dots, N$, the decentralized filter gains.

The goal is to design optimum filter gains $\mathbf{K}_i(k)$ according to a performance criteria. Note that, due to the coupling between systems in the update step of each local filter (5), each gain $\mathbf{K}_i(k)$, $i = 1, \dots, N$, cannot be designed independently. For that reason, the local filter dynamics of each system are concatenated to write the global filter dynamics, which are used to formulate a global estimation problem. The challenge is, then, to enforce the communication, computational, and memory requirements of a decentralized configuration in the solution of the corresponding optimization problem. The global prediction and update steps can be written as

$$\begin{cases} \hat{\mathbf{x}}(k+1|k) = \mathbf{f}(\hat{\mathbf{x}}(k|k), \mathbf{u}(k)) \\ \hat{\mathbf{x}}(k+1|k+1) = \hat{\mathbf{x}}(k+1|k) \\ \qquad +\mathbf{K}(k+1)\left(\mathbf{y}(k+1) - \mathbf{g}(k+1; \hat{\mathbf{x}}(k+1|k))\right), \end{cases} \tag{6}$$

where $\hat{\mathbf{x}}(k+1|k) := \mathrm{col}(\hat{\mathbf{x}}_1(k+1|k), \dots, \hat{\mathbf{x}}_N(k+1|k))$ denotes the global predicted state estimate at instant $k+1$, $\hat{\mathbf{x}}(k+1|k+1) := \mathrm{col}(\hat{\mathbf{x}}_1(k+1|k+1), \dots, \hat{\mathbf{x}}_N(k+1|k+1))$ the global updated state estimate at instant $k+1$, and $\mathbf{K}(k+1) \in \mathbb{R}^{n \times o}$ the global filter gain. Note that the global filter dynamics (6) are equivalent to the concatenation of the local filter dynamics (5) if and only if

$$\mathbf{K}(k+1) = \mathrm{diag}\left(\mathbf{K}_1(k+1), \dots, \mathbf{K}_N(k+1)\right). \tag{7}$$

More generally, (7) can be written as a sparsity constraint $\mathbf{K}(k+1) \in \mathrm{Sparse}(\mathbf{E})$, where $\mathbf{E} = \mathrm{diag}(\mathbf{1}_{n_1 \times o_1}, \dots, \mathbf{1}_{n_N \times o_N})$ is the sparsity pattern and

$$\mathrm{Sparse}(\mathbf{E}) = \left\{ [\mathbf{K}]_{ij} \in \mathbb{R}^{n \times o} : \right. \\ \left. [\mathbf{E}]_{ij} = 0 \implies [\mathbf{K}]_{ij} = 0;\ i = 1, \dots, n,\ j = 1, \dots, o \right\}.$$

If there were all-to-all communication, then $\mathbf{E} = \mathbf{1}$, which corresponds to a centralized configuration.

It is now possible to formulate the problem of designing a decentralized state estimation solution based on a Luenberger Kalman filter for the network of coupled systems with nonlinear dynamics. Note that designing a decentralized filter for a network of coupled systems, whose local dynamics are described by the nonlinear system (1), is equivalent to designing a global filter (6) for the global network (4), whose gain must follow a sparsity pattern. One aims to optimally compute a sequence of filter gains that follow the sparsity pattern required for a fully decentralized configuration. Note that the aforementioned sparsity constrain (7) allows to formulate the decentralized problem globally explicitly considering the communication restrictions associated with the decentralized setting. For an infinite-horizon, solve the optimization problem

$$\begin{aligned} \underset{\substack{\mathbf{K}(i) \in \mathbb{R}^{n \times o} \\ i \in \mathbb{N}}}{\text{minimize}} \quad & \lim_{T \to +\infty} \frac{1}{T} \sum_{k=1}^{T} \mathrm{tr}(\mathbf{P}(k|k)) \\ \text{subject to} \quad & \mathbf{K}(i) \in \mathrm{Sparse}(\mathrm{diag}(\mathbf{1}_{n_1 \times o_1}, \dots, \mathbf{1}_{n_N \times o_N})) \end{aligned} \tag{8}$$

where $\mathbf{P}(k|k) \succeq \mathbf{0} \in \mathbb{R}^{n \times n}$ is the global covariance of the state estimation error at instant k after the update step. The difficulty of solving this problem is twofold. On one hand, even for systems with linear dynamics, the optimization problem (8) is nonconvex and its optimal solution is still an open problem. To overcome this difficulty, the optimization problem may be relaxed so that it becomes convex, allowing for the use of well-known optimization techniques. Albeit optimal for the modified problem, the relaxed solution is only an approximation to the solution of the original problem. For this reason, careful relaxation is necessary to ensure that the separation between both solutions is minimal. This approach is designated convex relaxation and will be used to derive the method put forward in this paper. On the other hand, given that the system is nonlinear, the conditional probability density functions of the state are no longer Gaussian, even if the process and observation noise are Gaussian. For that reason, the probability distributions in question are no longer fully characterized by their mean and covariance, which are the only moments of the distributions in question that are propagated in the classical Kalman filter formulation. In this paper, to design a decentralized estimation solution, the system is linearized and the proposed solution is a fully decentralized variation of the extended Kalman filter, which approximates the probability distributions as Gaussian distributions.

2.3. Communication, computational, and memory constraints

It is of the utmost importance to remark that the solution devised for (8) must be feasible to implement in real-time and in a decentralized configuration. In particular, the procedure to compute each gain $\mathbf{K}_i(k)$ must follow several constraints required for a decentralized implementation, regarding communication, computational, and memory requirements. In this section, these constraints are presented with detail, arriving at a set of requirements for the decentralized estimation solution.

First, there are various communication requirements associated with each step of the decentralized filter. To compute the update step of (5) and to compute each local filter gain $\mathbf{K}_i(k)$, communication between systems is required. On one hand, a very important aspect to take into account is the synchronization of the data transmissions. There are variables, such as the updated state estimates $\hat{\mathbf{x}}_i(k-1|k-1)$ and the predicted state estimates $\hat{\mathbf{x}}_i(k|k-1)$, which are stored in system $\mathcal{S}_i$, that can be shared with another system $\mathcal{S}_j$ at any time since time instant $k-1$ until time instant k, because they are known to $\mathcal{S}_i$ at time instant $k-1$ and are only required by $\mathcal{S}_j$ at time instant k. For these transmissions, denoted as soft real-time transmissions, no complex synchronization protocols are required. However, for a system $\mathcal{S}_j$ to use the output of system $\mathcal{S}_i$, it would have to be transmitted instantaneously at time instant k, because the update step in $\mathcal{S}_j$ is carried out at time instant k and the output is only available in system $\mathcal{S}_i$ at time instant k. These are denoted as hard real-time transmissions, for which very complex synchronization algorithms are required, and which become unfeasible to implement in practice in large-scale networks. For this reason, in this paper, hard real-time transmissions are not allowed. Instead, a fully decentralized configuration is adopted, i.e., the update step of system $\mathcal{S}_i$ can only make use of the output $\mathbf{y}_i(k)$.

Second, the number of available communication links between systems is limited. Although there are decentralized solutions in the literature that require all-to-all communication (see Section 1), these are not scalable to very large-scale networks. Thus, to seek a scalable solution, the number of communications links established with each subsystem must not increase with an increase of the number of subsystems in the network, i.e., the communication complexity of each system ought to grow with $\mathcal{O}(1)$ with the dimension of the network. It is very important to point out that the aforementioned communication link limitations is addressed at the protocol level, i.e., the restrictions are applied to the exchange of data between systems, not on the physical communication links. In particular, system $\mathcal{S}_i$ is not allowed to access data from $\mathcal{S}_l$ via a path of systems through which the information could be retransmitted. Such a configuration, would, for large-scale networks, increase communication delays, decrease the robustness of the solution, result in an uneven distribution of communication coordination burden, and increase the complexity of the communication protocol.

Third, the amount of data transmitted per communication of a system must not scale with N, i.e., the transmission data size complexity ought to grow with $\mathcal{O}(1)$ with the dimension of the network.

Fourth, the memory of each computational unit $\mathcal{T}_i$ is limited. Although there are decentralized solutions in the literature that make use of extensive bookkeeping of outputs, it becomes unfeasible to implement as the dimension of the network increases. For that reason, the amount of data to be stored in each computational unit $\mathcal{T}_i$ must not depend on the dimension of the network. That is, the data storage complexity of each system ought to grow with $\mathcal{O}(1)$ with N.

Fifth, the time and computational resources available to computational unit $\mathcal{T}_i$ to implement the local filter are limited. The computational load of the global estimation algorithm of the network must be distributed across all computational units in such a way that each carries out computations concerning their own state estimate exclusively, which circumvents the curse of dimensionality. For that reason, the computational complexity of the floating point operations carried out by each computational unit must grow with $\mathcal{O}(1)$ with the dimension of the network.

2.3.1. Summed-up constraints

As a means of proposing a solution suitable to large-scale systems, several constraints are required on the complexity of the communication between systems and on the computations performed in each system. To sum-up, the decentralized estimation solution should follow:

1. No hard real-time transmissions are allowed;
2. The communication complexity of each system ought to be $\mathcal{O}(1)$;
3. The transmission data size complexity of each system ought to be $\mathcal{O}(1)$;
4. The data storage complexity of each system ought to be $\mathcal{O}(1)$;
5. The computational complexity of each system ought to be $\mathcal{O}(1)$.

3. Decentralized extended Kalman filter

The goal is to design a decentralized estimation solution to solve the optimization problem (8) subject to the communication, memory, and computational constraints 1–5 put forward in Section 2.3.1, which are imposed by the decentralized configuration for very large-scale systems. The derivation of the one-step decentralized extended Kalman filter follows three steps: (i) system linearization and covariance propagation, inspired by the extended Kalman filter; (ii) convex relaxation of the optimization problem of the estimation solution; and (iii) decoupling of the real-time gain synthesis procedure for application in a large-scale network. This solution is presented in a first instance for a time-invariant network topology and, in Section 3.5, it is extended for a time-varying topology.

3.1. Linearization

First, to address the nonlinearity of the network dynamics, the system is linearized and the proposed solution is a fully decentralized variation of the well-known extended Kalman filter. Defining $\mathbf{P}(k+1|k) \succeq \mathbf{0}$ as the covariance of the estimation error at instant $k+1$ after the prediction step of the global filter (6), one can write

$$\mathbf{P}(k+1|k) = \mathbf{A}(k)\mathbf{P}(k|k)\mathbf{A}^T(k) + \mathbf{Q}\left(k, \hat{\mathbf{x}}(k|k), \mathbf{u}(k)\right), \tag{9}$$

and

$$\begin{aligned}\mathbf{P}(k|k) = {} & \mathbf{K}(k)\mathbf{R}\left(k, \hat{\mathbf{x}}(k|k-1)\right)\mathbf{K}^T(k) \\ & + (\mathbf{I} - \mathbf{K}(k)\mathbf{C}(k))\mathbf{P}(k|k-1)(\mathbf{I} - \mathbf{K}(k)\mathbf{C}(k))^T,\end{aligned} \tag{10}$$

inspired by the extended Kalman filter. In these expressions, matrices $\mathbf{A}(k)$ and $\mathbf{C}(k)$ correspond to the linearization of the nonlinear global system dynamics and output equation, respectively, about the updated state estimate and predicted state estimate, respectively, i.e.,

$$\mathbf{A}(k) = \left.\frac{\partial \mathbf{f}(k, \mathbf{x}, \mathbf{u})}{\partial \mathbf{x}}\right|_{\substack{\mathbf{x}=\hat{\mathbf{x}}(k|k) \\ \mathbf{u}=\mathbf{u}(k)}} \text{ and } \mathbf{C}(k) = \left.\frac{\partial \mathbf{g}(k, \mathbf{x})}{\partial \mathbf{x}}\right|_{\mathbf{x}=\hat{\mathbf{x}}(k|k-1)}.$$

Before proceeding with the solution of the relaxed optimization problem, it is important to introduce the block decomposition of the matrices relevant for the gain computation and posterior decoupling. Let $\mathbf{A}_{(i,j)}(k) \in \mathbb{R}^{n_i \times n_j}$, $\mathbf{C}_{(i,j)}(k) \in \mathbb{R}^{o_i \times n_j}$, $\mathbf{Q}_{(i,j)}(k) \in \mathbb{R}^{n_i \times n_j}$, $\mathbf{R}_{(i,j)}(k) \in \mathbb{R}^{o_i \times o_j}$, $\mathbf{P}_{(i,j)}(k|k) \in \mathbb{R}^{n_i \times n_j}$, and $\mathbf{P}_{(i,j)}(k+1|k) \in \mathbb{R}^{n_i \times n_j}$ denote the blocks of indices (i,j) of the block decomposition of matrices $\mathbf{A}(k)$, $\mathbf{C}(k)$, $\mathbf{Q}\left(k, \hat{\mathbf{x}}(k|k), \mathbf{u}(k)\right)$, $\mathbf{R}\left(k, \hat{\mathbf{x}}(k|k-1)\right)$, $\mathbf{P}(k|k)$, and $\mathbf{P}(k+1|k)$, respectively. Note that expressions for $\mathbf{A}_{(i,j)}(k)$, $\mathbf{C}_{(i,j)}(k)$, $\mathbf{Q}_{(i,j)}(k)$, and $\mathbf{R}_{(i,j)}(k)$ can be easily obtained from the definition of the local dynamics presented in Section 2.1.

3.2. Convex relaxation

As aforementioned, the optimization problem (8) is nonconvex. Thus, to use standard optimization techniques, convex relaxation is performed. An approach similar to the one used for the unconstrained Kalman filter is used. The global gain in each instant is computed so that the trace of the covariance of the estimation error for that same

instant is minimized. The optimization problem (8) is, thus, modified to

$$\begin{aligned} \underset{\mathbf{K}(k)\in\mathbb{R}^{n\times o}}{\text{minimize}} \quad & \operatorname{tr}(\mathbf{P}(k|k)) \\ \text{subject to} \quad & \mathbf{K}(k) \in \operatorname{Sparse}(\operatorname{diag}(\mathbf{1}_{n_1\times o_1},\dots,\mathbf{1}_{n_N\times o_N})), \end{aligned} \tag{11}$$

for $k \in \mathbb{N}$, given the predicted estimation error covariance, $\mathbf{P}(k|k-1)$, at each time step. The relaxed optimization problem (11) is convex, allowing for the use of techniques similar to those used to solve the unconstrained problem, which yield a closed-form solution, as detailed in the following result.

Lemma 1 (*One-step Gain*)**.** *Define* $\mathbf{S}(k)$ *as the global innovation covariance at step* k*, given by*

$$\mathbf{S}(k) := \mathbf{C}(k)\mathbf{P}(k|k-1)\mathbf{C}^T(k) + \mathbf{R}(k)\,, \tag{12}$$

and $\mathbf{S}_{(i,j)}(k) \in \mathbb{R}^{o_i\times o_j}$ *as the block of indices* (i,j) *of its block decomposition. Then, the solution to the relaxed one-step optimization problem* (11) *is given by*

$$\mathbf{K}_i(k) = \sum_{p\in\mathcal{D}_i^-} \left(\mathbf{P}_{(i,p)}(k|k-1)\mathbf{C}^T_{(i,p)}(k)\right)\mathbf{S}^{-1}_{(i,i)}(k)\,, \tag{13}$$

$i = 1,\dots,N$*, where*

$$\mathbf{S}_{(i,i)}(k) = \mathbf{R}_{(i,i)}(k) + \sum_{p\in\mathcal{D}_i^-}\sum_{q\in\mathcal{D}_i^-} \mathbf{C}_{(i,p)}(k)\mathbf{P}_{(p,q)}(k|k-1)\mathbf{C}^T_{(i,q)}(k) \tag{14}$$

and

$$\mathbf{P}_{(p,q)}(k|k-1) = \mathbf{Q}_{(p,q)}(k-1) + \mathbf{A}_{(p,p)}(k-1)\mathbf{P}_{(p,q)}(k-1|k-1)\mathbf{A}^T_{(q,q)}(k-1). \tag{15}$$

Proof. See Appendix. □

3.3. Main result

Note that, although (13) provides an expression to compute the gain of each local filter, it cannot be computed without full knowledge of the estimation error covariance of the whole network, because of the computation of $\mathbf{P}(k|k)$. For this reason, this result is not suitable for application in a decentralized configuration. In this section, under reasonable approximations, the filter gain computation is decoupled and an algorithm is put forward to allow for its distributed computation across the computational units of each system such that the communication, memory, and computational constraints presented in Section 2.3.1 are met.

Approximation 1. *Consider the estimation error covariance after the update step of the form* $\mathbf{P}_{(p,q)}(k+1|k+1)$*, with* $p \in \mathcal{D}_i^-$ *and* $q \in \mathcal{D}_i^-$ *for some* i*, and the estimation error covariance after the prediction step of the form* $\mathbf{P}_{(r,s)}(k+1|k)$*, with* $r \in \mathcal{D}_p^-$ *and* $s \in \mathcal{D}_q^-$*. In the decentralized algorithm put forward in this paper,* $\mathbf{P}_{(r,s)}(k+1|k)$ *is considered to be null in the computation of* $\mathbf{P}_{(p,q)}(k+1|k+1)$ *in computational unit* $\mathcal{T}_i$ *if* $(r,s) \notin \psi_i$*, where*

$$\psi_i = \bigcup_{j\in\mathcal{D}_i^-} \phi_j \tag{16}$$

with

$$\phi_i := \mathcal{D}_i^- \times \mathcal{D}_i^- = \{(p,q) \in \mathbb{N}^2 : p \in \mathcal{D}_i^- \wedge q \in \mathcal{D}_i^-\}\,. \tag{17}$$

The main result of this paper is supported by Approximation 1. Note that this approximation makes sense in the context of estimation. Consider Fig. 1, which represents the topology of Approximation 1 in a graph. It is expected that the influence of $\mathbf{P}_{(r,s)}(k+1|k)$ is more dominant in the computation of $\mathbf{K}_i(k)$ if both the states of $\mathcal{S}_r$ and $\mathcal{S}_s$ are coupled with the output of a system $\mathcal{S}_l$ that is coupled with the output of $\mathcal{S}_i$. Having this in mind, to decouple the gain synthesis of each filter, each computational unit $\mathcal{T}_i$ keeps and updates each estimation error covariance matrix of the form $\mathbf{P}_{(p,q)}(k|k), (p,q) \in \phi_i$. Henceforth, matrix

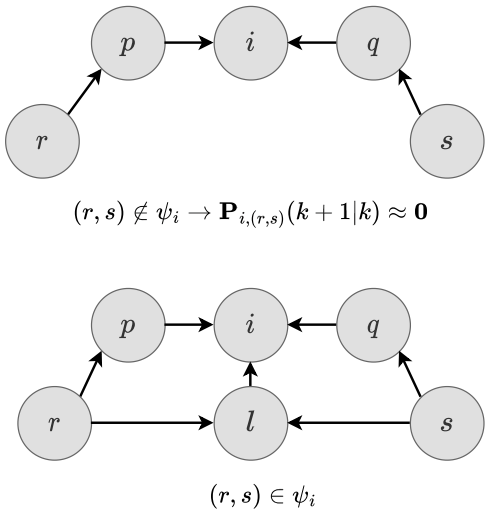


Fig. 1. Graphic illustration of Approximation 1.

$\mathbf{P}_{(p,q)}(k|k)$, stored and updated in system $\mathcal{S}_i$, is denoted as $\mathbf{P}_{i,(p,q)}(k|k)$. Performing the decomposition in block matrices of (10), one obtains

$$\begin{aligned} \mathbf{P}_{i,(p,q)}(k|k) &= \mathbf{K}_p(k)\mathbf{R}_{(p,q)}(k)\mathbf{K}_q^T(k) \\ &+ \sum_{r\in\mathcal{D}_p^-}\sum_{s\in\mathcal{D}_q^-} (\delta_{pr}\mathbf{I} - \mathbf{K}_p(k)\mathbf{C}_{(p,r)}(k))\mathbf{P}_{\mathcal{D}_i^-,(r,s)}(k|k-1)(\delta_{qs}\mathbf{I} - \mathbf{K}_q(k)\mathbf{C}_{(q,s)}(k))^T, \end{aligned} \tag{18}$$

which, making use of Approximation 1, becomes

$$\begin{aligned} \mathbf{P}_{i,(p,q)}(k|k) &= \mathbf{K}_p(k)\mathbf{R}_{(p,q)}(k)\mathbf{K}_q^T(k) \\ &+ \sum_{r\in\mathcal{D}_p^-}\sum_{\substack{s\in\mathcal{D}_q^- \\ (r,s)\in\psi_i}} (\delta_{pr}\mathbf{I} - \mathbf{K}_p(k)\mathbf{C}_{(p,r)}(k))\mathbf{P}_{\mathcal{D}_i^-,(r,s)}(k|k-1)(\delta_{qs}\mathbf{I} - \mathbf{K}_q(k)\mathbf{C}_{(q,s)}(k))^T, \end{aligned} \tag{19}$$

where the subscript $\mathcal{D}_i^-$ in $\mathbf{P}_{\mathcal{D}_i^-,(r,s)}(k|k-1)$ indicates, by abuse of notation, that $\mathbf{P}_{\mathcal{D}_i^-,(r,s)}(k|k-1)$ is computed in $\mathcal{T}_l$ with $l \in \mathcal{D}_i^-$, and δ_{pr} denotes the Kronecker delta.

Note that with Approximation 1, for system $\mathcal{S}_i$, the covariance propagation (19) can be computed in a distributed manner. It is important to remark that $\mathbf{P}_{\mathcal{D}_i^-,(r,s)}(k|k-1)$, inside the summation in (19), is not necessarily updated in $\mathcal{T}_i$, since only estimation error covariance matrices of the form $\mathbf{P}_{i,(p,q)}(k|k)$, with $(p,q) \in \phi_i$, are updated. Therefore, $\mathcal{S}_i$ has to receive $\mathbf{P}_{(r,s)}(k|k-1)$ through communication from a system $\mathcal{S}_l$, with $l \in \mathcal{D}_i^-$.

Lemma 2. *The approximation induced by Approximation 1 is exact if the topology of the network follows*

$$\forall i \in \{1,\dots,N\} \quad \forall p,q \in \mathcal{D}_i^- \quad \mathcal{D}_p^- \times \mathcal{D}_q^- \subseteq \bigcup_{j\in\mathcal{D}_i^-} \mathcal{D}_j^- \times \mathcal{D}_j^-.$$

Proof. Consider computational unit $\mathcal{T}_i$. Note that $\cup_{j\in\mathcal{D}_i^-}\mathcal{D}_j^- \times \mathcal{D}_j^- = \cup_{j\in\mathcal{D}_i^-}\phi_j = \psi_i$, i.e., the set of systems pairs whose covariance is known by the in-neighborhood of $\mathcal{T}_i$. According to (18) the exact covariance propagation in $\mathcal{T}_i$ requires that, for every $p,q \in \mathcal{D}_i^-$, the covariance between every $r \in \mathcal{D}_p^-$ and $s \in \mathcal{D}_q^-$ is accessible to $\mathcal{T}_i$. That is, the covariance matrices between the pairs in $\mathcal{D}_p^- \times \mathcal{D}_q^-$. Thus, if these are contained in ψ_i for every system $\mathcal{S}_i$, then the covariance propagation can be computed exactly. □

According to Lemma 2, the approximation induced by Approximation 1, which is used in the main result of these paper, is exact for some network topologies. In these cases, it is possible to distribute the computations of the globally synthesized filter across the computational units without disregarding any estimation error covariance

Algorithm 1 One-step decentralized extended Kalman filter algorithm at time instant k in computational unit $\mathcal{T}_i$ for time-invariant coupling topology.

Input: $\hat{\mathbf{x}}_i(k|k)$; $\mathbf{u}_i(k)$; $\mathbf{K}_i(k)$; $\mathbf{P}_{i,(p,q)}(k|k-1), \forall (p,q) \in \phi_i$; $\mathbf{R}_{(i,p)}(k), \forall p \in \mathcal{D}_i^-$; $\mathbf{C}_{(i,p)}(k), \forall p \in \mathcal{D}_i^-$
Output: $\hat{\mathbf{x}}_i(k+1|k+1)$; $\mathbf{K}_i(k+1)$; $\mathbf{P}_{i,(p,q)}(k+1|k), \forall (p,q) \in \phi_i$; $\mathbf{R}_{(i,p)}(k+1), \forall p \in \mathcal{D}_i^-$; $\mathbf{C}_{(i,p)}(k+1), \forall p \in \mathcal{D}_i^-$
Step 1: $\hat{\mathbf{x}}_i(k+1|k) \leftarrow \mathbf{f_i}(k, \hat{\mathbf{x}}_i(k|k), \mathbf{u}_i(k))$
$\mathbf{A}_{(i,i)}(k) \leftarrow \partial \mathbf{f_i}(k, \mathbf{x}_i(k), \mathbf{u}_i(k))/\partial \mathbf{x}_i(k)\Big|_{\mathbf{x}_i(k)=\hat{\mathbf{x}}_i(k|k)}$
$\mathbf{Q}_{(i,i)}(k) \leftarrow \mathbf{Q}_i(k, \hat{\mathbf{x}}_i(k|k), \mathbf{u}_i(k))$
Step 2: **Transmit**: $\hat{\mathbf{x}}_i(k+1|k)$; $\mathbf{A}_{(i,i)}(k)$; $\mathbf{Q}_{(i,i)}(k)$; $\mathbf{K}_i(k)$; $\mathbf{R}_{(i,j)}(k), \forall j \in \mathcal{D}_i^-$;
$\mathbf{C}_{(i,p)}(k), \forall p \in \mathcal{D}_i^-$; and $\mathbf{P}_{i,(p,q)}(k|k-1)$, $\forall (p,q) \in \phi_i$; to $r \in \mathcal{D}_i^+ \setminus \{i\}$
Step 3: **Receive**: $\hat{\mathbf{x}}_p(k+1|k), \mathbf{A}_{(p,p)}(k), \mathbf{Q}_{(p,p)}(k)$, and $\mathbf{K}_p(k)$ from all $p \in \mathcal{D}_i^- \setminus \{i\}$
$\mathbf{R}_{(p,q)}(k)$, $\forall (p,q) \in \phi_i \setminus \{(i,i)\}$, from either p or q
$\mathbf{C}_{(p,r)}(k)$, $\forall r \in \mathcal{D}_p^-$, from $p \in \mathcal{D}_i^- \setminus \{i\}$
$\mathbf{P}_{p,(r,s)}(k|k-1)$, for some $(r,s) \in \phi_p$ from $p \in \mathcal{D}_i^- \setminus \{i\}$
Step 4: $\mathbf{P}_{i,(p,q)}(k|k) \leftarrow \mathbf{K}_p(k)\mathbf{R}_{(p,q)}(k)\mathbf{K}_q^T(k) +$
$\sum_{r\in\mathcal{D}_p^-} \sum_{\substack{s\in\mathcal{D}_q^- \\ (r,s)\in\psi_i}} (\delta_{pr}\mathbf{I} - \mathbf{K}_p(k)\mathbf{C}_{(p,r)}(k))\mathbf{P}_{\mathcal{D}_i^-,(r,s)}(k|k-1)(\delta_{qs}\mathbf{I} - \mathbf{K}_q(k)\mathbf{C}_{(q,s)}(k))^T$, $\forall (p,q) \in \phi_i$
Step 5: $\mathbf{C}_{(i,j)}(k+1) \leftarrow \partial \mathbf{g_i}(k+1; \mathbf{x}_j(k+1), j \in \mathcal{D}_i^-)/\partial \mathbf{x}_j(k+1)\Big|_{\mathbf{x}_j(k+1)=\hat{\mathbf{x}}_j(k+1|k), j\in\mathcal{D}_i^-}$, $\forall j \in \mathcal{D}_i^-$
$\mathbf{R}_{(i,j)}(k+1) \leftarrow \mathbf{R}_{i,j}(k, \hat{\mathbf{x}}_i(k+1|k), \hat{\mathbf{x}}_j(k+1|k))$, $\forall j \in \mathcal{D}_i^-$
Step 6: $\mathbf{P}_{i,(p,q)}(k+1|k) \leftarrow \mathbf{A}_{(p,p)}(k)\mathbf{P}_{i,(p,q)}(k|k)\mathbf{A}_{(q,q)}^T(k) + \mathbf{Q}_{(p,q)}(k)$, $\forall (p,q) \in \phi_i$
$\mathbf{S}_{(i,i)}(k+1) \leftarrow \sum_{p\in\mathcal{D}_i^-} \sum_{q\in\mathcal{D}_i^-} \mathbf{C}_{(i,p)}(k+1)\mathbf{P}_{i,(p,q)}(k+1|k)\mathbf{C}_{(i,q)}^T(k+1) + \mathbf{R}_{(i,i)}(k+1)$
$\mathbf{K}_i(k+1) \leftarrow \sum_{p\in\mathcal{D}_i^-} \left(\mathbf{P}_{(i,p)}(k+1|k)\mathbf{C}_{(i,p)}^T(k+1)\right) \mathbf{S}_{(i,i)}^{-1}(k+1)$
Step 7: **Take measurement**: $\mathbf{y}_i(k+1)$
Step 8: $\hat{\mathbf{x}}_i(k+1|k+1) \leftarrow \hat{\mathbf{x}}_i(k+1|k) + \mathbf{K}_i(k+1)\left(\mathbf{y}_i(k+1) - \mathbf{g_i}(k+1; \hat{\mathbf{x}}_j(k+1|k), j \in \mathcal{D}_i^-)\right)$

components. Examples of these topologies are string, tree, and ring configurations. After introducing Approximation 1, which allows for the decoupling of the gain synthesis procedure, it is possible to state the proposed decentralized extended Kalman filter algorithm, as presented in the following result.

Theorem 1 (*One-step Decentralized Extended Kalman Filter*)**.** *The solution of optimization problem* (11) *for the linearized filter under Approximation* 1 *and subject to the communication, computational, and memory constraints detailed in Section* 2.3 *is given, for a time-invariant coupling topology, by the local filter iterations presented in Algorithm 1 (See Remarks* 1–5 *for some details regarding its implementation).*

Proof. The local gain computation is performed according to Lemma 1. Under Approximation 1, the covariance propagation is computed according to (19). The variables required by the gain computation and covariance propagation that are unknown to system $\mathcal{S}_i$ can obtained via local communication. Finally, for a detailed analysis of the communication, computational, and memory requirements of this algorithm, see Section 3.4. □

Corollary 1.1. For network topologies under the conditions of *Lemma* 2, the estimation error covariance propagation of the distributed decentralized EKF in Algorithm 1 is consistent.

Remark 1. For the initialization of Algorithm 1 in $\mathcal{S}_i$, i.e., for $k = 0$, one has to (i) initialize $\hat{\mathbf{x}}_i(k|k)$ with an initial estimate for $\mathbf{x}_i$; (ii) initialize $\mathbf{P}_{i,(p,q)}(0|0), \forall (p,q) \in \phi_i$; (iii) perform all steps except for Step 4; and (iv) in Step 3, receive only the variables required for Step 6.

Remark 2. Note that, in Algorithm 1, the data transmission and reception procedure is described generically. In fact, only a fraction of the data in the memory of a computational unit $\mathcal{T}_i$ has to be transmitted to a neighboring computational unit $\mathcal{T}_r$, with $r \in \mathcal{D}_i^+$. As an example, not all $\mathbf{P}_{i,(p,q)}(k|k-1)$, $\forall (p,q) \in \phi_i$ are required for the computations performed in $\mathcal{T}_r$. Therefore, the data transmission protocol should be adapted to each application, depending on the network topology, to keep data transmissions to a minimum.

Remark 3. For the computation in Step 4 performed in computational unit $\mathcal{T}_i$, there are estimation error covariance matrices $\mathbf{P}_{\mathcal{D}_i^-,(r,s)}(k|k-1)$ which have to be retrieved either from memory if $(r,s) \in \phi_i$ or from another computational unit $\mathcal{T}_j$, with $j \in \mathcal{D}_i^-$, in Step 3. Oftentimes, $\mathbf{P}_{\mathcal{D}_i^-,(r,s)}(k|k-1)$ is available from more than one source, so a selection rule should be put in place. There are several alternatives for this rule, which depend significantly on the application in question. In what follows a few examples are given. First, one can reduce the communication cost by prioritizing content in the memory of $\mathcal{T}_i$ and, if it is not available, retrieve it from $\mathcal{T}_j$ with which communication is cheaper. Second, one can associate, in each computational unit $\mathcal{T}_i$, a scalar loss $L_{i,(p,q)} \in \mathbb{R}$ to the computation of $\mathbf{P}_{i,(p,q)}(k|k-1)$, $\forall (p,q) \in \phi_i$ that is an empirical metric of its accuracy. An example of an empirical metric is the number of times that Approximation 1 was used in the computation of $\mathbf{P}_{i,(p,q)}(k-1|k-1)$, i.e., the number of terms of the summation in (18) that were disregarded, which is given by

$$L_{i,(p,q)} = \left| \left(\mathcal{D}_p^- \times \mathcal{D}_q^-\right) \setminus \psi_i \right|. \tag{20}$$

On one hand, for a time-invariant topology, the empirical losses of this example are constant in time and, thus, a computational unit may retrieve only $\mathbf{P}_{\mathcal{D}_i^-,(r,s)}(k|k-1)$ from the unit $\mathcal{T}_l$ with $l \in \mathcal{D}_i^-$ which is known to have the lowest $L_{j,(r,s)}$ $\forall j \in \mathcal{D}_i^-$, thereby reducing communication pressure. On the other hand, if the topology is time-varying, this metric has to be updated at each time instant in Step 4 of Algorithm 1. In general, in Step 2 all the available $\mathbf{P}_{\mathcal{D}_i^-,(r,s)}(k|k-1)$ are retrieved with their respective losses, and either their weighted average is considered or the average of the ones with the minimum loss.

Remark 4. Note that the computations performed in Step 4 of Algorithm 1 do not guarantee the positive definiteness of the resulting matrices. In applications with low process noise covariance or few communications links, it may lead to numerical problems. To prevent it, the computations in these steps should be written in matrix form as follows. Let $\mathcal{D}_p^- = \{r_1^p, \ldots, r_{|\mathcal{D}_p^-|}^p\}$, and consider $\tilde{\mathbf{W}}_p(k)$ of dimension $n_p \times \sum_{r\in\mathcal{D}_p^-} n_r$ and $\tilde{\mathbf{P}}_{\mathcal{D}_i^-}^{(p,q)}(k|k-1)$ of dimension $\sum_{r\in\mathcal{D}_p^-} n_r \times \sum_{s\in\mathcal{D}_q^-} n_s$, which are defined below. Consider the decomposition of these matrices in blocks of indices (i,j) denoted by $\left[\tilde{\mathbf{W}}_p(k)\right]_{(1,j)} \in \mathbb{R}^{n_p \times n_{r_j^p}}$ and

$\left[\tilde{\mathbf{P}}^{(p,q)}_{\mathcal{D}_i^-}(k|k-1)\right]_{(i,j)} \in \mathbb{R}^{n_{r_i^p} \times n_{r_j^q}}$. If one defines

$\left[\tilde{\mathbf{W}}_p(k)\right]_{(1,j)} := \delta_{pr_j^p}\mathbf{I} - \mathbf{K}_p(k)\mathbf{C}_{(p,r_j^p)}(k)$

and

$$\left[\tilde{\mathbf{P}}^{(p,q)}_{\mathcal{D}_i^-}(k|k-1)\right]_{(i,j)} := \begin{cases} \mathbf{P}_{\mathcal{D}_i^-,(r_i^p,r_j^q)}(k|k-1), & (r_i^p, r_j^q) \in \psi_i \\ \mathbf{0}, & (r_i^p, r_j^q) \notin \psi_i, \end{cases}$$

then (19) may be rewritten as

$$\mathbf{P}_{i,(p,q)}(k|k) = \mathbf{K}_p(k)\mathbf{R}_{(p,q)}(k)\mathbf{K}_q^T(k)\tilde{\mathbf{W}}_p(k)\tilde{\mathbf{P}}^{(p,q)}_{\mathcal{D}_i^-}(k|k-1)\tilde{\mathbf{W}}_q^T(k). \tag{21}$$

The covariance update step in matrix form in (21) is more robust to numerical error. Additionally, in networks with few communication links, numerical problems can be prevented by ensuring the positive definiteness of $\tilde{\mathbf{P}}^{(p,q)}_{\mathcal{D}_i^-}(k|k-1)$ with $p = q$ before performing the computation in (21). This should be done by: (i) performing an eigen-decomposition of $\tilde{\mathbf{P}}^{(p,q)}_{\mathcal{D}_i^-}(k|k-1)$; (ii) forcing the negative eigenvalues to a positive value, for instance the lowest positive eigenvalue; and (iii) recomputing $\tilde{\mathbf{P}}^{(p,q)}_{\mathcal{D}_i^-}(k|k-1)$ from the inverse of the decomposition. This procedure maintains the directions of the estimation error encoded in the covariance matrix. In Step 6 of Algorithm 1 the computation may be rewritten similarly to (21) to improve the robustness to numerical errors.

Remark 5. It is clear that, since there are no dynamical couplings between systems in the network considered in this problem, not only the predicted estimates $\hat{\mathbf{x}}_i(k|k)$, but also the prediction estimation error covariance matrices $\mathbf{P}_{i,(p,q)}(k+1|k), \forall (p,q) \in \phi_i$, can be computed in a decoupled fashion. This characteristic can be seen in Algorithm 1. This way, without requiring any additional communication and maintaining the computational complexity of the algorithm, the prediction step of the filter can be undergone making use of the Unscented Transformation, which is the backbone of the Unscented Kalman Filter proposed by Julier and Uhlmann (Julier & Uhlmann, 1997a, 2004). The use of this transformation, instead of the linearized dynamics, in the computation of $\hat{\mathbf{x}}_i(k|k)$ and $\mathbf{P}_{i,(p,q)}(k+1|k), \forall (p,q) \in \phi_i$ allows for the propagation of second-order terms of the probability distributions.

3.4. Communication, computational, and memory requirements

In this section, the communication, computational, and memory requirements of Algorithm 1 are analyzed and it is verified that the constraints 1–5 imposed in Section 2.3.1 are followed. To allow for a clearer analysis and lighter notation, in this section, the complexity is written for an homogeneous network, i.e., a network of identical systems. In particular, systems with the same order n_1, same output dimension o_1, and same number of output coupled systems ν_1^- are considered.

First, notice that the sequence of the computations in Algorithm 1 is such that a single instance of information exchange with other systems is required. Note that none of the variables exchanged in this step are required immediately after the transmission. In particular, each system $\mathcal{S}_i$ makes only use of their own output $\mathbf{y}_i(k+1)$ for the update step. Therefore, only soft real-time transmissions are required in this procedure.

Second, the communication directed graph $\mathcal{G}_c$, i.e., the directed graph representation of the available directed communication links, is restricted to the output coupling directed graph, i.e., $\mathcal{G}_c = \mathcal{G}$. In this configuration, system $\mathcal{S}_i$ can only receive information from system $\mathcal{S}_j$ if the output of $\mathcal{S}_i$ is coupled with the state of $\mathcal{S}_j$, i.e., $j \in \mathcal{D}_i^-$. Thus, system $\mathcal{S}_i$ requires the exchange of data through communication with $\nu_i^- - 1$ systems in each iteration of the filter. Thus, the communication complexity of system $\mathcal{S}_i$ is $\mathcal{O}(\nu_i^- - 1)$. Given that ν_i^-, the number of systems whose state is coupled with the output of $\mathcal{S}_i$, does not increase with the size of the network, then the communication complexity of each system grows with $\mathcal{O}(1)$ with the dimension of the network.

Third, note that, as far as the quantity of transferred data between systems is concerned, the most significant contribution for system $\mathcal{S}_i$ is the reception of the predicted covariance matrices $\mathbf{P}_{p,(r,s)}(k|k-1)$, $\forall (r,s) \in \phi_p \setminus \phi_i$ from $p \in \mathcal{D}_i^- \setminus \{i\}$. The number of floating point number to be received per iteration of the filter grows with $\mathcal{O}((\nu_1^-)^3 n_1^2)$. Given that neither ν_i^- nor n_i increase with an increase in the number of systems of the network, the transmission data size complexity of each system grows with $\mathcal{O}(1)$ with the dimension of the network.

Fourth, according to Algorithm 1, each system $\mathcal{S}_i$ has to store in memory: (i) the nonlinear functions $\mathbf{f_i} : \mathbb{R}^{n_i} \times \mathbb{R}^{m_i} \to \mathbb{R}^{n_i}$ and $\mathbf{g_i} : \prod_{j \in \mathcal{D}_i^-} \mathbb{R}^{n_j} \to \mathbb{R}^{o_i}$, that model the dynamics of the system and its output coupling with the other systems in its in-neighborhood; (ii) the process and observation noise covariance matrices $\mathbf{Q}_i(k, \mathbf{x}_i(k), \mathbf{u}_i(k)) \succeq \mathbf{0} \in \mathbb{R}^{n_i \times n_i}$ and $\mathbf{R}_{i,j}(k, \mathbf{x}_i(k), \mathbf{x}_j(k)) \succ \mathbf{0} \in \mathbb{R}^{o_i \times o_i}$, respectively; and (iii) $\mathbf{P}_{i,(p,q)}(k|k)$, with $(p,q) \in \phi_i$. Assuming that the nonlinear functions are stored efficiently, out of these, the most demanding is the latter, which requires storing a number of floating point numbers that grows with $\mathcal{O}((\nu_1^-)^2 n_1^2)$. Of course there are auxiliary variables which are required to be handled at each iteration, such as the covariance matrices received by communication, which require temporary memory storage. Thus, the data storage complexity of each system grows with $\mathcal{O}(1)$ with the dimension of the network, since the memory required does not scale with the number of systems in the network.

Fifth, as far as the computational complexity is concerned, the most intensive computations of Algorithm 1, assuming the evaluations of the nonlinear dynamics and output functions are not computationally demanding, are the propagation of the covariance matrices $\mathbf{P}_{i,(p,q)}(k|k)$, with $(p,q) \in \phi_i$, in Step 4. For each system and for each $(p,q) \in \phi_i$ the dominant computation is the double summation, which amounts to $\mathcal{O}((\nu_1^-)^2 n_1^2 \max(n_1, o_1))$ floating point operations. Therefore, the total number of floating point operations, in Step 4, for each system, grows with $\mathcal{O}((\nu_1^-)^4 n_1^2 \max(n_1, o_1))$. Given that neither ν_1^- nor $\max(n_1, o_1)$ increase with an increase in the number of systems of the network, it follows that the computational complexity of Algorithm 1 grows with $\mathcal{O}(1)$ with N. It is possible to conclude that the algorithm put forward in this paper follows the constraints presented in Section 2.3 to be suitable to be applied to very large-scale systems. It is important to note that, when designing an estimation solution for a large-scale solution, these aspects shall not be disregarded. For instance, in Pedroso et al. (2021) a state-of-the-art state estimation solution is developed for networks of systems with LTV dynamics, which can easily be adapted to an extended Kalman filter. However, using this method the linearization procedure requires all-to-all communication and the gain computation must be replicated in each system, which violates conditions 2 and 5 detailed in Section 2.3.1. For this reason, the method proposed in this paper is a significant step towards the application of decentralized state estimation solutions to large-scale nonlinear systems.

3.5. Extension to time-varying output coupling topologies

In this section, the algorithm proposed to implement a decentralized state estimation solution for a large-scale network of nonlinear systems is extended to allow for a time-varying output coupling topology. Oftentimes, the output couplings between systems vary with time due to: (i) the failure of systems of the network; (ii) the introduction of new systems in the network; or (iii) switching network configurations. To that purpose, consider, now, a time-varying directed graph $\mathcal{G}(k)$, with time-varying in-degree $\nu_i^-(k)$, in-neighborhood, $\mathcal{D}_i^-(k)$, and define $\phi_i(k)$ and $\psi_i(k)$ analogously to (17) and (16), respectively, for system $\mathcal{S}_i$. In this paper, the general case, in which the evolution of the output coupling topology with time is not known *a priori* is considered.

The extension of Algorithm 1 to a time-varying output coupling topology is given by Algorithm 2. First, notice that a similar outline is followed. Second, the local output coupling topology, defined by

Algorithm 2 One-step decentralized extended Kalman filter algorithm at time instant k in computational unit $\mathcal{T}_i$ for time-varying coupling topology.

Input: $\hat{\mathbf{x}}_i(k|k)$; $\mathbf{u}_i(k)$; $\mathcal{D}_i^-(k)$; $\mathbf{K}_i(k)$; $\mathbf{P}_{i,(p,q)}(k|k-1), \forall (p,q) \in \phi_i(k)$; $\mathbf{R}_{(i,p)}(k), \forall p \in \mathcal{D}_i^-(k)$; $\mathbf{C}_{(i,p)}(k), \forall p \in \mathcal{D}_i^-(k)$

Output: $\hat{\mathbf{x}}_i(k+1|k+1)$; $\mathbf{K}_i(k+1)$; $\mathcal{D}_i^-(k+1)$; $\mathbf{P}_{i,(p,q)}(k+1|k), \forall (p,q) \in \phi_i(k+1)$; $\mathbf{R}_{(i,p)}(k+1), \forall p \in \mathcal{D}_i^-(k+1)$; $\mathbf{C}_{(i,p)}(k+1), \forall p \in \mathcal{D}_i^-(k+1)$

Step 1: $\hat{\mathbf{x}}_i(k+1|k) \leftarrow \mathbf{f_i}(k, \hat{\mathbf{x}}_i(k|k), \mathbf{u}_i(k))$

$\mathbf{A}_{(i,i)}(k) \leftarrow \partial \mathbf{f_i}(k, \mathbf{x}_i(k), \mathbf{u}_i(k))/\partial \mathbf{x}_i(k)\big|_{\mathbf{x}_i(k)=\hat{\mathbf{x}}_i(k|k)}$

$\mathbf{Q}_{(i,i)}(k) \leftarrow \mathbf{Q}_i(k, \hat{\mathbf{x}}_i(k|k), \mathbf{u}_i(k))$

Step 2: **Define**: Predicted local output coupling topology, defined by $\hat{\mathcal{D}}_i^-(k+1)$ and $\hat{\mathcal{D}}_i^+(k+1)$

Step 3: **Transmit**: $\hat{\mathbf{x}}_i(k+1|k)$; $\mathbf{A}_{(i,i)}(k)$; $\mathbf{Q}_{(i,i)}(k)$; $\mathbf{K}_i(k)$; $\mathbf{R}_{(i,j)}(k), \forall j \in \mathcal{D}_i^-(k)$; $\mathbf{C}_{(i,p)}(k), \forall p \in \mathcal{D}_i^-(k)$; and $\mathbf{P}_{i,(p,q)}(k|k-1)$, $\forall (p,q) \in \phi_i(k)$; to $r \in \hat{\mathcal{D}}_i^+(k+1) \setminus \{i\}$.

Step 4: **Receive**: $\hat{\mathbf{x}}_p(k+1|k), \mathcal{D}_p^-(k), \mathbf{A}_{(p,p)}(k), \mathbf{Q}_{(p,p)}(k)$, and $\mathbf{K}_p(k)$ from all $p \in \hat{\mathcal{D}}_i^-(k+1) \setminus \{i\}$

$\mathbf{R}_{(p,q)}(k)$, $\forall (p,q) \in \hat{\phi}_i(k+1) \setminus \{(i,i)\}$, from either p or q

$\mathbf{C}_{(p,r)}(k)$, $\forall r \in \mathcal{D}_p^-(k)$ from $p \in \hat{\mathcal{D}}_i^-(k+1) \setminus \{i\}$

$\mathbf{P}_{p,(r,s)}(k|k-1)$, for some $(r,s) \in \phi_p(k)$ from $p \in \hat{\mathcal{D}}_i^-(k+1) \setminus \{i\}$

Step 5: **Define**: Local output coupling topology $\mathcal{D}_i^-(k+1)$ as the set of systems in $\hat{\mathcal{D}}_i^-(k+1)$ from which the required data was received.

Step 6: $\mathbf{P}_{i,(p,q)}(k|k) \leftarrow \mathbf{K}_p(k)\mathbf{R}_{(p,q)}(k)\mathbf{K}_q^T(k) + \sum_{r \in \mathcal{D}_p^-(k)} \sum_{\substack{s \in \mathcal{D}_q^-(k) \\ (r,s) \in \psi_i(k)}} (\delta_{pr}\mathbf{I} - \mathbf{K}_p(k)\mathbf{C}_{(p,r)}(k))\mathbf{P}_{i,(r,s)}(k|k-1)(\delta_{qs}\mathbf{I} - \mathbf{K}_q(k)\mathbf{C}_{(q,s)}(k))^T, \forall (p,q) \in \phi_i(k+1)$

Step 7: $\mathbf{C}_{(i,j)}(k+1) \leftarrow \partial \mathbf{g_i}(k; \mathbf{x}_j(k+1), j \in \mathcal{D}_i^-(k+1))/\partial \mathbf{x}_j(k+1)\big|_{\mathbf{x}_j(k+1)=\hat{\mathbf{x}}_j(k+1|k), j \in \mathcal{D}_i^-(k+1)}, \forall j \in \mathcal{D}_i^-(k+1)$

$\mathbf{R}_{(i,j)}(k+1) \leftarrow \mathbf{R}_{i,j}(k, \hat{\mathbf{x}}_i(k+1|k), \hat{\mathbf{x}}_j(k+1|k))$, $\forall j \in \mathcal{D}_i^-(k+1)$

Step 8: $\mathbf{P}_{i,(p,q)}(k+1|k) \leftarrow \mathbf{A}_{(p,p)}(k)\mathbf{P}_{i,(p,q)}(k|k)\mathbf{A}_{(q,q)}^T(k) + \mathbf{Q}_{(p,q)}(k)$, $\forall (p,q) \in \phi_i(k+1)$

$\mathbf{S}_{(i,i)}(k+1) \leftarrow \sum_{p \in \mathcal{D}_i^-(k+1)} \sum_{q \in \mathcal{D}_i^-(k+1)} \mathbf{C}_{(i,p)}(k+1)\mathbf{P}_{i,(p,q)}(k+1|k)\mathbf{C}_{(i,q)}^T(k+1) + \mathbf{R}_{(i,i)}(k+1)$

$\mathbf{K}_i(k+1) \leftarrow \sum_{p \in \mathcal{D}_i^-(k+1)} \left(\mathbf{P}_{(i,p)}(k+1|k)\mathbf{C}_{(i,p)}^T(k+1)\right) \mathbf{S}_{(i,i)}^{-1}(k+1)$

Step 9: **Take measurement**: $\mathbf{y}_i(k+1)$

Step 10: $\hat{\mathbf{x}}_i(k+1|k+1) \leftarrow \hat{\mathbf{x}}_i(k+1|k) + \mathbf{K}_i(k+1)\left(\mathbf{y}_i(k+1) - \mathbf{g_i}(k+1; \hat{\mathbf{x}}_j(k+1|k), j \in \mathcal{D}_i^-(k+1))\right)$

the in-neighborhood of each system $\mathcal{S}_i$, is defined in real-time. After computing its predicted state estimate, each system $\mathcal{S}_i$ computes a predicted coupling topology, defined by $\hat{\mathcal{D}}_i^-(k+1)$ and $\hat{\mathcal{D}}_i^+(k+1)$, possibly depending on the predicted state estimate. It, then, expects to receive data from each system $\mathcal{S}_p$, with $p \in \hat{\mathcal{D}}_i^-(k+1) \setminus \{i\}$, which is required to consider the effect of the output coupling of $\mathcal{S}_p$. Afterwards, the systems that transmit the required information define the actual local output coupling topology $\mathcal{D}_i^-(k+1)$. Note that the prediction of the local coupling topology, as well as the conditions for a system to respond to a request, vary greatly with the application in question. Third, it is important to remark that the sequence of steps on Algorithm 2 allows to compute the updated estimation error covariance matrix at time-instant k only between systems $\mathcal{S}_p$ and $\mathcal{S}_q$ whose state is both coupled with the output in time-instant $k+1$, i.e., $(p,q) \in \phi_i(k+1)$. This allows to lessen the communication burden, since the computations are efficiently distributed across all computational units.

4. Application to the onboard navigation problem of LEO mega-constellations using GNSS

In this section, the decentralized EKF algorithm developed in Section 3 is applied to the cooperative navigation of LEO mega-constellations. Note that, unlike the state-of-the-art decentralized estimation algorithms, this algorithm follows the communication, computational, and memory constraints presented in Section 2, which are necessary for an effective real-time decentralized implementation. Each satellite has access to inertial absolute position measurements provided by a GNSS system and also relative position measurements between satellites in close proximity, obtained with carrier-phase differential GNSS. All source code of the implementation of the proposed algorithm to this problem is available as an example of the DECENTER Toolbox at https://decenter2021.github.io/examples/DDEKFStarlinkConstellation/.

4.1. Mega-constellation model

Consider a constellation with a total of T satellites. The satellites are evenly distributed over P orbital planes at an inclination i, and with a relative phasing between adjacent planes of $\beta = 2\pi F/T$, where F is the phasing parameter. Such a configuration is designated as a Walker constellation and it is denoted by $i : T/P/F$. This constellation can be modeled as a network of interconnected systems, $\mathcal{S}_j$, each associated with a computational unit $\mathcal{T}_j$, with $j = 1, \ldots, T$. Let $\mathbf{x}_i^T := [\mathbf{p}_i^T \ \dot{\mathbf{p}}_i^T]$ denote the state of $\mathcal{S}_i$, where $\mathbf{p}_i \in \mathbb{R}^3$ and $\dot{\mathbf{p}}_i \in \mathbb{R}^3$ denote the position and velocity vectors, respectively, of $\mathcal{S}_i$. The dynamics of satellite of the constellation are modeled independently, since there is no dynamical coupling between them. All vectors are expressed in the J2000 Earth centered inertial (ECI) frame. The model of the translational dynamics used for the numerical simulations in Section 4.3 takes into account several perturbations. However, the model used for the propagation of the decentralized Kalman filter considers only J_2 and atmospheric drag perturbations. It is assumed that each satellite $\mathcal{S}_i$ is equipped with Hall effect thrusters that generate a force $\mathbf{u}_i \in \mathbb{R}^3$. The model of the dynamics of a single satellite $\mathcal{S}_i$ is, thus, given by

$$\begin{cases} \ddot{\mathbf{p}}_i = -\mu \mathbf{p}_i/\|\mathbf{p}_i\|_2^3 + \mathbf{a}_i^{J_2} + \mathbf{a}_i^D + \mathbf{u}_i/m_i \\ \dot{m}_i = -\|\mathbf{u}_i\|_1/(I_i^{sp} g_0), \end{cases} \tag{22}$$

where m_i denotes the mass of the satellite, μ denotes the gravitational parameter of the Earth, $\mathbf{a}_i^D \in \mathbb{R}^3$ and $\mathbf{a}_i^{J_2} \in \mathbb{R}^3$ denote the perturbation accelerations of J_2 and the atmospheric drag, respectively, I_i^{sp} denotes the specific impulse of the Hall thrusters of $\mathcal{S}_i$, and g_0 denotes the standard gravity acceleration. The J_2 perturbation acceleration can be computed as the symmetric of the gradient of the potential (Vallado, 1997, Chapter 7.6)

$$U_2(\mathbf{p}) = -\frac{\mu}{\|\mathbf{p}\|} J_2 \left(\frac{R_\oplus}{\|\mathbf{p}\|}\right)^2 \left(-\frac{1}{2} + \frac{3}{2}\left(\frac{[\mathbf{p}]_3}{\|\mathbf{p}\|}\right)^2\right),$$

i.e., $\mathbf{a}_i^{J_2} = -\nabla U_2(\mathbf{p}_i)$. The acceleration due to atmospheric drag of a generic satellite is given by Vallado (1997, Chapter 7.6)

$$\mathbf{a}^D = -\frac{1}{2}\frac{C_D A}{m}\rho\|\mathbf{v}_{rel}\|^2\frac{\mathbf{v}_{rel}}{\|\mathbf{v}_{rel}\|},$$

where C_D is the drag coefficient, A is the cross-sectional area, ρ is the air density, and $\mathbf{v}_{rel}$ denotes the velocity vector relative to the atmosphere, i.e.,

$$\mathbf{v}_{rel} = \dot{\mathbf{p}} - \omega_{\oplus}\mathbf{e}_3 \times \mathbf{p},$$

assuming a windless atmosphere, where $\omega_{\oplus}$ denotes the angular speed of the Earth and $\mathbf{e}_3^T := [0\ 0\ 1]$. The atmospheric density of the Earth is assumed to be modeled by a simple exponential evolution

$$\rho = \rho_0 \exp\left(-(\|\mathbf{p}\| - \mathbf{R}_{\oplus} - h_0)/H\right),$$

where the reference and scale altitude are denoted, respectively, by h_0 and H, which can be looked up in a table (see, for instance, Vallado (1997, Table 7.4)) depending on the nominal altitude of the constellation. To express the dynamics of each satellite $\mathcal{S}_i$ in discrete-time, consider a sampling time T_s and assume that the actuation $\mathbf{u}_i$ remains constant over each time interval. Then, one can write

$$\begin{cases} \mathbf{x}_i(k+1) = \mathbf{f}_{\mathbf{i}}(k, \mathbf{x}_i(k), \mathbf{u}_i(k), m_i(k)) + \mathbf{w}_i(k+1) \\ m_i(k+1) = m_i(k) - T_s\|\mathbf{u}_i(k)\|_1/(I_i^{sp} g_0), \end{cases} \tag{23}$$

where $\mathbf{f}_{\mathbf{i}}$ performs the integration of the aforementioned continuous-time dynamics models and $\mathbf{w}_i(k+1)$ is the process noise that accounts for accelerations due to other minor perturbations and modeling errors. The process noise is modeled by a zero-mean, white Gaussian process with associated covariance matrix $\mathbf{Q}_i(k, \mathbf{x}_i(k), \mathbf{u}_i(k))$ and $\mathbf{w}_p(k)$, and $\mathbf{w}_q(k)$ are assumed to be independent for $p \neq q$. Note that $\mathbf{f}_{\mathbf{i}}$ in (23) may depend on the time instant to account for possibly time-varying model parameters.

Each satellite is equipped with a GNSS receiver which is used to compute inertial position measurements in an Earth centered Earth fixed (ECEF) frame. After a coordinate transformation, the inertial measurement can be expressed in the ECI J2000 frame, which is a part of the output of each system $\mathcal{S}_i$. Then, considering that these measurements are available at a sampling period T_s, it is given by

$$\mathbf{y}_i^{in}(k) = \mathbf{x}_i(k) + \mathbf{v}_i^{in}(k)\,, \tag{24}$$

where $\mathbf{v}_i^{in}(k)$ is the sensor noise, which is modeled by a zero-mean, white Gaussian process, with associated covariance matrix $\mathbf{R}_i^{in}(k, \mathbf{x}_i(k))$, and $\mathbf{v}_p^{in}(k)$ and $\mathbf{v}_q^{in}(k)$ are assumed to be independent for $p \neq q$. Furthermore, it is considered that each satellite can communicate with neighboring satellites via an ISL. Such communication topology can be represented by a time-varying graph $\mathcal{G}(k)$, which features a node j for each satellite system $\mathcal{S}_j$, and a bidirectional edge for each ISL. It is considered that two satellite systems $\mathcal{S}_i$ and $\mathcal{S}_j$ communicate via a satellite link if they are in ISL range, i.e., $\|\mathbf{p}_i - \mathbf{p}_j\| < R_{ISL}$, where $R_{ISL} \in \mathbb{R}$ is the line-of-sight range for ISL transmissions. Making use of the ISL between two satellites and their GNSS receivers, it is possible to obtain a common relative position measurement between them with carrier-phase differential GNSS. Thus, for this application in particular, the output coupling graph could be treated as undirected. Single carrier and double carrier differences have been shown to allow for high precision relative measurements for short baselines (Allende-Alba, Montenbruck, Hackel, & Tossaint, 2018; Li, 2018; Montenbruck, Ebinuma, Lightsey, & Leung, 2002). The procedure for the determination of the relative measurements evolves resolving an integer ambiguity, as described, for instance, in Kroes, Montenbruck, Bertiger, and Visser (2005). It is also important to note that, for relative distances of the order of hundreds of kilometers, some nonlinear correction terms have to be considered, as described in Blewitt (1989), Tancredi et al. (2014), Wolfe et al. (2007) and Wu and Bar-Sever (2006). In this paper, provided that the main contribution is the decentralized EKF method, and for the sake of simplicity, the relative measurements are assumed to be given by a linear model with additive Gaussian noise. Nevertheless, note that, even for long baselines and real-time integer-ambiguities determination, the Gaussian assumption seems reasonable (Tancredi et al., 2014). Thus, the relative measurement output of satellite system $\mathcal{S}_i$ relative to $\mathcal{S}_j$, with $j \in \mathcal{D}_i^-$, is given by

$$\mathbf{y}_{i,j}^{rel}(k) = \mathbf{p}_i(k) - \mathbf{p}_j(k) + \mathbf{v}_{i,j}^{rel}(k), \tag{25}$$

where $\mathbf{v}_{i,j}^{rel}(k)$ is the relative measurement noise, which is modeled by a zero-mean, white Gaussian process characterized by: (i) $\mathrm{E}[\mathbf{v}_{i,j}^{rel}(k)\mathbf{v}_{i,j}^{rel\,T}(k)] = \mathbf{R}_{i,j}^{rel}(k, \mathbf{x}_i(k), \mathbf{x}_j(k))$; (ii) $\mathrm{E}[\mathbf{v}_{i,j}^{rel}(k)\mathbf{v}_{j,i}^{rel\,T}(k)] = -\mathbf{R}_{i,j}^{rel}(k, \mathbf{x}_i(k), \mathbf{x}_j(k))$, since $\mathbf{y}_{i,j}^{rel}(k) = -\mathbf{y}_{j,i}^{rel}(k)$; and (iii) $\mathrm{E}[\mathbf{v}_{i,j}^{rel}(k)\mathbf{v}_{p,q}^{rel\,T}(k)] = 0,\ \forall i, j, p, q : (i \neq p \vee j \neq q) \wedge (i \neq q \vee j \neq p)$. The local output dynamics of a satellite system $\mathcal{S}_i$, denoted by $\mathbf{y}_i(k)$, can be written as the concatenation of the inertial and relative outputs given, respectively, by (24) and (25). Let $\mathcal{D}_i^- \setminus \{i\} = \{j_1^i, \dots, j_{|\mathcal{D}_i^-|-1}^i\}$ and let $\mathbf{l}_j \in \mathbb{R}^{|\mathcal{D}_i^-|}$ denote a column vector whose entries are all set to zero except for the jth one, which is set to 1. Then $\mathbf{y}_i(k) := \mathrm{col}\left(\mathbf{y}_{i,j_1^i}^{rel}(k), \dots, \mathbf{y}_{i,j_{|\mathcal{D}_i^-|-1}^i}^{rel}(k), \mathbf{y}_i^{in}(k)\right)$ and

$$\mathbf{y}_i(k) = \sum_{p \in \mathcal{D}_i^-} \mathbf{C}_{(i,p)}(k)\mathbf{x}_p(k) + \mathbf{v}_i(k), \tag{26}$$

with

$$\mathbf{C}_{(i,p)} = \begin{cases} \left[\mathbf{1}_{|\mathcal{D}_i^-|\times 1}\ \ \mathbf{0}_{|\mathcal{D}_i^-|\times 1}\right] \otimes \mathbf{I}_3, & p = i \\ \left[-\mathbf{l}_l\ \ \mathbf{0}_{|\mathcal{D}_i^-|\times 1}\right] \otimes \mathbf{I}_3, & p = j_l^i \end{cases}$$

and the global measurement noise $\mathbf{v}(k) := \mathrm{col}(\mathbf{v}_1(k), \dots, \mathbf{v}_T(k))$ is a zero-mean white Gaussian process with associated covariance matrix $\mathbf{R}(k, \mathbf{x}_1(k), \dots, \mathbf{x}_T(k))$, defined, using the block decomposition presented in Section 3.1, by

$$\mathrm{E}[\mathbf{v}_i(k)\mathbf{v}_p^T(k)] = \mathbf{R}_{(i,p)}(k) = \begin{cases} \mathrm{diag}\left(\mathbf{R}_{i,j_1^i}^{rel}, \dots, \mathbf{R}_{i,j_{|\mathcal{D}_i^-|-1}^i}^{rel}, \mathbf{R}_i^{in}\right), & p = i \\ -\left(\mathbf{l}_r\mathbf{l}_s^T\right) \otimes \mathbf{R}_{i,p}^{rel}, & p = j_r^i \wedge i = j_s^p \\ \mathbf{0}_{|\mathcal{D}_i^-|\times|\mathcal{D}_p^-|}, & p \notin \mathcal{D}_i^-, \end{cases}$$

where the time and satellite state dependence on the covariance matrices was suppressed to ease the notation.

4.2. Filter implementation

Notice that the dynamics of the network, described by the local dynamics of each satellite, which are given by (23), together with the coupled output, described by the local measurement architecture, which is described by (26), is of the same form as the generic network of interconnected systems detailed in Section 2.1. This application is suitable for the application of the decentralized EKF method proposed in this paper. Before proceeding, it is important to point out some details regarding the filter implementation to this problem. First, the mass of the satellites is not estimated. The model for the evolution of the mass is considered to be exact, thus the state vector of the local EKF of satellite system $\mathcal{S}_i$ is $\mathbf{x}_i(k)$ as defined in Section 4.1. Second, note that the coupling graph $\mathcal{G}(k)$ is time-varying, thus one has to follow Algorithm 2. Note that the evolution of the output coupling topology is easily locally predictable in this problem, given the rigid configuration of the constellation. Thus, at each time instant, the predicted local output coupling topology, defined by $\hat{\mathcal{D}}_i^-(k+1) = \hat{\mathcal{D}}_i^+(k+1)$ can be easily predicted with over-confidence to ensure that information can be exchanged with every satellite in ISL range. Third, it is considered that each satellite establishes output coupling links with other satellites in ISL range up to a maximum of $|\mathcal{D}^-|_{\max} - 1 = |\mathcal{D}^+|_{\max} - 1$ neighbors. If more satellites are within ISL range than the maximum allowed neighborhood cardinality, links are removed until this constraint is satisfied. If necessary, the links that are removed are those that are established with the satellites that have the greatest number of links

established with other satellites. In this application in particular, given the periodicity of the topology, the rules for establishing ISL could also be set offline according to a given heuristic. Fourth, a covariance selection rule based on the time-varying empirical loss analogous to (20) is employed, as detailed in Remark 3. Fifth, to avoid numerical problems, the procedures detailed in Remark 4 are employed. Sixth, note that, as pointed out in Remark 5, the prediction making use of (23) is decoupled, thus the unscented transformation is used as described in Remark 5, to allow for the propagation of second-order terms of the probability distributions in the prediction step. The parameters $\alpha = 10^{-3}$, $\kappa = 0$, and $\beta = 2$ are the parameters of the implemented Unscented Transformation, defined according to Wan and Van Der Merwe (2000).

4.3. Simulation results

In this section, simulation results are presented for an illustrative mega-constellation of a single shell inspired by the first shell of the Starlink constellation to be deployed. The constellation is a Walker $53.0\,\text{deg} : 1584/72/17$. The phasing parameter of this Starlink shell is not publicly available, so it was chosen according to Liang, Chaudhry, and Yanikomeroglu (2021) such that the minimum distance between satellites is maximized. All satellites are assumed to be identical. The parameters that define the illustrative constellation are presented in Table 1. The realistic nonlinear numeric simulation was computed making use of the high fidelity TU Delft's Astrodynamic Toolbox[1] (TUDAT) (Kumar et al., 2012). The orbit propagation of the satellites of the constellation makes use of NASA's SPICE ephemerides, and accounts for several perturbations:

1. Earth's gravity field EGM96 spherical harmonic expansion (Lemoine, Factor, & Kenyon, 1998) up to degree and order 12;
2. Atmospheric drag NRLMSISE-00 model (Picone, Hedin, Drob, & Aikin, 2002), assuming constant drag coefficient and section area;
3. Cannon ball solar radiation pressure, assuming constant reflectivity coefficient and radiation area;
4. Third-body perturbations of the Sun, Moon, Venus, Jupiter, and Mars.

The numerical propagation is assured by a fourth-order Runge–Kutta integration method with fixed step-size of $T_s = 1$ s. Simulations start at 0 DBT seconds since J2000. As the goal of these simulations is to assess estimation performance, no maneuvers were performed. Note that, for this single shell, the concatenation of the states of every satellite amounts to $6 \times 1584 = 9504$ states, which is the dimension of the global system of an equivalent centralized framework. Thus, the implementation of a centralized estimation algorithm would require performing computations, in real-time, with very large dimension matrices. For instance, the estimation error covariance matrix, which is non-sparse in general, would amount to 722.6×10^6 bytes in double precision. Moreover, in a centralized framework, there would have to be all-to-all communication of large amounts of data over long distances via the mission control center, which is achieved by several ground stations scattered across the Earth. On top of that, in areas where it may not be possible to ensure a direct link with a ground station, the information must flow through a path of satellites to an available ground station, thereby putting a lot of pressure on the communication system of the satellites. As a result, the intensive real-time global computations and sheer communication requirements render a centralized framework inviable. Moreover, the large-scale of this system renders the simulation of other state-of-the-art methods computational impossible in a reasonable time frame. This is due to their computational, memory, or communication requirements, that grow with the size of the network, as pointed out in Section 1.

[1] Documentation available at https://docs.tudat.space/ and source code at https://github.com/tudat-team/tudat-bundle/.

Table 1
Parameters of the constellation.

Configuration	
Inclination (i)	53.0 deg
Number of satellites (T)	1584
Number of orbital planes (P)	72
Phasing parameter (F)	17
Semi-major axis (a)	6921.0 km
Eccentricity (e)	0
Satellites	
Initial mass	260 Kg
Drag coefficient (C_D)	2.2
Section area (A)	24.0 m^2
Solar radiation pressure coefficient (C_R)	1.2
Solar radiation pressure area (SRPA)	10.0 m^2

The simplified discrete-time model of a satellite (23) was validated, in a first instance, by assessing its accuracy considering the TUDAT propagation evolution as the ground truth. It was possible to conclude that, for this illustrative constellation in particular, which is at an height of 550 km, the exponential density model performs poorly and its short-term effect of the acceleration due to atmospheric drag is smaller than the effect of other perturbations which were not taken into account. For these reasons, henceforth, in the propagation making use of (23), the drag acceleration was neglected. Fig. 2(a) depicts the evolution of the propagation of the absolute value of the error of the simplified model (23). Fig. 2(b) depicts the evolution of the absolute propagation error of the simplified model (23) in a single time-step $T_s = 1$ s, i.e., each time-step, the states are propagated starting at the true state. In both these plots, the error is depicted along the radial, along-track, and cross-track components as a function of the mean argument of latitude for roughly 10 orbits. First, it is noticeable that, as expected, the dead-reckoning propagation diverges from the ground truth evolution, most notably in the along-track component. Second, the single time-step propagation accuracy is of the order of 10^{-4} m. Nevertheless, as expected, the propagation error does not seem to be well-modeled by a Gaussian distribution, as proposed in Section 4.1, given that correlations between time-steps are evident. Notwithstanding the poor process noise model, the EKF formulation has been shown multiple times to achieve good estimation results in these conditions (Garulli, Giannitrapani, Leomanni, & Scortecci, 2011). After careful error analysis and empirical experimentation, an adequate error covariance matrix of the process noise is given, in International System of Units (SI) units, by

$$\mathbf{Q}_j(k, \mathbf{x}_j(k)) = \begin{bmatrix} \text{diag}(1.967, 1.967, 1.456) & \text{diag}(0.2515, 0.2515, 0.2424) \\ \text{diag}(0.2515, 0.2515, 0.2424) & 10^{-2}\text{diag}(3.382, 3.382, 4.09) \end{bmatrix},$$

for $j = 1, \ldots, T$. Note that the standard deviation of the position error is four orders of magnitude above the magnitude of the error reported in Fig. 2(b).

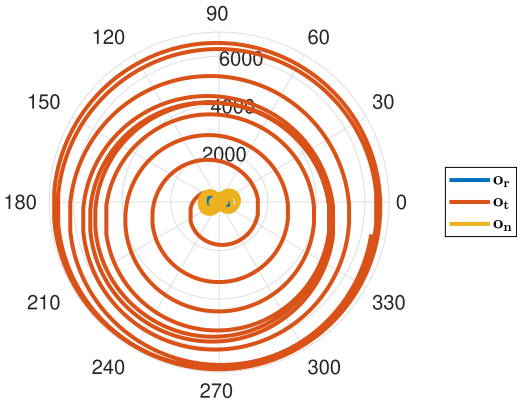
It is assumed that the inertial GNSS measurement error is uncorrelated between components and that each component has a standard deviation of 10 m, i.e.,

$$\mathbf{R}_j^{in}(k, \mathbf{x}_j(k)) = 10^2 \mathbf{I}_3,$$

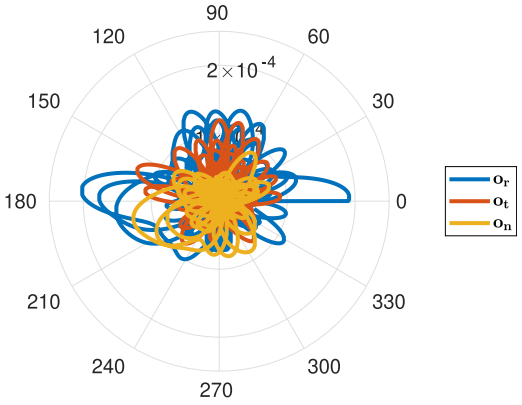
in SI units, for $j = 1, \ldots, T$. Even though experimental results are not available in the literature for baselines over 220 km, which were established for the GRACE mission (Kroes et al., 2005), since the main contribution is the decentralized EKF method, and for the sake of simplicity, it is considered that, due to common error canceling in the determination of the relative measurements and state-of-the-art ambiguity resolution methods, relative measurement error is two orders of magnitude lower than inertial measurement error. Thus,

$$\mathbf{R}_{j,p}^{rel}(k, \mathbf{x}_j(k), \mathbf{x}_j p(k)) = (0.1)^2 \mathbf{I}_3,$$

in SI units, for $j = 1, \ldots, T$ and $p \in \hat{\mathcal{D}}_j^-$.



(a) Dead-reckoning propagation.



(b) Propagation of a single time-step $T_s = 1$ s.

Fig. 2. Absolute value of the components of the propagation position error in radial, along-track, and cross-track directions as a function of the mean argument of latitude.

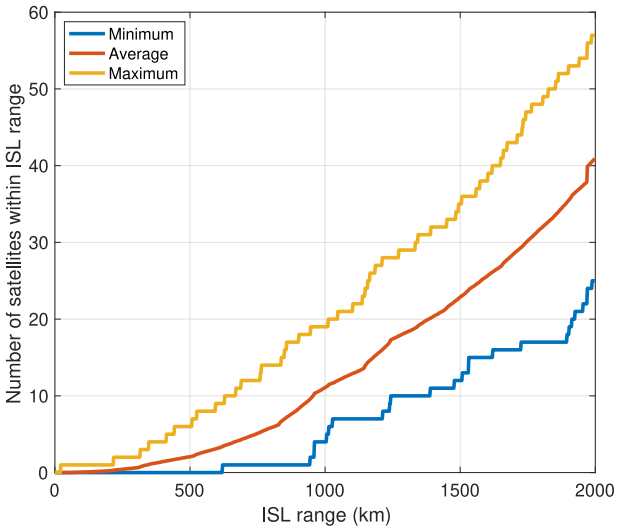


Fig. 3. Number of satellites within ISL range at 0 TDB seconds since J2000.

Table 2
RMSE of the global position error of the constellation over the whole simulation for various values of the maximum number output couplings per satellite.

$\lvert D^- \rvert_{\max} - 1$	RMSE (m)	Improvement
0 (Inertial)	6.65	–
2	2.14	−68%
3	1.29	−81%
4	1.16	−83%
∞ (All within ISL range)	1.13	−83%

The evolution of the minimum, maximum, and average number of satellites within ISL range of each satellite of the constellation as a function of the ISL range, at 0 Dynamical Barycentric Time (TDB) seconds since J2000, is depicted in Fig. 3. Notwithstanding that ISL ranges of the order of 1000 km and above have been reported (Kaur, Gupta, & Chaudhary, 2015; Radhakrishnan et al., 2016; Sharma & Kumar, 2013), as a means of saving power, reducing ISL pressure, and showing the flexibility of the proposed estimation method, a ISL range of $R_{ISL} = 750$ km is considered with $|D^-|_{\max} - 1 = |D^+|_{\max} - 1 = 3$, which corresponds to a maximum of three output couplings per satellite at a given time instant. Note that it allows for every satellite of the constellation to share a ISL with, at least, another satellite at any time.

In Fig. 4 a snapshot of the projection of the position of each satellite of the constellation over the Earth, as well as the ISL, at 0 TDB seconds since J2000 is shown. An animation of the evolution of the ground track of the constellation and of the ISL can be viewed in the aforementioned example of the DECENTER Toolbox. It is interesting to note that, due to the higher density of satellites in the extreme latitudes, much more ISL are established. This fact allows for more accurate position estimates in these regions, which is desirable to maintain the topology and avoid collisions.

Fig. 5 depicts, for satellite 1, the evolution of the position estimation error if only the GNSS inertial position measurements are used, i.e. $|D^-|_{\max} - 1 = 0$, which degenerates in a decoupled EKF. Fig. 6 depicts, for satellite 1, the evolution of the position estimation error employing the algorithm proposed in this paper, which makes use of both inertial and relative measurements with $|D^-|_{\max} - 1 = 3$. In both these plots, the evolution of the maximum 3σ bound among the three ECI frame components is also shown. The simulations were performed for a full orbital period. First, unlike the decoupled EKF with inertial measurements only, with the inclusion of relative measurements with the proposed algorithm, the estimation performance varies throughout the orbit. This behavior is due to the time-varying measurement topology. When more satellites are within ISL range, more measurements are received and the better the estimation performance is. This aspect is clear in Fig. 7, which depicts the evolution of the trace of the estimation error covariance matrix $\mathbf{P}_{i,(i,i)}$ and the number of satellites in ISL range, for satellite 1. Second, Fig. 8 depicts the evolution of the root mean squared error (RMSE) of the estimated position for satellite 1 computed along a centered moving window of length 301 s. It is possible to confirm that the inclusion of relative measurements with the proposed algorithm leads to a significant improvement in performance, that varies between two-fold near the equator and ten-fold near the poles. It is interesting to remark that near the equator fewer satellites are available to obtain relative position measurements, thus the switching nature of the coupling topology induces noticeable spikes in Figs. 6 and 7. Third, Table 2 presents the RMSE of the global position error of the constellation over the whole simulation of roughly a full orbit for various values of the maximum number of output couplings per satellite, i.e. $|D^-|_{\max} - 1$. It is possible to conclude that the fusion of relative measurements making use of the proposed distributed scheme with $|D^-|_{\max} - 1 = 3$ allows for an improvement of 81% on the position estimation accuracy. Note that the performance achieved is not significantly different from the performance that would be obtained if all output couplings within ILS range were used. Furthermore, limiting the number of neighboring satellites allows for a more homogeneous computational load in each satellite, since the number of output couplings does not vary as much between the poles and the equator.

5. Conclusion

The emergence of very large-scale networks, many of which have nonlinear dynamics, calls for very efficient filtering algorithms, whose

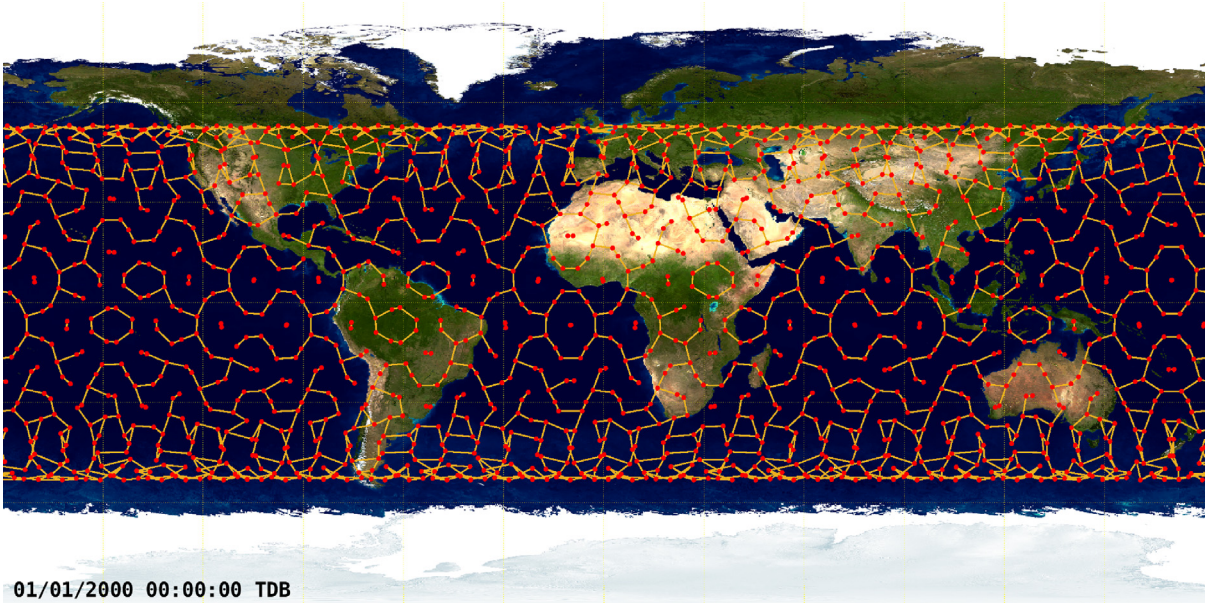


Fig. 4. Snapshot of ground track and ISL of the simulated constellation at 0 TDB seconds since J2000.

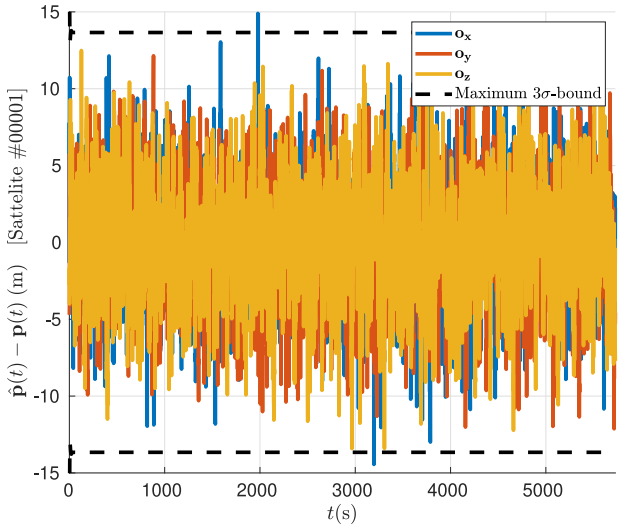


Fig. 5. Evolution of the position estimation error without the use of relative measurements, for satellite 1.

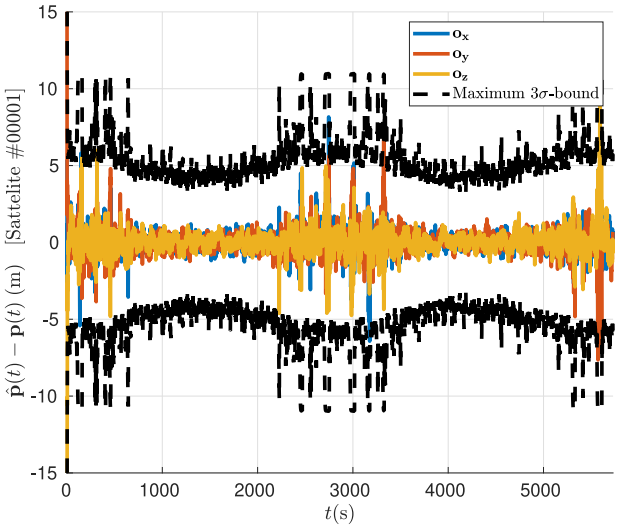


Fig. 6. Evolution of the position estimation error with the use of relative measurements, making use of the proposed algorithm, for satellite 1.

communication, memory, and computational requirements do not scale with the size of the network. In this paper, a decentralized and distributed filtering solution to large-scale networks of interconnected systems is proposed considering: (i) generic nonlinear dynamics and (ii) generic coupled nonlinear outputs in a generic, possibly time-varying, topology. The proposed solution consists of the local implementation of a filter in each system, which follows the structure of the extended Kalman filter, that estimates its own state exclusively, and has access to local communication. Heavy restrictions are taken into account to contribute to the scalability of the algorithm, which is of the upmost importance to allow for a feasible implementation to very large-scale networks of interconnected systems. The proposed estimation solution is shown: (i) not to require instantaneous transmissions of data between systems that are allowed to communicate; (ii) to make use of a number of local communication links per system that does not scale with the dimension of the network; and (iii) to require computational, memory, and data transmission capacity that does not scale with the dimension of the network. To the best of the authors' knowledge, no solutions that meet these constraints have been proposed in the literature. As a means of assessing the performance of the proposed algorithm, it is applied to the on-board absolute position estimation problem of LEO mega-constellations using GNSS. The economical viability of these constellations is dependent on the implementation of decentralized onboard filtering and control solutions. These constellations are very large-scale systems with nonlinear dynamics, make use of coupled measurements, have time-varying coupling topologies, and each satellite has very limited resources, which makes this application very challenging. The proposed method yields promising performance for a shell of the Starlink constellation. Although it is not provably consistent in general, the approach that is put forward is shown to lead to consistent estimates in this application. A MATLAB implementation

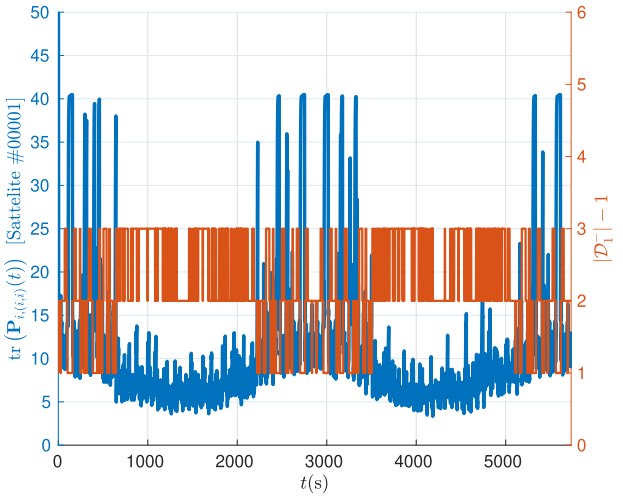


Fig. 7. Evolution of the trace of the estimation error covariance matrix $\mathbf{P}_{i,(i,i)}$ and number of satellites in ISL range, for satellite 1.

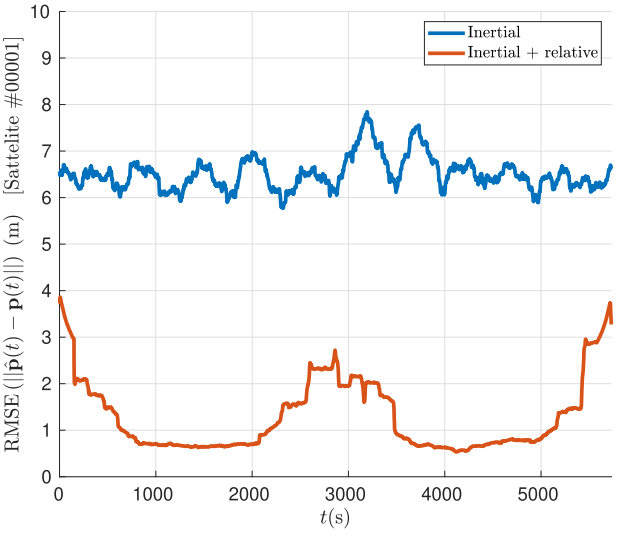


Fig. 8. Evolution of the RMSE of the estimated position for satellite 1 computed along a centered moving window of length 301 s.

of the proposed algorithm and all the source code of the simulations can be found in the DECENTER Toolbox, available online.

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.

Acknowledgments

The authors thank an anonymous reviewer for the suggestion to analyze the performance of the proposed solution for a limited number of output couplings between satellites. This work was supported by the Fundação para a Ciência e a Tecnologia (FCT) through LARSyS - FCT Project UIDB/50009/2020 and through the FCT project DECENTER [LISBOA-01-0145-FEDER-029605], funded by the Programa Operacional Regional de Lisboa 2020 and PIDDAC programs.

Appendix. Closed form solution of the one-step optimization problem

Substituting (10) in the objective function of the relaxed optimization problem (11) yields a quadratic function in relation to $\mathbf{K}(k)$. Taking its derivative w.r.t. $\mathbf{K}(k)$ yields

$$\frac{\partial \mathrm{tr}(\mathbf{P}(k|k))}{\partial \mathbf{K}(k)} = -2\mathbf{P}(k|k-1)\mathbf{C}^T(k) + 2\mathbf{K}(k)\mathbf{S}(k)\,. \tag{A.1}$$

Let $\left[\partial \mathrm{tr}(\mathbf{P}(k|k))/\partial \mathbf{K}\right]_{(i,j)} \in \mathbb{R}^{n_i \times o_j}$ and $\mathbf{K}_{(i,j)}(k) \in \mathbb{R}^{n_i \times o_j}$ denote the block of indices (i,j) of the block decomposition of (A.1) and $\mathbf{K}(k)$, respectively. Taking into account the sparsity condition of the optimization problem (11), the stationary condition verified by the optimal solution is

$$\begin{cases} \left[\partial \mathrm{tr}(\mathbf{P}(k|k))/\partial \mathbf{K}(k)\right]_{(i,j)} = \mathbf{0}, & i = j \\ \mathbf{K}_{(i,j)}(k) = \mathbf{0}, & i \neq j \end{cases}. \tag{A.2}$$

The first equation of (A.2) can be written as

$$\sum_{l=1}^{N} \mathbf{K}_{(i,l)}(k)\mathbf{S}_{(l,i)}(k) - \sum_{l=1}^{N} \mathbf{P}_{(i,l)}(k|k-1)\mathbf{C}_{(i,l)}^T(k) = \mathbf{0}. \tag{A.3}$$

Making use of the second equation in (A.2) and the fact that

$$l \notin \mathcal{D}_i^- \implies \mathbf{C}_{(i,l)}(k) = \left.\frac{\partial \mathbf{g}_i(k; \mathbf{x}_j(k), j \in \mathcal{D}_i^-)}{\partial \mathbf{x}_l(k)}\right|_{\substack{\mathbf{x}_j(k)=\hat{\mathbf{x}}_j(k|k-1) \\ j \in \mathcal{D}_i^-}} = 0, \tag{A.4}$$

then (A.3) can be written as (13). Furthermore, (14) follows from (A.4) and the definition of innovation covariance (12). Finally, (15) is obtained performing block decomposition of (9).

References

Allende-Alba, G., Montenbruck, O., Hackel, S., & Tossaint, M. (2018). Relative positioning of formation-flying spacecraft using single-receiver GPS carrier phase ambiguity fixing. *GPS Solutions*, *22*(3), 1–11.

Bereg, S., Díaz-Báñez, J. M., Lopez, M. A., Rozario, T., & Valavanis, K. (2015). A decentralized geometric approach for the formation keeping in unmanned aircraft navigation. In *2015 international conference on unmanned aircraft systems (ICUAS)* (pp. 989–997).

Blewitt, G. (1989). Carrier phase ambiguity resolution for the global positioning system applied to geodetic baselines up to 2000 km. *Journal of Geophysical Research: Solid Earth*, *94*(B8), 10187–10203.

Blondel, V. D., & Tsitsiklis, J. N. (2000). A survey of computational complexity results in systems and control. *Automatica*, *36*(9), 1249–1274.

Busse, F., & How, J. (2002). Real-time experimental demonstration of precise decentralized relative navigation for formation flying spacecraft. In *AIAA guidance, navigation, and control conference and exhibit* (p. 5003).

Carrillo-Arce, L. C., Nerurkar, E. D., Gordillo, J. L., & Roumeliotis, S. I. (2013). Decentralized multi-robot cooperative localization using covariance intersection. In *2013 IEEE/RSJ international conference on intelligent robots and systems* (pp. 1412–1417). IEEE.

Chen, S. (2011). Kalman filter for robot vision: a survey. *IEEE Transactions on Industrial Electronics*, *59*(11), 4409–4420.

Crassidis, J. L., Markley, F. L., & Cheng, Y. (2007). Survey of nonlinear attitude estimation methods. *Journal of Guidance, Control, and Dynamics*, *30*(1), 12–28.

Dai, M., Mu, H., & Wu, M. (2016). Junction-tree-based cooperative orbit determination for crosslink-augmented satellite constellations. *Proceedings of the Institution of Mechanical Engineers, Part G (Journal of Aerospace Engineering)*, *230*(10), 1848–1859.

Ferguson, P., & How, J. (2003). Decentralized estimation algorithms for formation flying spacecraft. In *AIAA guidance, navigation, and control conference and exhibit* (p. 5442).

Garulli, A., Giannitrapani, A., Leomanni, M., & Scortecci, F. (2011). Autonomous low-earth-orbit station-keeping with electric propulsion. *Journal of Guidance, Control, and Dynamics*, *34*(6), 1683–1693.

Ivanov, D., Monakhova, U., & Ovchinnikov, M. (2019). Nanosatellites swarm deployment using decentralized differential drag-based control with communicational constraints. *Acta Astronautica*, *159*, 646–657.

Julier, S. J., & Uhlmann, J. K. (1997a). New extension of the Kalman filter to nonlinear systems. In I. Kadar (Ed.), *Signal processing, sensor fusion, and target recognition VI, Vol. 3068* (pp. 182–193). International Society for Optics and Photonics. SPIE, http://dx.doi.org/10.1117/12.280797.

Julier, S. J., & Uhlmann, J. K. (1997b). A non-divergent estimation algorithm in the presence of unknown correlations. *4*, In *Proceedings of the 1997 american control conference (Cat. No. 97CH36041)* (pp. 2369–2373). IEEE.

Julier, S. J., & Uhlmann, J. K. (2004). Unscented filtering and nonlinear estimation. *Proceedings of the IEEE*, *92*(3), 401–422.

Kaur, P., Gupta, A., & Chaudhary, M. (2015). Comparative analysis of inter satellite optical wireless channel for NRZ and RZ modulation formats for different levels of input power. *Procedia Computer Science*, *58*, 572–577.

Kroes, R., Montenbruck, O., Bertiger, W., & Visser, P. (2005). Precise GRACE baseline determination using GPS. *Gps Solutions*, *9*(1), 21–31.

Kumar, K., van Barneveld, P., Dirkx, D., Melman, J., Mooij, E., & Noomen, R. (2012). Tudat: a modular and robust astrodynamics toolbox. In *5th ICATT conference* (pp. 1–8). ESA.

Lemoine, F., Factor, J., & Kenyon, S. (1998). *The development of the joint NASA GSFC and the national imagery and mapping agency (NIMA) geopotential model EGM96, Vol. 206861*. National Aeronautics and Space Administration, Goddard Space Flight Center.

Leung, K. Y., Barfoot, T. D., & Liu, H. H. (2009). Decentralized localization of sparsely-communicating robot networks: A centralized-equivalent approach. *IEEE Transactions on Robotics*, *26*(1), 62–77.

Li, Y. (2014). Offtake feedforward compensation for irrigation channels with distributed control. *IEEE Transactions on Control Systems Technology*, *22*(5), 1991–1998.

Li, Y. (2018). A novel ambiguity search algorithm for high accuracy differential GNSS relative positioning. *Aerospace Science and Technology*, *78*, 418–426.

Li, W., Jia, Y., & Du, J. (2017). Distributed consensus extended Kalman filter: a variance-constrained approach. *IET Control Theory & Applications*, *11*(3), 382–389. http://dx.doi.org/10.1049/iet-cta.2016.1054.

Li, H., Nashashibi, F., & Yang, M. (2013). Split covariance intersection filter: Theory and its application to vehicle localization. *IEEE Transactions on Intelligent Transportation Systems*, *14*(4), 1860–1871. http://dx.doi.org/10.1109/TITS.2013.2267800.

Liang, J., Chaudhry, A. U., & Yanikomeroglu, H. (2021). Phasing parameter analysis for satellite collision avoidance in starlink and kuiper constellations. In *2021 IEEE 4th 5G world forum (5GWF)* (pp. 493–498). IEEE.

Luft, L., Schubert, T., Roumeliotis, S. I., & Burgard, W. (2016). Recursive decentralized collaborative localization for sparsely communicating robots. In *Robotics: science and systems*. New York.

Luft, L., Schubert, T., Roumeliotis, S. I., & Burgard, W. (2018). Recursive decentralized localization for multi-robot systems with asynchronous pairwise communication. *International Journal of Robotics Research*, *37*(10), 1152–1167.

Madhavan, R., Fregene, K., & Parker, L. E. (2002). Distributed heterogeneous outdoor multi-robot localization. In *Proceedings 2002 IEEE international conference on robotics and automation (Cat. No. 02CH37292)* (pp. 374–381). IEEE.

Madhavan, R., Fregene, K., & Parker, L. E. (2004). Distributed cooperative outdoor multirobot localization and mapping. *Autonomous Robots*, *17*(1), 23–39.

Mandic, M., Breger, L., & How, J. (2004). Analysis of decentralized estimation filters for formation flying spacecraft. In *AIAA guidance, navigation, and control conference and exhibit* (p. 5135).

Martinelli, A. (2007). Improving the precision on multi robot localization by using a series of filters hierarchically distributed. In *2007 IEEE/RSJ international conference on intelligent robots and systems* (pp. 1053–1058). IEEE.

Montenbruck, O., Ebinuma, T., Lightsey, E. G., & Leung, S. (2002). A real-time kinematic GPS sensor for spacecraft relative navigation. *Aerospace Science and Technology*, *6*(6), 435–449.

Panzieri, S., Pascucci, F., & Setola, R. (2006). Multirobot localisation using interlaced extended kalman filter. In *2006 IEEE/RSJ international conference on intelligent robots and systems* (pp. 2816–2821). IEEE.

Park, C.-W., Ferguson, P., Pohlman, N., & How, J. P. (2001). Decentralized relative navigation for formation flying spacecraft using augmented CDGPS. In *Proceedings of the 14th international technical meeting of the satellite division of the institute of navigation (ION GPS 2001)* (pp. 2304–2315).

Pedroso, L., Batista, P., Oliveira, P., & Silvestre, C. (2021). Discrete-time distributed Kalman filter design for networks of interconnected systems with linear time-varying dynamics. *International Journal of Systems Science*, 1–18.

Picone, J., Hedin, A., Drob, D. P., & Aikin, A. (2002). NRLMSISE-00 empirical model of the atmosphere: Statistical comparisons and scientific issues. *Journal of Geophysical Research, Space Physics*, *107*(A12), SIA–15.

Prodan, I., Lefevre, L., Genon-Catalot, D., et al. (2017). Distributed model predictive control of irrigation systems using cooperative controllers. *IFAC-PapersOnLine*, *50*(1), 6564–6569.

Radhakrishnan, R., Edmonson, W. W., Afghah, F., Rodriguez-Osorio, R. M., Pinto, F., & Burleigh, S. C. (2016). Survey of inter-satellite communication for small satellite systems: Physical layer to network layer view. *IEEE Communications Surveys & Tutorials*, *18*(4), 2442–2473.

Russell Carpenter, J. (2002). Decentralized control of satellite formations. *International Journal of Robust and Nonlinear Control*, *12*(2–3), 141–161. http://dx.doi.org/10.1002/rnc.680.

Sharma, V., & Kumar, N. (2013). Improved analysis of 2.5 Gbps-inter-satellite link (ISL) in inter-satellite optical-wireless communication (IsOWC) system. *Optical Communications*, *286*, 99–102.

Tancredi, U., Renga, A., & Grassi, M. (2014). Real-time relative positioning of spacecraft over long baselines. *Journal of Guidance, Control, and Dynamics*, *37*(1), 47–58.

Thien, R. T., & Kim, Y. (2018). Decentralized formation flight via PID and integral sliding mode control. *Aerospace Science and Technology*, *81*, 322–332.

Vallado, D. A. (1997). *Fundamentals of astrodynamics and applications*. McGraw-Hill Companies, Inc. College Custom Series.

Viegas, D., Batista, P., Oliveira, P., & Silvestre, C. (2012). Decentralized observers for position and velocity estimation in vehicle formations with fixed topologies. *Systems & Control Letters*, *61*(3), 443–453. http://dx.doi.org/10.1016/j.sysconle.2011.12.004.

Wallis, W. D. (2010). *A beginner's guide to graph theory*. Springer Science & Business Media.

Wan, E. A., & Van Der Merwe, R. (2000). The unscented Kalman filter for nonlinear estimation. In *Proceedings of the IEEE 2000 adaptive systems for signal processing, communications, and control symposium (Cat. No. 00EX373)* (pp. 153–158). IEEE.

Wanasinghe, T. R., Mann, G. K., & Gosine, R. G. (2014). Decentralized cooperative localization for heterogeneous multi-robot system using split covariance intersection filter. In *2014 canadian conference on computer and robot vision* (pp. 167–174). IEEE.

Wang, X., Qin, W., Bai, Y., & Cui, N. (2016). A novel decentralized relative navigation algorithm for spacecraft formation flying. *Aerospace Science and Technology*, *48*, 28–36.

West, D. B., et al. (1996). *Introduction to graph theory, Vol. 2*. Prentice hall Upper Saddle River, NJ.

Wolfe, J. D., Speyer, J. L., Hwang, S., Lee, Y.-J., & Lee, E. (2007). Estimation of relative satellite position using transformed differential carrier-phase GPS measurements. *Journal of Guidance, Control, and Dynamics*, *30*(5), 1217–1227.

Wu, S.-C., & Bar-Sever, Y. E. (2006). Real-time sub-cm differential orbit determination of two low-earth orbiters with GPS bias fixing. In *Proceedings of the 19th international technical meeting of the satellite division of the institute of navigation (ION GNSS 2006)* (pp. 2515–2522).

Yuan, C., Licht, S., & He, H. (2017). Formation learning control of multiple autonomous underwater vehicles with heterogeneous nonlinear uncertain dynamics. *IEEE Transactions on Cybernetics*, 1–15.