# Distributed decentralized control for very large-scale systems with application to LEO satellite mega-constellations

Leonardo Pedroso
leonardo.pedroso@tecnico.ulisboa.pt

Instituto Superior Técnico, Lisboa, Portugal

November 2022

**Abstract**

The advantages and tremendous potential of very large-scale complex networks of interconnected systems are indisputable in a myriad of engineering fields, not only as business opportunities but also as a natural change towards efficiency, reliability, and scalability. State-of-the-art networked control solutions fail to meet the onerous implementation requirements befitting the very large-scale of these applications. The goal of this work follows from the self-evident void in the state-of-the-art, aiming to bring these endeavors to fruition. First, the control problem is formulated in a receding horizon control framework alongside the severe large-scale feasibility constraints. Second, a convex relaxation procedure is proposed and validated resorting to numeric and experimental results. Third, a novel distributed and decentralized networked control solution is developed leveraging the proposed convex relaxation, an approximation, and a scheduling procedure to comply with the feasibility constraints on a very large-scale. Fourth, the potential of the proposed solution is illustrated for the cooperative on-board orbit control of the Starlink mega-constellation yielding promising results.
**Keywords:** Decentralized Control, Distributed Robot Systems, Networked Control, Space Robotics, Autonomous Agents, Mega-constellations

## 1. Introduction

The classical control solutions, which are developed in a centralized framework, require: i) infrastructure for the centralized coordination; ii) the transmission of a large amount of information between every agent and the central node; and iii) serious computational power for real-time processing in the central node. Moreover, these strategies offer little robustness to failure of the central processing node or the communication infrastructure. For these reasons, as the scale of the systems increases, the classical control solutions eventually become infeasible to implement in practice. This challenge is well-known for decades and many alternatives have been proposed in a decentralized control framework. In fact, decentralized solutions rely on local computations and local communication. Thus, no central computing unit is required and, at no moment in time, any entity in the network has knowledge about the global state of the network. A plethora of applications of decentralized control have been conceptualized such as precision agriculture [1], fire-fighting, surveillance, light shows, exploration and navigation on Mars [2], irrigation networks [3], traffic networks [4], and power distribution networks [5].

Despite the very compelling robustness, flexibility, and scalability properties of decentralized architectures, most of the given examples are: i) yet to transition from conceptualization to deployment; ii) deployed only in very controlled environments as a proof of concept; or iii) implemented in practice in a very small scale with a small number of agents. The reason is clear: there are inhibiting technical challenges, especially regarding the feasibility and economic viability of the implementation of state-of-the-art algorithms to these very large-scale systems.

One remarkable exception is the development of large-scale low Earth orbit (LEO) constellations. However, although some prototypes have already started being deployed on a large-scale, their economical viability is doomed unless control algorithms befitting the challenges of such large-scale systems are developed. As pointed out in [6], the tracking telemetry and command (TT&C) system projected for these constellations does not differ from the TT&C system architecture employed for a single satellite. This system consists of a single centralized mission control center (MCC) with several ground terminals scattered across the globe to allow for continuous monitoring of the whole constellation, which is very challenging and expensive to implement because of the dimension of the network. In a decentralized configuration, low level constellation operations, such as orbit determination and constellation control, are carried out cooperatively resorting to local communication between satellites via inter-satellite links (ISL). For these reasons, the cost effectiveness of LEO mega-constellations would greatly improve with the adoption of a decentralized architecture.

Despite the large research effort in decentralized control, it remains an open problem even for linear time-invariant (LTI) systems due to its intractability [7]. Even though a considerable fraction of real-life systems can be modeled as LTI, there is a multitude of engineering problems that either require a time-varying model or can be approximated by a time-varying system employing linearization techniques. However, when it comes to the implementation of time-varying decentralized algorithms, more challenges are brought to light in addition to the intricacies of the decentralized problem. As a result, the inevitable change from large to very large-scale networked control systems calls for a consistent paradigm revolution from a control standpoint. For these reasons, heavy constrains at the: i) topological, ii) communication, iii) synthesis, iv) computational, and v) memory level must be enforced to enable a seamless practical implementation.

### 1.1. State-of-the-art

Although plenty of work has been carried out in decentralized control of LTI systems, research on solutions for linear time-varying (LTV) systems, which is naturally more challenging, has been undergone to a much lesser extent. One of the proposed approaches for the design of a decentralized controller for an arbitrary network of interconnected LTI systems is to design an $\mathcal{H}_2$-optimal control policy, which is extended for LTV systems in [8] and for time-varying network topologies in [9]. Another promising approach is to relax the underlying optimization problem so that it becomes convex, which is applied to LTI systems in [10]. Furthermore, it has been shown that the solution of the decentralized design control problem is the result of a convex optimization problem if and only if quadratic invariance of the controller set is ensured [11, 12]. Solutions to systems that satisfy the quadratic invariance condition, such as [11], are very interesting from a theoretical standpoint, but the limiting assumptions on the control networked system, imposed to achieve tractability, are rarely encountered in real-life applications. Another approach found in the literature is to decouple the network of agents into clusters of agents and consider the interactions between distinct clusters as disturbances [13]. Unlike the aforementioned approaches, the couplings between clusters are not considered at the synthesis level and, thus, are sub-optimal and require intra-cluster all-to-all communication.

Farhood et al. [8] reduce the finite-horizon regulator problem of a network of interconnected LTV systems into a sequence of linear matrix inequalities. Since the computational and memory requirements of this solution cannot be distributed across the systems in the network, the computational, memory, and communication burden render such solution unfeasible in practice for very large-scale networks. To attain a distributed solution, the problem is tackled oftentimes in a receding horizon control (RHC) framework. Research into distributed control schemes, even for decoupled LTV systems with a common control objective, is rather limited and focuses mainly on particular control problems. Nevertheless, some results for decoupled nonlinear systems have already matured, such as in [14] for leader-follower topologies and in [15] assuming a priori knowledge of the overall equilibrium. Moreover, decentralized orbit control has already been given attention for constellations of satellites, commonly employing the bounding-box method [16].

### 1.2. Goals and Outline

Comparing the state-of-the-art with the technical challenges that emerge with the envisioned very large-scale applications, such as the LEO mega-constellation venture, it is clear that they are not addressed adequately. The goal of this work follows from this self-evident void, with the aim of enabling ground-braking applications with profound societal impact that rely on very large-scale systems. This work was published in international peer-reviewed journals [17, 18] and it is also under peer-review in an international journal [19].

This paper is organized as follows. In Section 2, the decentralized RHC problem is formulated alongside the communication, computational, and memory implementation feasibility constraints. In Section 3, a convex relaxation procedure is derived and validated. In Section 4, the proposed distributed and decentralized RHC algorithm is derived. In Section 5, the distributed and decentralized RHC algorithm put forward in this work is applied to the orbit control problem of LEO mega-constellations. Finally, Section 6 presents the main conclusions of this work.

### 1.3. Notation

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 entry $(i,j)$ of a matrix $\mathbf{A}$ is denoted by $[\mathbf{A}]_{ij}$. The

$i$-th component of a vector $\mathbf{v} \in \mathbb{R}^n$ is denoted by $[\mathbf{v}]_i$ and $\mathrm{diag}(\mathbf{v})$ denotes the $n \times n$ square diagonal matrix whose diagonal is $\mathbf{v}$. The column-wise concatenation of vectors $\mathbf{x}_1, \ldots, \mathbf{x}_N$ is denoted by $\mathrm{col}(\mathbf{x}_1, \ldots, \mathbf{x}_N)$ and $\mathrm{diag}(\mathbf{A}_1, ..., \mathbf{A}_N)$ denotes the block diagonal matrix whose diagonal blocks are given by matrices $\mathbf{A}_1, ..., \mathbf{A}_N$. The Kronecker delta is denoted by $\boldsymbol{\delta}_{ij}$. 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 Cartesian product of two sets $\mathcal{A}$ and $\mathcal{B}$ is denoted by $\mathcal{A} \times \mathcal{B}$. The modulo operation is denoted by $a \bmod b$, which returns the remainder of the integer division of $a \in \mathbb{N}$ by $b \in \mathbb{N}$. The greatest integer less than or equal to $x \in \mathbb{R}$ is denoted by $\lfloor x \rfloor$.

## 2. Problem Statement

Consider a network of $N$ systems, $\mathcal{S}_i$ with $i = 1, \ldots, N$, each associated with one computational unit, $\mathcal{T}_i$. Each system is modeled by LTV dynamics, which are coupled with a set of other systems accoding to the directed graph $\mathcal{G}_d$. Each system has also an LTV tracking output, which is coupled with another set of systems according to the directed graph $\mathcal{G}_o$. For more details on the graph representation and inherent notation see [20]. The dynamics of system $\mathcal{S}_i$ are modeled by the discrete-time LTV system

$$\begin{cases} \mathbf{x}_i(k+1) = \sum\limits_{j \in {}^d\mathcal{D}_i^-} \mathbf{A}_{i,j}(k)\mathbf{x}_j(k) + \sum\limits_{j \in {}^d\mathcal{D}_i^-} \mathbf{B}_{i,j}(k)\mathbf{u}_j(k) \\ \mathbf{z}_i(k) = \sum_{j \in {}^o\mathcal{D}_i^-} \mathbf{H}_{i,j}(k)\mathbf{x}_j(k), \end{cases} \tag{1}$$

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, and $\mathbf{z}_i(t) \in \mathbb{R}^{o_i}$ is the tracking output vector, all of system $\mathcal{S}_i$; ${}^d\mathcal{D}_i^-$ and ${}^o\mathcal{D}_i^-$ are the in-neighborhood of system $\mathcal{S}_i$ in graphs $\mathcal{G}_d$ and $\mathcal{G}_o$, respectively; matrices $\mathbf{A}_{i,j}(k)$ with $j \in {}^d\mathcal{D}_i^-$, $\mathbf{B}_{i,j}(k)$ with $j \in {}^d\mathcal{D}_i^-$, and $\mathbf{H}_{i,j}(k)$ with $j \in {}^o\mathcal{D}_i^-$ are time-varying matrices that model the dynamics of system $\mathcal{S}_i$ and its tracking output couplings with the other systems in its in-neighborhood. Note that linearization techniques can be employed to approximate the dynamics of a nonlinear system with a nonlinear tracking output as an LTV system of the form of (1). The communication topology can also be represented by a directed graph $\mathcal{G}_c$. If $\mathcal{S}_i$ has access to the state of $\mathcal{S}_j$ via directed communication from $\mathcal{S}_j$ to $\mathcal{S}_i$, then it is represented by an edge directed from vertex $j$ towards vertex $i$ in $\mathcal{G}_c$. Fig. 1 depicts a scheme of the dynamic, output, and communication topologies. The control input of $\mathcal{S}_i$ is, thus, of the form

$$\mathbf{u}_i(k) = -\sum_{j \in {}^c\mathcal{D}_i^-} \mathbf{K}_{i,j}(k)\mathbf{x}_j(k), \tag{2}$$

where $\mathbf{K}_{i,j}(k)$ for $j \in {}^c\mathcal{D}_i^-$ are the controller gains of $\mathcal{S}_i$.

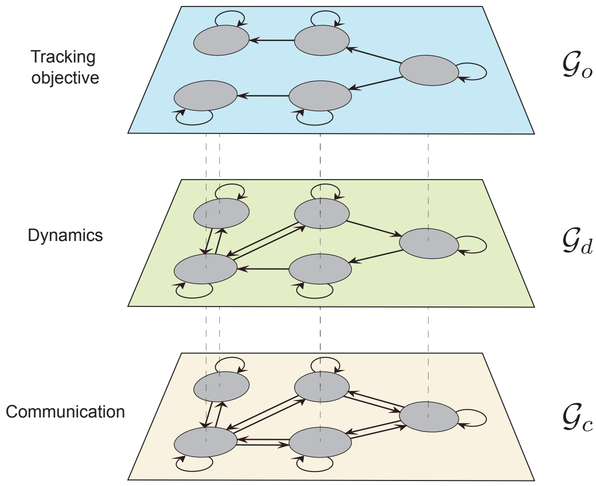



Figure 1: Scheme of the dynamics, tracking output, and communication topologies.

The global dynamics of the network can, then, be modeled by the discrete-time LTV system

$$\begin{cases} \mathbf{x}(k+1) = \mathbf{A}(k)\mathbf{x}(k) + \mathbf{B}(k)\mathbf{u}(k) \\ \mathbf{z}(k) = \mathbf{H}(k)\mathbf{x}(k), \end{cases}$$

where $\mathbf{x}(k) := \mathrm{col}(\mathbf{x}_1(k), \ldots, \mathbf{x}_N(k)) \in \mathbb{R}^n$ is the global state vector; $\mathbf{u}(k) := \mathrm{col}(\mathbf{u}_1(k), \ldots, \mathbf{u}_N(k)) \in \mathbb{R}^m$ is the global input vector; $\mathbf{z}(k) := \mathrm{col}(\mathbf{z}_1(k), \ldots, \mathbf{z}_N(k)) \in \mathbb{R}^o$ is the global tracking output vector; $\mathbf{A}(k)$ is a block matrix whose block of indices $(i,j)$ is $\mathbf{A}_{i,j}(k)$, if $j \in {}^d\mathcal{D}_i^-$, and $\mathbf{0}_{n_i \times n_j}$ otherwise; $\mathbf{B}(k)$ is a block matrix whose block of indices $(i,j)$ is $\mathbf{B}_{i,j}(k)$, if $j \in {}^d\mathcal{D}_i^-$, and $\mathbf{0}_{n_i \times m_j}$ otherwise; and $\mathbf{H}(k)$ is a block matrix whose block of indices $(i,j)$ is $\mathbf{H}_{i,j}(k)$, if $j \in {}^o\mathcal{D}_i^-$, and $\mathbf{0}_{o_i \times n_j}$ otherwise. The global control input is given by

$$\mathbf{u}(k) = -\mathbf{K}(k)\mathbf{x}(k), \tag{3}$$

where $\mathbf{K}(k)$ is the global gain matrix. Note that the global control law (3) is equivalent to the concatenation of the local control laws (2) if and only if $\mathbf{K}(k)$ follows the sparsity pattern of block matrix $\mathbf{E}_{\mathcal{G}_c}$, whose block of indices $(i,j)$ is given by

$$\mathbf{E}_{\mathcal{G}_c\, i,j} = \begin{cases} \mathbf{1}_{m_i \times n_j}, & j \in {}^c\mathcal{D}_i^- \\ \mathbf{0}_{m_i \times n_j}, & j \notin {}^c\mathcal{D}_i^-. \end{cases}$$

This sparsity condition is denoted as $\mathbf{K}(k) \in \mathrm{Sparse}(\mathbf{E}_{\mathcal{G}_c})$, with

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

The goal is to minimize an infinite-horizon performance cost

$$J_\infty := \sum_{i=1}^{N}\sum_{\tau=0}^{\infty}\big(\mathbf{z}_i^T(\tau)\mathbf{Q}_i(\tau)\mathbf{z}_i(\tau)+\mathbf{u}_i^T(\tau)\mathbf{R}_i(\tau)\mathbf{u}_i(\tau)\big),$$

where $\mathbf{Q}_i(\tau) \succeq 0$ and $\mathbf{R}_i(\tau) \succ 0$ are known time-varying matrices of appropriate dimensions that weigh the local tracking output and input of each system $\mathcal{S}_i$, respectively. The proposed method consists of an approximation to the solution of the infinite-horizon problem above, considering multiple global finite-horizon problems with an associated cost of the form

$$J(k) := \sum_{i=1}^{N}\Bigg(\mathbf{z}_i^T(k+H)\mathbf{Q}_i(k+H)\mathbf{z}_i(k+H) + \sum_{\tau=k}^{k+H-1}\big(\mathbf{z}_i^T(\tau)\mathbf{Q}_i(\tau)\mathbf{z}_i(\tau)+\mathbf{u}_i^T(\tau)\mathbf{R}_i(\tau)\mathbf{u}_i(\tau)\big)\Bigg),$$

where $H \in \mathbb{N}$ denotes the length of the finite window. The extension of this problem to an infinite-horizon is achieved by making use of the RHC scheme. At each discrete time instant $k$, one considers a finite window $\{k, ..., k+H\}$. Then, the gains that minimize $J(k)$ are computed for the appropriate window and only the first is actually used to compute the control action for that time instant, discarding the remaining gains. At the next time instant, $k+1$, a new finite window is considered and a new sequence of gains is computed to minimize $J(k+1)$, and so forth. To reduce the computational load, $d \in \mathbb{N}$ gains may be used, instead of just one, defining a new window and computing the gains associated with it every $d$ time steps.

One aims to optimally compute a sequence of gains that follow the sparsity pattern required for a fully decentralized configuration. For a finite-horizon, solve the optimization problem

$$\begin{aligned}
\underset{\substack{\mathbf{K}(\tau)\in\mathbb{R}^{m\times n}\\ \tau\in\{k,\dots,k+H-1\}}}{\text{minimize}} \quad & J(k)\\
\text{subject to} \quad & \mathbf{K}(\tau) \in \mathrm{Sparse}(\mathbf{E}_{\mathcal{G}_c}),\\
& \mathbf{u}(\tau) = -\mathbf{K}(\tau)\mathbf{x}(\tau),\\
& \mathbf{x}(\tau+1) = \mathbf{A}(\tau)\mathbf{x}(\tau)+\mathbf{B}(\tau)\mathbf{u}(\tau),\\
& \tau = k,\dots,k+H-1.
\end{aligned} \tag{4}$$

It is of the utmost importance to remark that the solution devised for (4) must be feasible to implement in real-time in a decentralized configuration. In particular, the procedure to compute each gain $\mathbf{K}_{i,j}(k)$, with $j \in {}^{c}\mathcal{D}_i^-$, in $\mathcal{T}_i$ must abide by several constraints regarding communication, computational, and memory requirements. For more details see [20]. In short, the control solution must satisfy the following constraints.

**Constraint 1.** Hard real-time transmissions are not allowed for the synthesis of controller gains.

**Constraint 2.** The communication complexity of each system ought to grow with $\mathcal{O}(1)$ with $N$.

**Constraint 3.** The data storage complexity of each unit ought to grow with $\mathcal{O}(1)$ with $N$.

**Constraint 4.** The computational complexity of each unit ought grow with $\mathcal{O}(1)$ with $N$.

## 3. Decentralized linear quadratic control

In this work, a divide-and-conquer approach is followed. In this section, the decentralized RHC problem stated in Section 2 is addressed disregarding the computational, memory, and communication feasibility constraints. These results are then leveraged, in Section 4, to devise a distributed synthesis procedure that abides by them, for the particular case of dynamically decoupled systems. In this section, for simplicity, we consider time-invariant topologies, but the results in Section 4 are extended to consider time-varying topologies.

The optimization problem (4) is nonconvex. Thus, to use standard optimization techniques, convex relaxation is performed. In this work, the proposed convex relaxation procedure, designated one-step relaxation, is derived from an analysis of the necessary conditions of a constrained minimum of (4). Nevertheless, to compute the solution to those conditions, the state at the beginning of the window, i.e, $\mathbf{x}(k)$, would have to be known, which is impossible to achieve without all-to-all communication. The intuition behind the proposed one-step relaxation is to achieve a decentralized gain synthesis procedure that does not depend on the initial state of each finite window, arriving at the conditions

$$\begin{cases}\left[\mathbf{S}(\tau)\mathbf{K}(\tau)-\mathbf{B}^T(\tau)\mathbf{P}(\tau+1)\mathbf{A}(\tau)\right]_{ji}=0, & [\mathbf{E}_{\mathcal{G}_c}]_{ji}\neq 0\\ [\mathbf{K}(\tau)]_{ji} = 0, & [\mathbf{E}_{\mathcal{G}_c}]_{ji}=0,\end{cases} \tag{5}$$

for $\tau = k,\dots,k+H-1$, where $\mathbf{P}(\tau)$ is a symmetric positive semidefinite matrix given by recurrently

$$\begin{aligned}\mathbf{P}(\tau) = {} & \mathbf{H}(\tau)^T\mathbf{Q}(\tau)\mathbf{H}(\tau) + \mathbf{K}^T(\tau)\mathbf{R}(\tau)\mathbf{K}(\tau) + \\ & (\mathbf{A}(\tau)-\mathbf{B}(\tau)\mathbf{K}(\tau))^T\mathbf{P}(\tau+1)(\mathbf{A}(\tau)-\mathbf{B}(\tau)\mathbf{K}(\tau)),\end{aligned} \tag{6}$$

for $\tau = k,\dots,k+H-1$,

$$\mathbf{P}(k+H) = \mathbf{H}^T(k+H)\mathbf{Q}(k+H)\mathbf{H}(k+H), \tag{7}$$

and

$$\mathbf{S}(\tau) := \mathbf{B}^T(\tau)\mathbf{P}(\tau+1)\mathbf{B}(\tau) + \mathbf{R}(\tau), \tag{8}$$

for $\tau = k,\dots,k+H-1$. For more details on the principles behind the one-step relaxation see [20].

**Theorem 3.1.** *Let $\mathbf{l}_j$ denote a column vector whose entries are all set to zero except for the $j$-th one, which is set to* 1*, and $\boldsymbol{\mathcal{L}}_j := \mathrm{diag}(\mathbf{l}_j)$. Define a vector $\mathbf{m}_j \in \mathbb{R}^m$ to encode the non-zero entries in the $j$-th column of $\mathbf{K}(\tau)$ as*

$$\begin{cases} [\mathbf{m}_j]_i = 0, & [\mathbf{E}_{\mathcal{G}_c}]_{ij} = 0 \\ [\mathbf{m}_j]_i = 1, & [\mathbf{E}_{\mathcal{G}_c}]_{ij} \neq 0 \end{cases}, \; i = 1, ..., m \,,$$

*and let $\boldsymbol{\mathcal{M}}_j := \mathrm{diag}(\mathbf{m}_j)$. Then, the gain of the one-step sub-optimal solution to* (4) *is given by*

$$\begin{aligned} \mathbf{K}(\tau) = \sum_{j=1}^{n} \left(\mathbf{I} - \boldsymbol{\mathcal{M}}_j + \boldsymbol{\mathcal{M}}_j \mathbf{S}(\tau) \boldsymbol{\mathcal{M}}_j\right)^{-1} \\ \boldsymbol{\mathcal{M}}_j \mathbf{B}^T(\tau) \mathbf{P}(\tau+1) \mathbf{A}(\tau) \boldsymbol{\mathcal{L}}_j, \end{aligned} \tag{9}$$

$\tau = k, \ldots, k+H-1$.

It is important to remark that the sequence of gains that arises in Theorem 3.1 can only be computed backward in time. It is solved sequentially starting at $\tau = k+H-1$ with $\mathbf{P}(k+H)$ known by the boundary condition in (7). Then, the one-step solution is found by taking turns computing $\mathbf{K}(\tau)$ with (9) and $\mathbf{P}(\tau)$ with (6).

Note that the problem is formulated globally, thus, the local decentralized controller gains can then be extracted from the globally synthesized sparse gain matrices, which allows for its decentralized implementation according to (2), leveraging local communication exclusively. It is very important to remark that the computation of the one-step solution according to Theorem 3.1, makes use of global matrices. For that reason, it follows that all-to-all communication for the one-step gain synthesis procedure can only be avoided if the global computations are replicated in each computational unit. In this architecture, the global optimization problem (4) would have to be solved approximately in each computational unit at each time instant to extract the local gain corresponding to that system. Thus, according to published work by the author [21], the computational and memory complexity in each computational unit would grow with $\mathcal{O}(N^3)$ with the dimension of the network. These requirements are not in line with the computational and memory feasibility constraints set forth in Section 2.

More often than not, one is interested in tracking a reference signal, $\mathbf{r}(k)$, with the output of the system, $\mathbf{z}(k)$, instead of driving the output of the system to zero. On one hand, if the reference signal is feasible, in the sense that there exists a sequence of inputs that drive the output along the desired trajectory, the tracking problem degenerates into a regulation problem [22, Chapter 4]. In fact, in these conditions, it easy to define the tracking error dynamics and apply the one-step regulator solution. On the other hand, if the trajectory is not feasible, the extension is not as straightforward. In [20], a tracker suitable both for centralized and decentralized configurations is designed building on the one-step regulator.

In [20], the one-step regulator and tracker methods are numerically applied to a network of $N = 40$ interconnected tanks, which corresponds to the generalization of the quadruple-tank network introduced in [23]. It is also validated resorting to experimental results in a network of four tanks, which is a particularization of the $N$ tanks network for $N = 4$. The experimental setup is proposed and thoroughly analyzed in published work by the author [18]. First, it was possible to validate the convex relaxation approach and to show the scalability of the proposed methods. Second, it was concluded that the synthesized decentralized control law is able to successfully track a time-varying reference signal and to reject impulsive and constant disturbances.

## 4. Distributed and Decentralized RHC

In this section, the one-step relaxation is leveraged to design a decentralized control solution that approximates the solution to optimization problem (4) subject to the communication, computational, and memory restrictions in Constraints 1–4 put forward in Section 2, which are critical for a feasible implementation to very large-scale systems. The particular case of dynamically decoupled systems is considered. The solution is presented, in a first instance, for a time-invariant network topology and then it is extended to a time-varying topology.

Before proceeding with the derivation, it is convenient to make a few considerations to lighten the notation employed henceforth. Dynamically decoupled systems are considered, thus, $\mathbf{A} := \mathrm{diag}(\mathbf{A}_1, \ldots, \mathbf{A}_N)$ and $\mathbf{B} := \mathrm{diag}(\mathbf{B}_1, \ldots, \mathbf{B}_N)$, where $\mathbf{A}_i$ and $\mathbf{B}_i$ denote, for simplicity, matrices $\mathbf{A}_{i,i}$ and $\mathbf{B}_{i,i}$, respectively, for $i \in \{1, \ldots, N\}$. Furthermore, local state feedback via directed communication is allowed according to graph $\mathcal{G}_c$, which, for the sake of simplicity, is selected to be equal to the tracking output coupling graph $\mathcal{G}_o$. These graphs are henceforth denoted by $\mathcal{G} = \mathcal{G}_o = \mathcal{G}_c$.

Define a block decomposition of $\mathbf{P}(\tau)$ and $\mathbf{S}(\tau)$, whose blocks of indices $(i,j)$ are denoted by $\mathbf{P}_{i,j}(\tau) \in \mathbb{R}^{n_i \times n_j}$ and $\mathbf{S}_{i,j}(\tau) \in \mathbb{R}^{m_i \times m_j}$, respectively. Making use of this block decomposition, one can also express the blocks of $\mathbf{S}(\tau)$ as a function of the blocks of $\mathbf{P}(\tau+1)$ as

$$\mathbf{S}_{i,j}(\tau) = \mathbf{B}_i^T(\tau) \mathbf{P}_{i,j}(\tau+1) \mathbf{B}_j(\tau) + \boldsymbol{\delta}_{ij} \mathbf{R}_j(\tau),$$

which follows immediately from (8).

Moreover, leveraging the aforementioned block decomposition, the relaxed conditions (5) of the feedback gains of the form $\mathbf{K}_{j,i}(\tau)$ can also be written in a decoupled manner, for each $i \in \{1, \ldots, N\}$, as

$$\begin{cases} \sum_{p \in \mathcal{D}_i^+} \mathbf{S}_{j,p}(\tau)\mathbf{K}_{p,i}(\tau) \\ \quad -\mathbf{B}_j^T(\tau)\mathbf{P}_{j,i}(\tau+1)\mathbf{A}_i(\tau) = \mathbf{0}, & j \in \mathcal{D}_i^+ \\ \mathbf{K}_{j,i}(\tau) = \mathbf{0}, & j \notin \mathcal{D}_i^+. \end{cases} \tag{10}$$

For each set $\mathcal{D}_i^+$, let $\mathcal{D}_i^+ = \left\{p_1^i, \ldots, p_{|\mathcal{D}_i^+|}^i\right\}$. Then, concatenating the feedback gains of the form $\mathbf{K}_{j,i}(\tau)$, with $j \in \mathcal{D}_i^+$, and combining the corresponding decoupled relaxed conditions of the first member of (10), it follows that

$$\tilde{\mathbf{K}}_i(\tau) = \tilde{\mathbf{S}}_i(\tau)^{-1}\tilde{\mathbf{P}}_i(\tau+1), \tag{11}$$

where

$$\tilde{\mathbf{K}}_i(\tau) := \begin{bmatrix} \mathbf{K}_{p_1^i,i} \\ \vdots \\ \mathbf{K}_{p^i_{|\mathcal{D}_i^+|},i} \end{bmatrix},$$

$$\tilde{\mathbf{S}}_i(\tau) := \begin{bmatrix} \mathbf{S}_{p_1^i,p_1^i} & \cdots & \mathbf{S}_{p_1^i,p^i_{|\mathcal{D}_i^+|}} \\ \vdots & \ddots & \vdots \\ \mathbf{S}_{p^i_{|\mathcal{D}_i^+|},p_1^i} & \cdots & \mathbf{S}_{p^i_{|\mathcal{D}_i^+|},p^i_{|\mathcal{D}_i^+|}} \end{bmatrix},$$

and

$$\tilde{\mathbf{P}}_i(\tau+1) := \begin{bmatrix} \mathbf{B}^T_{p_1^i}(\tau)\mathbf{P}_{p_1^i,i}(\tau+1)\mathbf{A}_i(\tau) \\ \vdots \\ \mathbf{B}^T_{p^i_{|\mathcal{D}_i^+|}}(\tau)\mathbf{P}_{p^i_{|\mathcal{D}_i^+|},i}(\tau+1)\mathbf{A}_i(\tau) \end{bmatrix}.$$

The propagation of $\mathbf{P}(\tau)$, according to (6), is required for the computation of the decoupled gains according to (11) but it cannot be decoupled. Nevertheless, that can be achieved under a reasonable approximation.

**Approximation 4.1.** Consider $\mathbf{P}_{p,q}(\tau)$, with $p \in \mathcal{D}_i^+$ and $q \in \mathcal{D}_i^+$ for some $i$, and $\mathbf{P}_{r,s}(\tau+1)$, with $r \in \mathcal{D}_p^+$ and $s \in \mathcal{D}_q^+$. In the decentralized algorithm put forward in this work, $\mathbf{P}_{r,s}(\tau+1)$ is considered to be null in the computation of $\mathbf{P}_{p,q}(\tau)$ in the computational unit $\mathcal{T}_i$ if $(r,s) \notin \psi_i$, where

$$\psi_i = \bigcup_{j \in \mathcal{D}_i^+} \phi_j,$$

with

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

The main result of this chapter is supported by Approximation 4.1. Next, it is argued that this approximation makes sense in the context of RHC of a large-scale network. Note that $\mathbf{P}_{r,s}(\tau)$ is a measure of the contribution of the correlation between the states of systems $\mathcal{S}_r$ and $\mathcal{S}_s$ to the global cost. Consider Fig. 2, which represents the topology of Approximation 4.1 in a graph. Intuitively, it is expected that the influence of $\mathbf{P}_{r,s}(\tau+1)$ is more dominant in the computation of $\mathbf{K}_{p,i}(k)$ for $p \in \mathcal{D}_i^+$ if both the states of $\mathcal{S}_r$ and $\mathcal{S}_s$ are coupled with the output of a system $\mathcal{S}_k$ that is coupled with the output of $\mathcal{S}_i$. Having this in mind, to decouple the gain synthesis of each local controller, each computational unit $\mathcal{T}_i$ keeps and updates each $\mathbf{P}_{p,q}(\tau)$ with $(p,q) \in \phi_i$. Henceforth, the approximation of matrix $\mathbf{P}_{p,q}(\tau)$ that stored and updated in $\mathcal{T}_i$ is denoted by $\mathbf{P}_{i,(p,q)}(\tau)$.

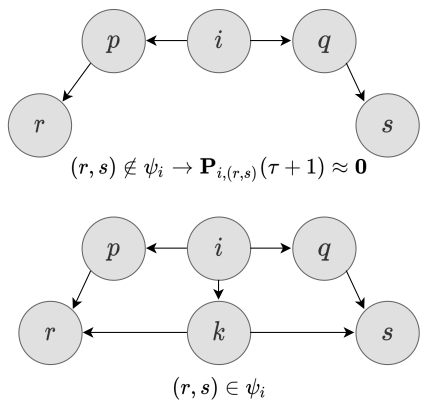



Figure 2: Graphic illustration of Approximation 4.1.

Making use of Approximation 4.1, one can rewrite (6) in a decoupled manner for each of the blocks of $\mathbf{P}(\tau)$ as a function of the local dynamics and tracking output matrices as

$$\mathbf{P}_{p,q}(k+H) = \sum_{r \in \mathcal{D}_p^+ \cap \mathcal{D}_q^+} \mathbf{H}^T_{r,i}(k+H)\mathbf{Q}_r(k+H)\mathbf{H}_{r,j}(k+H)$$

and

$$\begin{aligned} \mathbf{P}_{i,(p,q)}(\tau) = & \sum_{r \in \mathcal{D}_p^+ \cap \mathcal{D}_q^+} \mathbf{H}^T_{r,i}(\tau)\mathbf{Q}_r(\tau)\mathbf{H}_{r,j}(\tau) \\ & + \sum_{r \in \mathcal{D}_p^+ \cap \mathcal{D}_q^+} \mathbf{K}^T_{r,i}(\tau)\mathbf{R}_r(\tau)\mathbf{K}_{r,j}(\tau) \\ & + \sum_{r \in \mathcal{D}_p^+} \sum_{\substack{s \in \mathcal{D}_q^+ \\ (r,s) \in \psi_i}} (\mathbf{A}_p(\tau)\boldsymbol{\delta}_{pr} - \mathbf{B}_r(\tau)\mathbf{K}_{r,p}(\tau))^T \\ & \mathbf{P}_{\mathcal{D}_i^+,(r,s)}(\tau+1)\left(\mathbf{A}_q(\tau)\boldsymbol{\delta}_{qs} - \mathbf{B}_s(\tau)\mathbf{K}_{s,q}(\tau)\right), \end{aligned} \tag{12}$$

where the subscript $\mathcal{D}_i^+$ in $\mathbf{P}_{\mathcal{D}_i^+,(r,s)}(\tau+1)$ indicates, by abuse of notation, that $\mathbf{P}_{\mathcal{D}_i^+,(r,s)}(\tau+1)$ is computed in $\mathcal{T}_k$ with $k \in \mathcal{D}_i^+$. Note that, with Approximation 4.1, the propagation of $\mathbf{P}(\tau)$ in (12) can be computed in a distributed manner. It is important to remark that $\mathbf{P}_{\mathcal{D}_i^+,(r,s)}(\tau+1)$, inside the summation in (12), is not necessarily computed in $\mathcal{T}_i$, since only $\mathbf{P}_{i,(p,q)}(k|k)$, with $(p,q) \in \phi_i$, are updated in $\mathcal{T}_i$. Therefore, $\mathcal{S}_i$ has to receive $\mathbf{P}_{k,(r,s)}(\tau+1)$ through communication from a system $\mathcal{S}_k$, with $k \in \mathcal{D}_i^+$.

After introducing Approximation 4.1, which allows for the decoupling of the gain synthesis procedure, it is possible to state the proposed RHC algorithm, which is suppressed due to space constraints. This algorithm allows for distributing the global computation across the computational units of the systems that make up the network. Nevertheless, recall that, as put forward in Section 2, for the application of this framework to the infinite-horizon problem, a new window of gains of length $H$ has to be computed every $d$ time steps, of which only $d$ gains are used to compute the control input according to (2). Furthermore, the decoupled gain computation is carried out backward in time. Thus, at the time instant that corresponds to the beginning of each window, all RHC gains over that window must have already been computed. Since these computations involve several communication instances, they have to be properly scheduled in each local unit. A scheduling procedure that abides by the very large-scale implementation communication, memory, and computational feasibility constraints detailed in Section 2 is presented in [20].

Oftentimes, the tracking 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 tracking configurations. In [20], the distributed and decentralized control solution is extended to allow for a time-varying tracking output coupling topology. Even though the extension to a time-varying topology is quite straightforward as far as distributing the gain synthesis across of the systems is concerned, that is not the case for the scheduling of the computations over time.

## 5. Application to on-board orbit control of LEO mega-constellations

In this section, the distributed decentralized RHC algorithm developed in Section 4 is applied to the cooperative orbit control problem of LEO mega-constellations. The scheme presented in this section is novel and it is developed aiming for efficiency and fuel saving in a distributed and decentralized framework.

Consider a constellation with a total of $T$ satellites. The satellites are evenly distributed over $P$ orbital planes at a nominal inclination $\bar{i}$ and with a nominal relative phasing between adjacent planes of $\bar{\beta} = 2\pi F/T$, where $F$ is the phasing parameter. Such a configuration is designated as a Walker constellation and it is denoted by $\bar{i} : T/P/F$. The nominal orbits are circular and have a semi-major axis of $\bar{a}$. This constellation can be modeled as a network of coupled systems, $\mathcal{S}_j$, each associated with a computational unit $\mathcal{T}_j$, with $j = 1, ..., T$. Each satellite $\mathcal{S}_i$ is equipped with Hall effect thrusters, aligned according to the local TNW frame ($x$ axis along the velocity vector, $z$ axis along the orbit's angular momentum vector, and $y$ axis completes the right-handed coordinate system) that generate a force $\mathbf{u}_i \in \mathbb{R}^3$ expressed in the TNW local frame. Each thruster has a maximum thrust, $C_{t1}$.

In this application, for control law synthesis purposes, the parameterization of the orbits of each satellite of the constellation is achieved by the set of non-singular mean orbital elements for near-circular inclined orbits $(a, u, e_x, e_y, i, \Omega)$, respectively semi-major axis, mean argument of latitude, two eccentricity vector components, inclination, and longitude of ascending node. Denote the state of a satellite $\mathcal{S}_i$, made up of the aforementioned six non-singular mean orbital elements, by $\mathbf{x}_i(t) = [a_i(t)\, u_i(t)\, e_{xi}(t)\, e_{yi}(t)\, i_i(t)\, \Omega_i(t)]^T$.

The satellite orbital mechanics are nonlinear and, thus, have to be linearized to employ the distributed and decentralized method put forward in this work. The linearization of the dynamics of each satellite is carried out about a nominal orbit. These are defined such that the set of nominal orbits of all satellites makes up a consistent nominal constellation, in the sense that the nominal separations: i) along-track; ii) inter-plane; and iii) in relative phasing between adjacent planes are enforced. It is very important to remark that this nominal constellation is used for linearization purposes only – it is not employed for bounding-box tracking of each individual satellite at any point. The nominal state of $\mathcal{S}_i$ at time instant $t$, $\bar{\mathbf{x}}_i(t) = \left[\bar{a}_i(t)\ \bar{u}_i(t)\ \bar{e}_{xi}(t)\ \bar{e}_{yi}(t)\ \bar{i}_i(t)\ \bar{\Omega}_i(t)\right]^T$, is defined by

$$\begin{cases} \bar{a}_i(t) = \bar{a} \\ \bar{u}_i(t) = \bar{u}_{t_0} + ((i-1) \bmod T/P)\, 2\pi P/T \\ \quad + \lfloor (i-1)P/T \rfloor 2\pi F/T + (\dot{M} + \dot{\omega})(t - t_0) \\ \bar{e}_{x,i}(t) = 0 \\ \bar{e}_{y,i}(t) = 0 \\ \bar{i}_i(t) = \bar{i} \\ \bar{\Omega}_i(t) = \bar{\Omega}_{t_0} + \lfloor (i-1)P/T \rfloor 2\pi/P + \dot{\Omega}(t - t_0), \end{cases} \tag{13}$$

where $\dot{M}$, $\dot{\omega}$, and $\dot{\Omega}$ are the secular rates, including the effect of $J_2$, on the mean anomaly, argument of perigee, and longitude of ascending node, respectively, which are given in [24, Chapter 8]. Note that the nominal orbits of all satellites in (13) depend on three constellation-wise parameters $(t_0, \bar{u}_{t_0}, \bar{\Omega}_{t_0})$, whose physical meaning is that the nominal orbit of $\mathcal{S}_1$ has mean argument of latitude $\bar{u}_{t_0}$ and longitude of ascending node $\bar{\Omega}_{t_0}$ at time instant $t_0$. These three parameters are designed herein as the anchor of the nominal constellation. There are a few aspects worth pointing out regarding the anchor of the nominal constellation: i) all satellites must agree on an anchor for the nominal constellation at any time instant; ii) to minimize linearization errors, the anchor should be selected such that the nominal position of each satellite is as close as possible to their actual position according to a selected metric; iii) the evolution of the nominal states takes the effect of the Earth's oblateness into account, which significantly decreases the frequency with which the anchor has to be updated; iv) given that the anchor is computed very sporadically, it can either be computed in a centralized node and then the solution broadcast to the network or solved distributively over a period of time making use of distributed gradient methods with asymptotic consensus guarantees.

The evolution of the state of $\mathcal{S}_i$ is linearized about the aforementioned nominal orbits, defining a relative position $\delta\mathbf{x}_i(t)$ based the set of orbital elements $\delta\mathbf{x}_i(t) := [a_i(t)\ \delta u_i(t)\ \delta e_{x,i}(t)\ \delta e_{y,i}(t)\ \delta i_i(t)\ \delta\Omega_i(t)]$, introduced in [25], which is defined as

$$\delta\mathbf{x}_i(t) := \begin{bmatrix} a_i(t)/\bar{a}_i(t) - 1 \\ u_i(t)-\bar{u}_i(t) + \big(\Omega_i(t)-\bar{\Omega}_i(t)\big)\cos\bar{i}_i(t) \\ e_{x,i}(t) - \bar{e}_{x,i}(t) \\ e_{y,i}(t) - \bar{e}_{y,i}(t) \\ i_i(t) - \bar{i}_i(t) \\ \big(\Omega_i(t) - \bar{\Omega}_i(t)\big)\sin\bar{i}_i(t) \end{bmatrix}.$$

This set parameterizes the position of the satellite, $\mathbf{x}_i(t)$, in relation to its nominal position, $\bar{\mathbf{x}}_i(t)$. In [26], the satellite orbital dynamics, taking the effect of $J_2$ into account but neglecting the remaining perturbations, are linearized about near-circular nominal orbits. Making use of Floquet theory, system transition and convolution matrices are derived to write the discrete-time LTV system

$$\delta\mathbf{x}_i((k+1)T_c) = \mathbf{A}_i(k)\delta\mathbf{x}_i(kT_c) + \mathbf{B}_i(k)\frac{\mathbf{u}_i(kT_c)}{m_i(kT_c)}$$

with a sampling time $T_c$ and assuming that $\mathbf{u}_i(t)$ and $m_i(t)$ remain constant over each interval $[kT_c; (k+1)T_c[$. The state transition matrix $\mathbf{A}_i(k)$ and the convolution matrix $\mathbf{B}_i(k)$ are given in [26].

In this work, in an attempt to reduce fuel consumption and to follow the communication, computational, and memory constraints detailed in Section 2, a control scheme is devised such that the satellites control their position relative to each other. On one hand, the semi-major axis, eccentricity, and inclination of the orbit of each satellite may be controlled in a decoupled fashion, thus an inertial tracking output component given by

$$\mathbf{z}_{i,in}(k) = \begin{bmatrix} a_i(k) - \bar{a}_i(k) \\ e_{x,i}(k) - \bar{e}_{x,i}(k) \\ e_{y,i}(k) - \bar{e}_{y,i}(k) \\ i_i(k) - \bar{i}_i(k) \end{bmatrix} = \begin{bmatrix} \bar{a}_i(k)\delta a_i(k) \\ \delta e_{x,i}(k) \\ \delta e_{y,i}(k) \\ \delta i_i(k) \end{bmatrix}$$

is considered for each satellite $\mathcal{S}_i$, which is not coupled with any other satellites. On the other hand, to maintain the shape of the constellation, $\delta u_i(k)$ and $\delta\Omega_i(k)$ ought to be controlled in relation to other satellites. It is considered that two satellites are coupled if they are within a tracking range $R$ of each other, i.e., $||\mathbf{p}_i - \mathbf{p}_j|| \leq R$, up to a maximum of $|\mathcal{D}^-|_{\max}$ satellites in $\mathcal{D}_i^-$. If more than $|\mathcal{D}^-|_{\max} - 1$ satellites other than $\mathcal{S}_i$ are within a tracking range of $\mathcal{S}_i$, only the $|\mathcal{D}^-|_{\max} - 1$ closest are considered. Let $\mathcal{D}_i^- \setminus \{i\} = \left\{j_1^i, \ldots, j^i_{|\mathcal{D}_i^-|-1}\right\}$. Then the relative tracking output component is given by

$$\mathbf{z}_{i,rel}(k) := \mathrm{col}\left(\mathbf{z}^{ref}_{i,j_1^i}(k), \ldots, \mathbf{z}^{ref}_{i,j^i_{|\mathcal{D}_i^-|-1}}(k)\right),$$

with

$$\mathbf{z}^{ref}_{i,j}(k) := \begin{bmatrix} u_i(k) - u_j(k) - (\bar{u}_i(k) - \bar{u}_j(k)) \\ \Omega_i(k) - \Omega_j(k) - \big(\bar{\Omega}_i(k) - \bar{\Omega}_j(k)\big) \end{bmatrix}.$$

Defining the tracking output of $\mathcal{S}_i$ as $\mathbf{z}_i(k) := \mathrm{col}(\mathbf{z}_{i,rel}(k), \mathbf{z}_{i,in}(k))$, it can be written as

$$\mathbf{z}_i(k) = \sum_{p \in \mathcal{D}_i^-} \mathbf{H}_{i,p}(k)\delta\mathbf{x}_p(k).$$

### 5.1. Simulation results

The illustrative mega-constellation, inspired in the first shell of the Starlink constellation, is chosen to assess the performance of the method devised in this work. The constellation is a Walker 53.0 deg : 1584/72/17. The parameters that characterize the illustrative constellation are presented in [20]. The maximum in-neighborhood cardinality is set to $|\mathcal{D}^-|_{\max} = 6$. In Fig. 3, a snapshot of the position of each satellite of the constellation over the Earth's spheroid, as well as the tracking output couplings at 0 Dynamical Barycentric Time (TDB) seconds since J2000, are shown. The parameters $H = 100$ and $d = 25$ were chosen. For more details on the tuning of the parameters of the RHC methods see [20].

A realistic nonlinear numerical simulation was performed making use of the high-fidelity TU Delft Astrodynamic Toolbox (TUDAT) [27]. The orbit propagation of the satellites of the constellation accounts for several perturbations: i) Earth's gravity field EGM96 spherical harmonic expansion up to degree and order 24; ii) atmospheric drag NRLMSISE-00 model; iii) cannon ball solar radiation pressure; and iv) third-body perturbations of

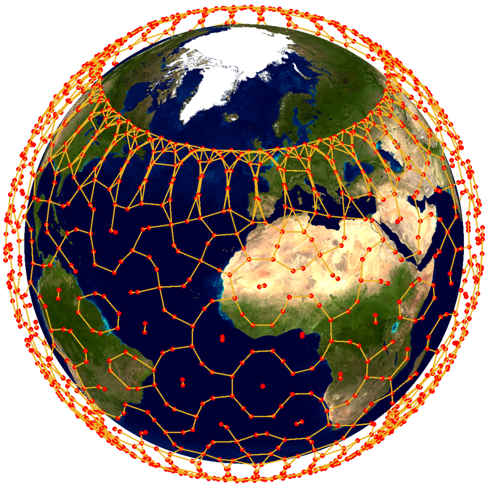


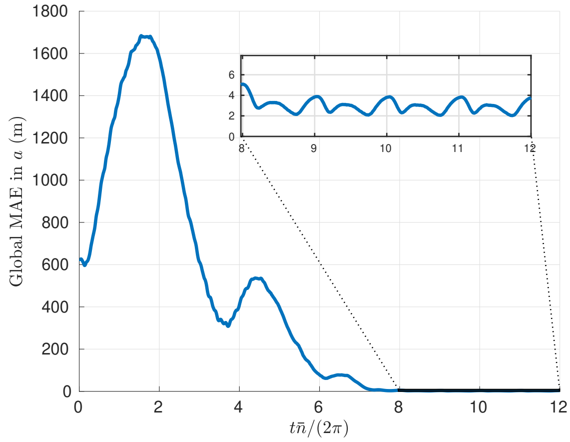
Figure 3: Simulated constellation and tracking output couplings over the Earth at 0 TDB seconds since J2000.

the Sun, Moon, Venus, Mars, and Jupiter. A simulation of the mega-constellation during 12 orbital periods is carried out. An anchor for the nominal constellation is computed at 0 TDB seconds since J2000 and it is not updated during the simulation. The evolution of the mean absolute error (MAE) in the semi-major axis is depicted in Fig. 4(a), where $\bar{n} := \sqrt{\mu/\bar{a}^3}$ and $\mu$ is the standard gravitational parameter of the Earth. To evaluate the performance of the relative tracking between the satellites, the mean argument of latitude error, $e_{u_i}(k)$, and longitude of ascending node error, $e_{\Omega_i}(k)$, are defined for each satellite $i$. Consider an instantaneous hypothetical anchor computed at each time instant $k$. These errors are defined as $e_{u_i}(k) := u_i(k) - \bar{u}_i(k)$ and $e_{\Omega_i}(k) := \Omega_i(k) - \bar{\Omega}_i(k)$, where $\bar{u}_i(k)$ and $\bar{\Omega}_i(k)$ are computed according to (13) making use of the aforementioned instantaneous hypothetical anchor for time instant $k$. It is important to remark that these anchors are only employed for performance assessment purposes in a post-processing step - they are not involved in the control law in any way. The evolution of the MAE of the mean argument of latitude and longitude of ascending node is depicted in Fig. 4(b). The steady-state MAE, obtained by averaging the MAE of the last three orbital periods of the simulation, is depicted in Table 1.

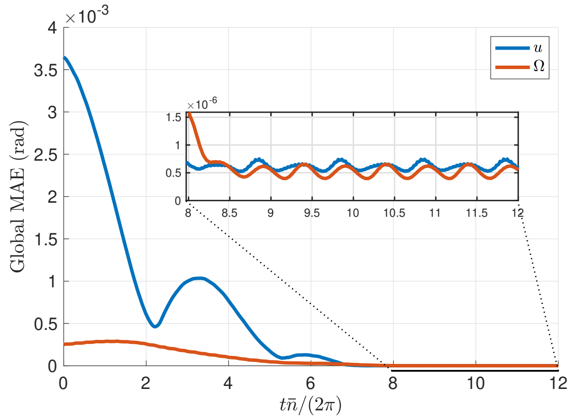
Table 1: Steady-state MAE.

| | $a - \bar{a}$ (m) | $\bar{a}e_{u_i}$ (m) | $\bar{a}e_{\Omega_i}$ (m) |
|---|---|---|---|
| MAE | 2.887 | 4.253 | 3.582 |

First, it is visible that the satellites of the constellation are successfully driven to their nominal semi-major axis and relative separations, despite the large initial errors. Second, although the method




(a) Semi-major axis.




(b) Mean argument of latitude and longitude of ascending node.

Figure 4: Evolution of the MAE.

proposed in this work is designed for LTV systems under very strict communication, computational, and memory limitations, it is able to perform well in a network of systems with nonlinear dynamics. Third, it is visible in Table 1 that this solution reaches meter-level accuracy, not only for the semi-major axis, but also for the relative tracking components.

## 6. Conclusion

Comparing the state-of-the-art solutions with the requirements befitting the very large-scale of emerging tasks, it is clear that the there is a void that needs to be addressed to enable these groundbreaking innovations. First, it was possible to formulate the decentralized control problem in a RHC framework, alongside the implementation feasibility constraints on a very large-scale. Second, a convex relaxation procedure is proposed to approximate the optimal solution of the synthesis problem without full knowledge of the state of the network, which is validated resorting to a large-scale numeric simulation and experimental results. Third, a distributed and decentralized control solution was successfully devised for the particular case of decoupled dynamics, building on the proposed convex relaxation procedure. Its requirements are in line with the large-scale implementation feasibility constraints. Fourth, the distributed decentralized control solution is applied to the orbit control problem of LEO mega-constellations. The control problem is formulated relying on a set of relative orbital elements, which allowed write the shape-keeping task as a coupled sparse tracking output regulation problem. The proposed method shows promising performance for the orbit control problem of a shell of the Starlink mega-constellation.

## References


[1] P. Radoglou-Grammatikis, P. Sarigiannidis, T. Lagkas, I. Moscholios, A compilation of uav applications for precision agriculture, Computer Networks 172 (2020) 107148.

[2] A. Petrovsky, I. Kalinov, P. Karpyshev, D. Tsetserukou, A. Ivanov, A. Golkar, The two-wheeled robotic swarm concept for mars exploration, Acta Astronautica 194 (2022) 1–8.

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

[4] K. Aboudolas, M. Papageorgiou, E. Kosmatopoulos, Store-and-forward based methods for the signal control problem in large-scale congested urban road networks, Transportation Research Part C: Emerging Technologies 17 (2) (2009) 163–174.

[5] V. P. Singh, N. Kishor, P. Samuel, Load frequency control with communication topology changes in smart grid, IEEE Transactions on Industrial Informatics 12 (5) (2016) 1943–1952.

[6] Y. Zhan, P. Wan, C. Jiang, X. Pan, X. Chen, S. Guo, Challenges and solutions for the satellite tracking, telemetry, and command system, IEEE Wireless Communications 27 (6) (2020) 12–18.

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

[8] M. Farhood, Z. Di, G. E. Dullerud, Distributed control of linear time-varying systems interconnected over arbitrary graphs, International Journal of Robust and Nonlinear Control 25 (2) (2015) 179–206.

[9] D. Viegas, P. Batista, P. Oliveira, C. Silvestre, C. P. Chen, Distributed state estimation for linear multi-agent systems with time-varying measurement topology, Automatica 54 (2015) 72–79.

[10] D. Viegas, P. Batista, P. Oliveira, C. Silvestre, Distributed controller design and performance optimization for discrete-time linear systems, Optimal Control Applications and Methods (2020) 1–18doi:10.1002/oca.2669.

[11] L. Lessard, S. Lall, Internal quadratic invariance and decentralized control, in: Proceedings of the 2010 American Control Conference, IEEE, 2010, pp. 5596–5601.

[12] L. Lessard, S. Lall, Convexity of decentralized controller synthesis, IEEE Transactions on Automatic Control 61 (10) (2015) 3122–3127.

[13] P. Chanfreut, J. M. Maestre, E. F. Camacho, A survey on clustering methods for distributed and networked control systems, Annual Reviews in Control 52 (2021) 75–90.

[14] A. Richards, J. How, A decentralized algorithm for robust constrained model predictive control, in: Proceedings of the 2004 American control conference, Vol. 5, IEEE, 2004, pp. 4261–4266.

[15] T. Keviczky, F. Borrelli, G. J. Balas, Decentralized receding horizon control for large scale dynamically decoupled systems, Automatica 42 (12) (2006) 2105–2115.

[16] M. Tillerson, G. Inalhan, J. P. How, Co-ordination and control of distributed spacecraft systems using convex optimization techniques, International Journal of Robust and Nonlinear Control 12 (2-3) (2002) 207–242.

[17] L. Pedroso, P. Batista, Discrete-time decentralized linear quadratic control for linear time-varying systems, International Journal of Robust and Nonlinear Control (2021). doi:10.1002/rnc.5772.

[18] L. Pedroso, P. Batista, Reproducible low-cost flexible quadruple-tank process experimental setup for control educators, practitioners, and researchers, Journal of Process Control 118 (2022) 82–94. doi:10.1016/j.jprocont.2022.08.010.

[19] L. Pedroso, P. Batista, Distributed decentralized receding horizon control for very large-scale networks with application to leo satellite mega-constellations, arXiv. Preprint (2022). doi:10.48550/arXiv.2209.14951.

[20] L. Pedroso, Distributed decentralized control for very large-scale systems with application to leo satellite mega-constellations, Master's thesis, Instituto Superior Técnico, University of Lisbon (2022).

[21] L. Pedroso, P. Batista, Efficient algorithm for the computation of the solution to a sparse matrix equation in distributed control theory, Mathematics 9 (13) (2021). doi:10.3390/math9131497.

[22] B. Anderson, J. B. Moore, Optimal control: linear quadratic methods (1990).

[23] K. H. Johansson, The quadruple-tank process: A multivariable laboratory process with an adjustable zero, IEEE Transactions on Control Systems Technology (2000). doi:10.1109/87.845876.

[24] D. A. Vallado, Fundamentals of Astrodynamics and Applications, McGraw-Hill Companies, Inc. College Custom Series, 1997.

[25] S. D'Amico, Autonomous formation flying in low earth orbit, Ph.D. thesis, TU Delft (2010).

[26] G. Di Mauro, R. Bevilacqua, D. Spiller, J. Sullivan, S. D'Amico, Continuous maneuvers for spacecraft formation flying reconfiguration using relative orbit elements, Acta Astronautica 153 (2018) 311–326.

[27] K. Kumar, P. van Barneveld, D. Dirkx, J. Melman, E. Mooij, R. Noomen, Tudat: a modular and robust astrodynamics toolbox, in: 5th ICATT Conference, ESA, 2012, pp. 1–8.