Received: 23 March 2021 | Accepted: 23 March 2022

DOI: 10.1002/rnc.6130

SPECIAL ISSUE ARTICLE

WILEY

Decentralized control and state estimation of linear time-periodic systems

Ivan Andrushka[1] | **Pedro Batista**[1] | **Paulo Oliveira**[2] | **Carlos Silvestre**[3]

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

[2]IDMEC-Institute of Mechanical Engineering, Instituto Superior Técnico, Universidade de Lisboa, Lisbon, Portugal

[3]Department of Electrical and Computer Engineering, University of Macau, Taipa, Macau

Correspondence
Pedro Batista, Institute for Systems and Robotics, Instituto Superior Técnico, Universidade de Lisboa, Lisbon, Portugal.
Email: pbatista@isr.tecnico.ulisboa.pt

Funding information
Fundação para a Ciência e a Tecnologia, Grant/Award Numbers: DECENTER [LISBOA-01-0145-FEDER-029605], LAETA FCT project UIDB/50022/2020, LARSyS - FCT Project UIDB/50009/2020; Macau Science and Technology Development Fund, Grant/Award Numbers: FDCT/0031/2020/AFJ, FDCT/0146/2019/A3

Abstract

The main contributions of this article are the design of a decentralized controller and state estimator for linear time-periodic systems with fixed network topologies. The proposed method to tackle both problems consists of reformulating the linear periodic dynamics as a linear time-invariant system by applying a time-lifting technique and designing a discrete-time decentralized controller and state estimator for the time-lifted formulation. The problem of designing the decentralized estimator is formulated as a discrete-time Kalman filter subject to sparsity constraints on the gains. Two different algorithms for the computation of steady-state observer gains are tested and compared. The control problem is posed as a state feedback gain optimization problem over an infinite-horizon quadratic cost, subject to a sparsity constraint on the gains. An equivalent formulation that consists in the optimization of the steady-state solution of a matrix difference equation is presented and an algorithm for the computation of the decentralized gain is detailed. Simulation results for the practical cases of the quadruple-tank process and an extended 40-tank process are presented that illustrate the performance of the proposed solutions, complemented with numerical simulations using the Monte Carlo method.

KEYWORDS

decentralized control, decentralized estimation, linear time-periodic, sparse gains, time-lifting

1 | INTRODUCTION

The study of linear time-periodic (LTP) systems[1] is of paramount importance since many systems cannot be accurately described by linear time-invariant (LTI) dynamics. This can be seen, for example, in rotating systems[2,3] or multirate systems,[4] where sensors have different sampling rates. This class of systems is also quite significant in the study of nonlinear systems since linearizing nonlinear dynamics about periodic state trajectories leads to LTP systems. A classic example of this is the dynamics of a satellite when linearized about its orbit.[5]

While many systems can be modeled by LTP dynamics, the literature available for this specific type of system in what concerns state estimation is limited. Some centralized observers for LTP systems have been studied, such as in References 1 and 6. However, this centralized approach may lead to robustness problems due to the reliance on a central processing node. Besides, as networks grow larger, communication and computational loads impose severe bottlenecks. Decentralized observers, while more difficult to design and analyze, provide a solution to the problems of robustness and scalability.

The topic of decentralized state estimation has received much attention. However, most solutions focus on LTI systems or nonlinear systems. Nonetheless, many interesting solutions have been proposed following the Kalman filtering approach, ranging from the more classic hierarchical structures[7] to fully decentralized approaches, such as the ones found in References 8-11. For nonlinear filters, the most common solution comes in the form of extended Kalman filters and some valuable contributions can be seen in References 12 and 13. Other decentralized solutions, which do not follow the Kalman filtering scheme, have also been proposed, such as particle filters[12] and H_∞ filters.[14,15] However, these approaches lead to computationally heavy algorithms.

In contrast to the problem of decentralized state estimation for LTP systems, decentralized control[16,17] is a well-established area, in which a more significant body of research is available. A well-known approach to this problem, and the one used in this article, is the design of periodic controllers based on a time-invariant reformulation of the periodic dynamics, known as time-lifting. Some examples of this, designed specifically with multi-rate systems in mind, can be seen in References 18 and 19. Another valuable contribution to this topic can be seen in Reference 20, where the authors provide a method of synthesizing periodic controllers for generic LTP systems. Nevertheless, other solutions that do not make use of the lifted dynamics have been proposed. A notable method is the use of model predictive control, some examples of which can be seen in References 21 and 22. Some other interesting solutions can be seen in Reference 23, where the authors use linear matrix inequalities to solve the problem with guaranteed H_∞ performance, and in Reference 24, where a consensus approach is taken. Notwithstanding, it is important to stress that the problems of decentralized control and estimation are extremely difficult, see Reference 25 and references therein, and remain open problems. In fact, even for very simple linear systems, the optimal solution may be nonlinear.[26] Moreover, the distributed control problem has been shown to be convex if and only if quadratic invariance of the controller set is ensured, see Reference 27.

This article addresses the problems of decentralized control and state estimation for LTP systems with limited communication between each component of the system. The topology of the system is fixed and it is modeled using a sparse matrix that describes the availability of communication channels. The proposed method makes use of the well-known relationship between LTP and LTI systems, namely the fact that a LTP system is invariant when described over a full period. Alluding to this property, it is possible to reformulate a LTP system as a LTI system, such as described in References 28 and 29. The technique adopted in this article is called time-lifting, where the system dynamics over one period are condensed into augmented vectors and matrices that describe the dynamics of the LTP system in a time-invariant fashion. The advantage of using this approach is that it allows the use of algorithms that have been previously developed specifically for LTI systems. The algorithms used to design the observer are the one-step and the finite-horizon methods, introduced in Reference 9, whereas the controller is designed using the one-step method described in Reference 30. The one-step and finite-horizon algorithms for state estimation solve the problem of Kalman filtering subject to sparsity constraints by approximating the steady-state behavior, which can be solved in closed-form, leading to computationally efficient and well-performing solutions. Note, however, that the proposed solution is suboptimal, since the one-step method provides the optimal solution for a convex relaxation of the original optimization problem, not the actual problem. The one-step method for control seeks to obtain well-performing steady-state gains by optimizing the solution of a matrix difference equation, which is obtained by reformulating the infinite-horizon quadratic cost function. Likewise, the control solution proposed in this article is also suboptimal. The goodness of the proposed solutions is supported by simulation results for two cases: a randomly generated system, and a system of four interconnected water tanks, known as the quadruple-tank process[31] (QTP), operating about periodic state trajectories. For both cases, Monte Carlo runs are used to evaluate the performance of the estimator. The simulation results are further expanded for a system of N interconnected water tanks, henceforth known as the N-tank process. Notably, the system of interconnected tanks, operating about periodic state trajectories, is an example of a nonlinear system that can be tackled resorting to the class of LTP system when its dynamics are linearized about the periodic state trajectories.

The article is organized as follows. Section 2 formally introduces the problems to be solved. Section 3 describes the time-lifting procedure. In Section 4, the one-step and finite-horizon methods for computing the observer gains for a LTP system are introduced, and a simple example using a randomly generated system is showcased. Section 5 describes the one-step method for computing steady-state feedback control gains. Section 6 details the application of this framework to the QTP, with simulation results to assess the performance. Section 7 extends the simulation results for the N-tank process. Finally, Section 8 summarizes the main results and conclusions of the article.

1.1 | Notation

A block diagonal matrix $\mathbf{A}$ with n blocks $\mathbf{A}_1, \mathbf{A}_2, \ldots, \mathbf{A}_n$ is denoted by $\mathbf{A} = \text{diag}(\mathbf{A}_1, \mathbf{A}_2, \ldots, \mathbf{A}_n)$. The transpose of a matrix or vector is denoted by $(.)^T$. For simplicity of notation, a signal at step k is defined as $\mathbf{v}(k) = \mathbf{v}_k$. The trace of a matrix $\mathbf{A}$ is denoted by $\text{tr}(\mathbf{A})$.

2 | PROBLEM STATEMENT

Let $T > 1 \in \mathbb{N}$ be the period of a LTP system. The dynamics of said system can be compactly described in state-space form by

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{A}_k \mathbf{x}_k + \mathbf{B}_k \mathbf{u}_k + \mathbf{w}_k, \\ \mathbf{y}_k = \mathbf{C}_k \mathbf{x}_k + \mathbf{v}_k, \end{cases} \tag{1}$$

where $\mathbf{x}_k \in \mathbb{R}^n$ is the state, $\mathbf{u}_k \in \mathbb{R}^m$ is the input, $\mathbf{y}_k \in \mathbb{R}^p$ is the output, and $\mathbf{w}_k \in \mathbb{R}^n$ and $\mathbf{v}_k \in \mathbb{R}^p$ are the process and observation noises, respectively. These are assumed to be independent zero-mean white Gaussian processes with associated covariance matrices $\mathbf{Q} \succcurlyeq 0$ and $\mathbf{R} \succ 0$, respectively. $\mathbf{A}_k \in \mathbb{R}^{n \times n}$, $\mathbf{B}_k \in \mathbb{R}^{n \times m}$, and $\mathbf{C}_k \in \mathbb{R}^{p \times n}$ are time-varying periodic matrices, such that $\mathbf{A}_{k+T} = \mathbf{A}_k$, $\mathbf{B}_{k+T} = \mathbf{B}_k$, and $\mathbf{C}_{k+T} = \mathbf{C}_k$ for all $k = 0, 1, 2, \ldots$

2.1 | Decentralized estimation of LTP systems

Consider the sparsity pattern $\mathbf{E}$ and denote the Kalman filter gain by $\mathbf{K}_k$. The set of gains that follow this sparsity pattern is defined by

$$\text{Sparse}(\mathbf{E}) = \{\mathbf{K}_k \in \mathbb{R}^{n \times p} : [\mathbf{E}]_{ij} = 0 \Rightarrow [\mathbf{K}_k]_{ij} = 0, \ i = 1, \ldots, n, j = 1, \ldots, p\}.$$

Like in the centralized Kalman filter, the state estimate is obtained through prediction and filtering. The prediction step is given by

$$\hat{\mathbf{x}}_{k+1|k} = \mathbf{A}_k \hat{\mathbf{x}}_{k|k} + \mathbf{B}_k \mathbf{u}_k, \tag{2}$$

where $\hat{\mathbf{x}}_{k+1|k}$ denotes the predicted estimate at step $k+1$. The predicted error covariance $\mathbf{P}_{k+1|k} = \mathbb{E}\{(\mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1|k})(\mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1|k})^T\} \succcurlyeq \mathbf{0}$ at step $k+1$ is given by

$$\mathbf{P}_{k+1|k} = \mathbf{A}_k \mathbf{P}_{k|k} \mathbf{A}_k^T + \mathbf{Q}. \tag{3}$$

The filtered estimate $\hat{\mathbf{x}}_{k+1|k+1}$ is given by

$$\hat{\mathbf{x}}_{k+1|k+1} = \hat{\mathbf{x}}_{k+1|k} + \mathbf{K}_{k+1}(\mathbf{y}_{k+1} - \mathbf{C}_{k+1} \hat{\mathbf{x}}_{k+1|k}) \tag{4}$$

and the updated error covariance $\mathbf{P}_{k+1|k+1} = \mathbb{E}\{(\mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1|k+1})(\mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1|k+1})^T\} \succcurlyeq \mathbf{0}$ follows

$$\mathbf{P}_{k+1|k+1} = \mathbf{K}_{k+1} \mathbf{R} \mathbf{K}_{k+1}^T + (\mathbf{I} - \mathbf{K}_{k+1} \mathbf{C}_{k+1}) \mathbf{P}_{k+1|k} (\mathbf{I} - \mathbf{K}_{k+1} \mathbf{C}_{k+1})^T. \tag{5}$$

The problem of designing a Kalman filter subject to sparsity constraints for a finite-time window W can be formulated as follows. Given a sparsity pattern $\mathbf{E}$ and an initial state estimate $\hat{\mathbf{x}}_{0|0}$, with error covariance $\mathbf{P}_{0|0} \succcurlyeq \mathbf{0}$, solve the optimization problem

$$\begin{aligned} \underset{\mathbf{K}_i, i=1,\ldots,W}{\text{minimize}} \quad & \sum_{k=1}^{W} \text{tr}(\mathbf{P}_{k|k}), \\ \text{subject to} \quad & \mathbf{K}_i \in \text{Sparse}(\mathbf{E}), \quad i = 1, 2, \ldots, W. \end{aligned} \tag{6}$$

Notice that W can be made arbitrarily large, as long as it is a multiple of the period T, and its choice depends on the system model. More complex systems require a larger window size to achieve steady-state performance. Throughout the article, the following assumption is made.

Assumption 1. The decentralized state observer is stabilizable, that is, there exists a set of gains $\mathbf{K}_i \in \text{Sparse}(\mathbf{E}), i = 1, \dots, T$, such that, in the absence of noise, the state estimate $\hat{\mathbf{x}}_{k|k}$ converges globally exponentially fast to the state $\mathbf{x}_k$.

2.2 | Decentralized control of LTP systems

The decentralized control problem for LTP systems can be described as follows. Consider the LTP system

$$\mathbf{x}_{k+1} = \mathbf{A}_k \mathbf{x}_k + \mathbf{B}_k \mathbf{u}_k \tag{7}$$

and the set of controller gains $\mathbf{K}_k \in \mathbb{R}^{m \times n}$, which follows the sparsity pattern $\mathbf{E}_c \in \mathbb{R}^{m \times n}$ defined by

$$\text{Sparse}(\mathbf{E}_c) = \{\mathbf{K}_k \in \mathbb{R}^{m \times n} : [\mathbf{E}_c]_{ij} = 0 \Rightarrow [\mathbf{K}_k]_{ij} = 0, \ i = 1, \dots, m, j = 1, \dots, n\}.$$

The goal is to compute a feedback control law for (7), subject to a sparsity constraint, of the form

$$\mathbf{u}_k = -\mathbf{K}_k \mathbf{x}_k, \quad \mathbf{K}_k \in \text{Sparse}(\mathbf{E}_c), \tag{8}$$

which minimizes the infinite-horizon quadratic cost function

$$J = \sum_{k=0}^{+\infty} (\mathbf{x}_k^T \mathbf{Q} \mathbf{x}_k + \mathbf{u}_k^T \mathbf{R} \mathbf{u}_k), \tag{9}$$

where $\mathbf{Q} \succcurlyeq 0$ and $\mathbf{R} \succ 0$. The stabilization problem is not studied in this article, as it has been addressed in other works, such as in References 19 and 20. In the same vein, controllability (and observability) analysis for LTP systems can be found in Reference 1. In this article, the following assumption is made.

Assumption 2. There exists a set of stabilizing decentralized gains for (7), that is, there exists a set of gains $\mathbf{K}_i \in \text{Sparse}(\mathbf{E}_c), i = 1, \dots, T$, such that the eigenvalues of the closed-loop matrix $(\mathbf{A}_k - \mathbf{B}_k \mathbf{K}_k)$ lie inside the unit circle.

3 | TIME INVARIANT REFORMULATION

The one-step and finite-horizon methods for computing the observer gains and the one-step method for computing controller gains proposed in References 9 and 30, respectively, apply to LTI systems. In the approach proposed in this article to tackle LTP systems, these are first reformulated as LTI systems. To that end, a technique known as time-lifting, based on the work done in Reference 29, is employed. Consider the LTP system

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{A}_k \mathbf{x}_k + \mathbf{B}_k \mathbf{u}_k + \mathbf{w}_k, \\ \mathbf{y}_k = \mathbf{C}_k \mathbf{x}_k + \mathbf{v}_k. \end{cases} \tag{10}$$

To simplify the presentation, consider that the period of this system is $T = 3$. The state equation can be expanded as follows

$$\begin{aligned} \mathbf{x}_1 &= \mathbf{A}_0 \mathbf{x}_0 + \mathbf{B}_0 \mathbf{u}_0 + \mathbf{w}_0, \\ \mathbf{x}_2 &= \mathbf{A}_1 \mathbf{x}_1 + \mathbf{B}_1 \mathbf{u}_1 + \mathbf{w}_1, \\ \mathbf{x}_3 &= \mathbf{A}_2 \mathbf{x}_2 + \mathbf{B}_2 \mathbf{u}_2 + \mathbf{w}_2, \\ \mathbf{x}_4 &= \mathbf{A}_0 \mathbf{x}_3 + \mathbf{B}_0 \mathbf{u}_3 + \mathbf{w}_3, \\ \mathbf{x}_5 &= \mathbf{A}_1 \mathbf{x}_4 + \mathbf{B}_1 \mathbf{u}_4 + \mathbf{w}_4, \\ \mathbf{x}_6 &= \mathbf{A}_2 \mathbf{x}_5 + \mathbf{B}_2 \mathbf{u}_5 + \mathbf{w}_5, \end{aligned}$$

and so forth. The periodic system can be regrouped in matrix form as

$$\begin{bmatrix}\mathbf{x}_1\\ \mathbf{x}_2\\ \mathbf{x}_3\end{bmatrix}=\begin{bmatrix}\mathbf{0} & \mathbf{0} & \mathbf{A}_0\\ \mathbf{0} & \mathbf{0} & \mathbf{A}_1\mathbf{A}_0\\ \mathbf{0} & \mathbf{0} & \mathbf{A}_2\mathbf{A}_1\mathbf{A}_0\end{bmatrix}\begin{bmatrix}\mathbf{0}\\ \mathbf{0}\\ \mathbf{x}_0\end{bmatrix}+\begin{bmatrix}\mathbf{B}_0 & \mathbf{0} & \mathbf{0}\\ \mathbf{A}_1\mathbf{B}_0 & \mathbf{B}_1 & \mathbf{0}\\ \mathbf{A}_2\mathbf{A}_1\mathbf{B}_0 & \mathbf{A}_2\mathbf{B}_1 & \mathbf{B}_2\end{bmatrix}\begin{bmatrix}\mathbf{u}_0\\ \mathbf{u}_1\\ \mathbf{u}_2\end{bmatrix}+\begin{bmatrix}\mathbf{I} & \mathbf{0} & \mathbf{0}\\ \mathbf{A}_1 & \mathbf{I} & \mathbf{0}\\ \mathbf{A}_2\mathbf{A}_1 & \mathbf{A}_2 & \mathbf{I}\end{bmatrix}\begin{bmatrix}\mathbf{w}_0\\ \mathbf{w}_1\\ \mathbf{w}_2\end{bmatrix},$$
$$\begin{bmatrix}\mathbf{x}_4\\ \mathbf{x}_5\\ \mathbf{x}_6\end{bmatrix}=\begin{bmatrix}\mathbf{0} & \mathbf{0} & \mathbf{A}_0\\ \mathbf{0} & \mathbf{0} & \mathbf{A}_1\mathbf{A}_0\\ \mathbf{0} & \mathbf{0} & \mathbf{A}_2\mathbf{A}_1\mathbf{A}_0\end{bmatrix}\begin{bmatrix}\mathbf{x}_1\\ \mathbf{x}_2\\ \mathbf{x}_3\end{bmatrix}+\begin{bmatrix}\mathbf{B}_0 & \mathbf{0} & \mathbf{0}\\ \mathbf{A}_1\mathbf{B}_0 & \mathbf{B}_1 & \mathbf{0}\\ \mathbf{A}_2\mathbf{A}_1\mathbf{B}_0 & \mathbf{A}_2\mathbf{B}_1 & \mathbf{B}_2\end{bmatrix}\begin{bmatrix}\mathbf{u}_3\\ \mathbf{u}_4\\ \mathbf{u}_5\end{bmatrix}+\begin{bmatrix}\mathbf{I} & \mathbf{0} & \mathbf{0}\\ \mathbf{A}_1 & \mathbf{I} & \mathbf{0}\\ \mathbf{A}_2\mathbf{A}_1 & \mathbf{A}_2 & \mathbf{I}\end{bmatrix}\begin{bmatrix}\mathbf{w}_3\\ \mathbf{w}_4\\ \mathbf{w}_5\end{bmatrix},$$

and so forth. The associated output is given by

$$\begin{bmatrix}\mathbf{0}\\ \mathbf{0}\\ \mathbf{y}_0\end{bmatrix}=\begin{bmatrix}\mathbf{C}_1 & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{C}_2 & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{C}_0\end{bmatrix}\begin{bmatrix}\mathbf{0}\\ \mathbf{0}\\ \mathbf{x}_0\end{bmatrix}+\begin{bmatrix}\mathbf{0}\\ \mathbf{0}\\ \mathbf{v}_0\end{bmatrix},\quad\begin{bmatrix}\mathbf{y}_1\\ \mathbf{y}_2\\ \mathbf{y}_3\end{bmatrix}=\begin{bmatrix}\mathbf{C}_1 & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{C}_2 & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{C}_0\end{bmatrix}\begin{bmatrix}\mathbf{x}_1\\ \mathbf{x}_2\\ \mathbf{x}_3\end{bmatrix}+\begin{bmatrix}\mathbf{v}_1\\ \mathbf{v}_2\\ \mathbf{v}_3\end{bmatrix},$$
$$\begin{bmatrix}\mathbf{y}_4\\ \mathbf{y}_5\\ \mathbf{y}_6\end{bmatrix}=\begin{bmatrix}\mathbf{C}_1 & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{C}_2 & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{C}_0\end{bmatrix}\begin{bmatrix}\mathbf{x}_4\\ \mathbf{x}_5\\ \mathbf{x}_6\end{bmatrix}+\begin{bmatrix}\mathbf{v}_4\\ \mathbf{v}_5\\ \mathbf{v}_6\end{bmatrix}.$$

In general, for $k \geq 0$,

$$\begin{bmatrix}\mathbf{x}_{kT+1}\\ \mathbf{x}_{kT+2}\\ \mathbf{x}_{kT+3}\end{bmatrix}=\begin{bmatrix}\mathbf{0} & \mathbf{0} & \mathbf{A}_0\\ \mathbf{0} & \mathbf{0} & \mathbf{A}_1\mathbf{A}_0\\ \mathbf{0} & \mathbf{0} & \mathbf{A}_2\mathbf{A}_1\mathbf{A}_0\end{bmatrix}\begin{bmatrix}\mathbf{x}_{(k-1)T+1}\\ \mathbf{x}_{(k-1)T+2}\\ \mathbf{x}_{(k-1)T+3}\end{bmatrix}+\begin{bmatrix}\mathbf{B}_0 & \mathbf{0} & \mathbf{0}\\ \mathbf{A}_1\mathbf{B}_0 & \mathbf{B}_1 & \mathbf{0}\\ \mathbf{A}_2\mathbf{A}_1\mathbf{B}_0 & \mathbf{A}_2\mathbf{B}_1 & \mathbf{B}_2\end{bmatrix}\begin{bmatrix}\mathbf{u}_{kT}\\ \mathbf{u}_{kT+1}\\ \mathbf{u}_{kT+2}\end{bmatrix}+\begin{bmatrix}\mathbf{I} & \mathbf{0} & \mathbf{0}\\ \mathbf{A}_1 & \mathbf{I} & \mathbf{0}\\ \mathbf{A}_2\mathbf{A}_1 & \mathbf{A}_2 & \mathbf{I}\end{bmatrix}\begin{bmatrix}\mathbf{w}_{kT}\\ \mathbf{w}_{kT+1}\\ \mathbf{w}_{kT+2}\end{bmatrix},$$
$$\begin{bmatrix}\mathbf{y}_{kT+1}\\ \mathbf{y}_{kT+2}\\ \mathbf{y}_{kT+3}\end{bmatrix}=\begin{bmatrix}\mathbf{C}_1 & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{C}_2 & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{C}_0\end{bmatrix}\begin{bmatrix}\mathbf{x}_{kT+1}\\ \mathbf{x}_{kT+2}\\ \mathbf{x}_{kT+3}\end{bmatrix}+\begin{bmatrix}\mathbf{v}_{kT+1}\\ \mathbf{v}_{kT+2}\\ \mathbf{v}_{kT+3}\end{bmatrix}.$$

This result can be extended to a LTP system with period T. Define the stacked quantities

$$\overline{\mathbf{x}}_k=\begin{bmatrix}\mathbf{x}_{(k-1)T+1}\\ \mathbf{x}_{(k-1)T+2}\\ \vdots\\ \mathbf{x}_{(k-1)T+T}\end{bmatrix},\ \overline{\mathbf{x}}_0=\begin{bmatrix}\mathbf{0}\\ \mathbf{0}\\ \vdots\\ \mathbf{x}_0\end{bmatrix},\overline{\mathbf{u}}_k=\begin{bmatrix}\mathbf{u}_{kT}\\ \mathbf{u}_{kT+1}\\ \vdots\\ \mathbf{u}_{kT+T-1}\end{bmatrix},\ \overline{\mathbf{y}}_k=\begin{bmatrix}\mathbf{y}_{(k-1)T+1}\\ \mathbf{y}_{(k-1)T+2}\\ \vdots\\ \mathbf{y}_{(k-1)T+T}\end{bmatrix},\overline{\mathbf{w}}_k=\begin{bmatrix}\mathbf{w}_{kT}\\ \mathbf{w}_{kT+1}\\ \vdots\\ \mathbf{w}_{kT+T-1}\end{bmatrix},\ \overline{\mathbf{v}}_k=\begin{bmatrix}\mathbf{v}_{(k-1)T+1}\\ \mathbf{v}_{(k-1)T+2}\\ \vdots\\ \mathbf{v}_{(k-1)T+T}\end{bmatrix},$$
$$\overline{\mathbf{A}}=\begin{bmatrix}\mathbf{0} & \dots & \mathbf{0} & \mathbf{A}_0\\ \mathbf{0} & \dots & \mathbf{0} & \mathbf{A}_1\mathbf{A}_0\\ \vdots & & \vdots & \vdots\\ \mathbf{0} & \dots & \mathbf{0} & \mathbf{A}_{T-1}\dots\mathbf{A}_1\mathbf{A}_0\end{bmatrix},\ \overline{\mathbf{B}}=\begin{bmatrix}\mathbf{B}_0 & \mathbf{0} & \dots & \dots & \mathbf{0}\\ \mathbf{A}_1\mathbf{B}_0 & \mathbf{B}_1 & \ddots & & \vdots\\ \mathbf{A}_2\mathbf{A}_1\mathbf{B}_0 & \mathbf{A}_2\mathbf{B}_1 & \ddots & \ddots & \vdots\\ \vdots & \vdots & \ddots & \mathbf{B}_{T-2} & \mathbf{0}\\ \mathbf{A}_{T-1}\dots\mathbf{A}_1 & \mathbf{A}_{T-1}\dots\mathbf{A}_2 & \dots & \mathbf{A}_{T-1}\mathbf{B}_{T-2} & \mathbf{B}_{T-1}\end{bmatrix},$$
$$\overline{\mathbf{C}}=\operatorname{diag}(\mathbf{C}_1,\ \mathbf{C}_2,\ \dots,\ \mathbf{C}_{T-1},\ \mathbf{C}_0),\ \overline{\mathbf{G}}=\begin{bmatrix}\mathbf{I} & \mathbf{0} & \dots & \dots & \mathbf{0}\\ \mathbf{A}_1 & \mathbf{I} & \ddots & & \vdots\\ \mathbf{A}_2\mathbf{A}_1 & \mathbf{A}_2 & \ddots & \ddots & \vdots\\ \vdots & \vdots & \ddots & \mathbf{I} & \mathbf{0}\\ \mathbf{A}_{T-1}\dots\mathbf{A}_1 & \mathbf{A}_{T-1}\dots\mathbf{A}_2 & \dots & \mathbf{A}_{T-1} & \mathbf{I}\end{bmatrix}. \tag{11}$$

The lifted system can then be compactly written as

$$\begin{cases} \overline{\mathbf{x}}_{k+1} = \overline{\mathbf{A}}\overline{\mathbf{x}}_k + \overline{\mathbf{B}}\overline{\mathbf{u}}_k + \overline{\mathbf{G}}\overline{\mathbf{w}}_k, \\ \overline{\mathbf{y}}_k = \overline{\mathbf{C}}\overline{\mathbf{x}}_k + \overline{\mathbf{v}}_k, \end{cases} \tag{12}$$

where $\overline{\mathbf{x}}_k \in \mathbb{R}^{nT}$, $\overline{\mathbf{u}}_k \in \mathbb{R}^{mT}$, and $\overline{\mathbf{y}}_k \in \mathbb{R}^{pT}$ are the stacked state, input, and output, respectively. The vectors $\overline{\mathbf{w}}_k \in \mathbb{R}^{nT}$ and $\overline{\mathbf{v}}_k \in \mathbb{R}^{pT}$ represent the stacked process and measurement noises with associated covariance matrices given by $\overline{\mathbf{Q}} = \text{diag}(\mathbf{Q}, \dots, \mathbf{Q}) \in \mathbb{R}^{nT\times nT}$ and $\overline{\mathbf{R}} = \text{diag}(\mathbf{R}, \dots, \mathbf{R}) \in \mathbb{R}^{pT\times pT}$, respectively. $\overline{\mathbf{A}} \in \mathbb{R}^{nT\times nT}$, $\overline{\mathbf{B}} \in \mathbb{R}^{nT\times mT}$, and $\overline{\mathbf{C}} \in \mathbb{R}^{pT\times nT}$ are the invariant system matrices and $\overline{\mathbf{G}} \in \mathbb{R}^{nT\times nT}$ encodes the effect of the process noise on the system dynamics.

4 | OBSERVER GAINS

To compute the observer gains, two algorithms are used. First, the one-step algorithm, which solves the decentralized Kalman filter (DKF) in closed-form and works by propagating the Kalman filter equations. The second algorithm is the finite-horizon, which solves the DKF by iterating over a finite-time window, fixing all of the gains except for one single $\mathbf{K}_k$ in each iteration, and solving a partial optimization problem.

4.1 | Time-lifted Kalman filter

The design of the decentralized LTP Kalman filter is posed as an optimization problem, as presented in Section 2. The approach followed in this article is to compute the observer gains by solving a lifted variation of problem (6), which allows to consider a LTI system, obtained by time-lifting a LTP system, and applying the resulting gains to the periodic filter (2) and (4).

Consider the time-lifted system (12). The Kalman filter equations for the estimate and error covariance (2)–(5) can be rewritten, for the time-lifted invariant system, as

$$\hat{\overline{\mathbf{x}}}_{k+1|k} = \overline{\mathbf{A}}\hat{\overline{\mathbf{x}}}_{k|k} + \overline{\mathbf{B}}\overline{\mathbf{u}}_k, \tag{13}$$

$$\overline{\mathbf{P}}_{k+1|k} = \overline{\mathbf{A}\mathbf{P}}_{k|k}\overline{\mathbf{A}}^T + \overline{\mathbf{G}\mathbf{Q}\mathbf{G}}^T, \tag{14}$$

$$\hat{\overline{\mathbf{x}}}_{k+1|k+1} = \hat{\overline{\mathbf{x}}}_{k+1|k+1} + \overline{\mathbf{K}}_{k+1}(\overline{\mathbf{y}}_{k+1} - \overline{\mathbf{C}}\hat{\overline{\mathbf{x}}}_{k+1|k}), \tag{15}$$

$$\overline{\mathbf{P}}_{k+1|k+1} = \overline{\mathbf{K}}_{k+1}\overline{\mathbf{R}}\overline{\mathbf{K}}_{k+1}^T + (\mathbf{I} - \overline{\mathbf{K}}_{k+1}\overline{\mathbf{C}})\overline{\mathbf{P}}_{k+1|k}(\mathbf{I} - \overline{\mathbf{K}}_{k+1}\overline{\mathbf{C}})^T, \tag{16}$$

and the optimization problem (6) can now be reformulated as

$$\begin{aligned} \underset{\overline{\mathbf{K}}_i, i=1,\dots,W}{\text{minimize}} \quad & \sum_{k=1}^{W} \text{tr}(\overline{\mathbf{P}}_{k|k}), \\ \text{subject to} \quad & \overline{\mathbf{K}}_i \in \text{Sparse}(\overline{\mathbf{E}}), \quad i = 1, 2, \dots, W, \end{aligned} \tag{17}$$

where $\overline{\mathbf{E}}$ follows a block diagonal structure, $\overline{\mathbf{E}} = \text{diag}(\mathbf{E}, \dots, \mathbf{E}) \in \mathbb{R}^{nT\times pT}$.

Solving problem (17) is not exactly equivalent to solving problem (6) due to the way $\overline{\mathbf{P}}_{k|k}$ is propagated but serves as a good approximation. Indeed, consider for instance a system with period $T = 2$. The predicted error covariance (14) can be expanded as

$$\begin{aligned} \begin{bmatrix} \mathbf{P}_{1|0} \\ \mathbf{P}_{2|1} \end{bmatrix} &= \begin{bmatrix} \mathbf{0} & \mathbf{A}_0 \\ \mathbf{0} & \mathbf{A}_1\mathbf{A}_0 \end{bmatrix} \begin{bmatrix} \mathbf{0} \\ \mathbf{P}_{0|0} \end{bmatrix} \begin{bmatrix} \mathbf{0} & \mathbf{A}_0 \\ \mathbf{0} & \mathbf{A}_1\mathbf{A}_0 \end{bmatrix}^T + \overline{\mathbf{G}\mathbf{Q}\mathbf{G}}^T \\ &= \begin{bmatrix} \mathbf{A}_0\mathbf{P}_{0|0}\mathbf{A}_0^T & \mathbf{A}_0\mathbf{P}_{0|0}(\mathbf{A}_1\mathbf{A}_0)^T \\ (\mathbf{A}_1\mathbf{A}_0)\mathbf{P}_{0|0}\mathbf{A}_0^T & (\mathbf{A}_1\mathbf{A}_0)\mathbf{P}_{0|0}(\mathbf{A}_1\mathbf{A}_0)^T \end{bmatrix} + \begin{bmatrix} \mathbf{Q} & \mathbf{Q}\mathbf{A}_1^T \\ \mathbf{A}_1\mathbf{Q} & \mathbf{A}_1\mathbf{Q}\mathbf{A}_1^T + \mathbf{Q} \end{bmatrix} \end{aligned}$$

and the updated error covariance (16) is given by

$$\begin{bmatrix}\mathbf{P}_{1|1}\\ \mathbf{P}_{2|2}\end{bmatrix} = \overline{\mathbf{K}}_1\overline{\mathbf{R}}\overline{\mathbf{K}}_1 + (\mathbf{I} - \overline{\mathbf{K}}_1\overline{\mathbf{C}})\begin{bmatrix}\mathbf{P}_{1|0}\\ \mathbf{P}_{2|1}\end{bmatrix}(\mathbf{I} - \overline{\mathbf{K}}_1\overline{\mathbf{C}})^T.$$

The nondiagonal terms of the predicted covariance are calculated with inconsistent dynamics matrices. However, due to the sparsity constraint on the gain in the optimization problem, which follows a block diagonal structure, the nondiagonal blocks are disregarded in the gain computation. Thus, the relevant terms are given by

$$\begin{cases}\mathbf{P}_{1|0} = \mathbf{A}_0\mathbf{P}_{0|0}\mathbf{A}_0^T + \mathbf{Q},\\ \mathbf{P}_{2|1} = \mathbf{A}_1\mathbf{A}_0\mathbf{P}_{0|0}(\mathbf{A}_1\mathbf{A}_0)^T + \mathbf{A}_1\mathbf{Q}\mathbf{A}_1^T + \mathbf{Q}\\ \qquad = \mathbf{A}_1\mathbf{P}_{1|0}\mathbf{A}_1^T + \mathbf{Q}.\end{cases}$$

Compare this to the expressions obtained by expanding the periodic error covariance (3) and (5)

$$\begin{aligned}\mathbf{P}_{1|0} &= \mathbf{A}_0\mathbf{P}_{0|0}\mathbf{A}_0^T + \mathbf{Q},\\ \mathbf{P}_{1|1} &= \mathbf{K}_1\mathbf{R}\mathbf{K}_1 + (\mathbf{I} - \mathbf{K}_1\mathbf{C}_1)(\mathbf{A}_0\mathbf{P}_{0|0}\mathbf{A}_0^T + \mathbf{Q})(\mathbf{I} - \mathbf{K}_1\mathbf{C}_1)^T,\\ \mathbf{P}_{2|1} &= \mathbf{A}_1\mathbf{P}_{1|1}\mathbf{A}_1^T + \mathbf{Q}.\end{aligned}$$

In the periodic filter, the predicted covariance $\mathbf{P}_{2|1}$ is computed using the filtered covariance $\mathbf{P}_{1|1}$. On the other hand, the time-lifted filter computes $\mathbf{P}_{2|1}$ as a function of $\mathbf{P}_{1|0}$. In essence, while the periodic filter computes the covariance step by step, in the augmented filter T estimates are predicted and then filtered. Consequently, for periodic systems with a large period T, this method is farther from optimality. But, for systems where T is not as sizable, the resulting filter performs very well, as it will be shown in the simulation results.

4.2 | One-step method

For the one-step method, the strategy is to minimize the error covariance at each step, as in the classical Kalman filter. This can be done for the time-lifted system by solving the optimization problem

$$\begin{aligned}\underset{\overline{\mathbf{K}}_{k+1}}{\text{minimize}} \quad & \operatorname{tr}(\overline{\mathbf{P}}_{k+1|k+1}),\\ \text{subject to} \quad & \overline{\mathbf{K}}_{k+1} \in \operatorname{Sparse}(\overline{\mathbf{E}}).\end{aligned} \tag{18}$$

Notice that all the matrices involved in the optimal problem (18) are constant. Hence, the optimal solution of this problem, detailed and derived in Reference 9, can be applied here. The theorem describing the solution, specifically adapted for time-lifted system, is presented next. Notice that the difference lies on the system matrices, which correspond to the time-lifted formulation that was previously introduced.

Theorem 1. *Consider a sparsity pattern $\overline{\mathbf{E}}$. Let $\mathbf{l}_i$ denote a column vector, whose entries are all set to zero except for the ith one, which is set to 1, and define $\boldsymbol{\mathcal{L}}_i := \operatorname{diag}(\mathbf{l}_i)$. Define a vector $\boldsymbol{m}_i$ to encode the nonzero entries in the ith row of $\overline{\mathbf{K}}_{k+1}$, following*

$$\begin{cases}\boldsymbol{m}_i(j) = 0 \text{ if } [\overline{\mathbf{E}}]_{ij} = 0,\\ \boldsymbol{m}_i(j) = 1 \text{ if } [\overline{\mathbf{E}}]_{ij} \neq 0,\end{cases}$$

and let $\boldsymbol{\mathcal{M}}_i = \operatorname{diag}(\boldsymbol{m}_i)$. Then, the optimal one-step gain is given by

$$\overline{\mathbf{K}}_{k+1} = \sum_{i=1}^{n} \boldsymbol{\mathcal{L}}_i\overline{\mathbf{P}}_{k+1|k}\overline{\mathbf{C}}^T\boldsymbol{\mathcal{M}}_i(\mathbf{I} - \boldsymbol{\mathcal{M}}_i + \boldsymbol{\mathcal{M}}_i\overline{\mathbf{S}}_{k+1}\boldsymbol{\mathcal{M}}_i)^{-1}, \tag{19}$$

where $\overline{\mathbf{S}}_{k+1}$ is the innovation covariance at step $k+1$, given by

$$\overline{\mathbf{S}}_{k+1} = \overline{\mathbf{C}}\overline{\mathbf{P}}_{k+1|k}\overline{\mathbf{C}}^T + \overline{\mathbf{R}}. \tag{20}$$

To obtain the steady-state gains that stabilize the observer, propagate the error covariance equations (14) and (16) using the computed gains $\overline{\mathbf{K}}_{k+1}$ until the error covariance reaches a constant steady-state value.

To provide an example, this method was applied to a randomly generated system with $n = 5$, $p = 4$, and $T = 3$. The relevant system matrices are given by

$$\mathbf{A}_0 = \begin{bmatrix} 0.4279 & -0.2368 & 0.2380 & -0.0711 & -0.0421 \\ -0.3167 & 0.4445 & 0.0860 & 0.0732 & 0.2731 \\ 0.1251 & -0.0783 & 0.2045 & 0.0933 & 0.2217 \\ -0.0244 & 0.1118 & 0.0514 & 0.7449 & -0.1913 \\ 0.0527 & 0.3360 & 0.1149 & -0.1822 & 0.5821 \end{bmatrix}, \mathbf{A}_1 = \begin{bmatrix} 0.1243 & 0.0740 & 0.0666 & 0.0240 & 0.2917 \\ 0.0478 & 0.2650 & 0.1951 & 0.1417 & 0.3739 \\ 0.1123 & 0.2366 & 0.1091 & -0.0389 & 0.2680 \\ 0.0580 & 0.1633 & -0.0228 & 0.0768 & 0.0865 \\ 0.2781 & 0.3349 & 0.3145 & 0.1259 & 0.4346 \end{bmatrix},$$

$$\mathbf{A}_2 = \begin{bmatrix} 0.2142 & -0.0281 & 0.2117 & 0.2907 & 0.3462 \\ -0.0281 & -0.0392 & 0.1449 & 0.3959 & 0.1578 \\ 0.2117 & 0.1449 & 0.5101 & 0.2480 & 0.0562 \\ 0.2907 & 0.3959 & 0.2480 & 0.3354 & -0.1079 \\ 0.3462 & 0.1578 & 0.0562 & -0.1079 & 0.0269 \end{bmatrix}, \mathbf{C}_0 = \begin{bmatrix} 0 & 1.1174 & 1.1006 & -0.7423 & 0.7481 \\ -0.7697 & -1.0891 & 1.5442 & -1.0616 & -0.1924 \\ 0.3714 & 0.0326 & 0.0859 & 2.3505 & 0.8886 \\ -0.2256 & 0 & -1.4916 & -0.6156 & 0 \end{bmatrix},$$

$$\mathbf{C}_1 = \begin{bmatrix} -2.3299 & 0.4517 & 0.8620 & 0 & 1.2607 \\ -1.4491 & -0.1303 & 0 & 0 & 0.6601 \\ 0 & 0.1837 & 0.4550 & 1.0391 & -0.0679 \\ 0.3914 & -0.4762 & -0.8487 & -1.1176 & -0.1952 \end{bmatrix}, \mathbf{C}_2 = \begin{bmatrix} 1.1921 & 1.0205 & -2.4863 & 0.0799 & 0.8577 \\ -1.6118 & 0.8617 & 0.5812 & -0.9485 & 0 \\ -0.0245 & 0.0012 & 0 & 0.4115 & 0.4494 \\ 0 & -0.0708 & 0 & 0.6770 & 0 \end{bmatrix},$$

$$\mathbf{Q} = \begin{bmatrix} 3.4936 & -0.0628 & -0.1711 & -0.0747 & 0.1485 \\ -0.0628 & 4.7013 & -0.0263 & -0.1224 & 0.1059 \\ -0.1711 & -0.0263 & 2.4672 & -0.0859 & -0.1648 \\ -0.0747 & -0.1224 & -0.0859 & 5.4152 & 0.0290 \\ 0.1485 & 0.1059 & -0.1648 & 0.0290 & 3.1286 \end{bmatrix}, \mathbf{R} = \begin{bmatrix} 4.8408 & 0.0130 & 0.1004 & 0.0556 \\ 0.0130 & 13.1181 & -0.0201 & -0.0560 \\ 0.1004 & -0.0201 & 5.6185 & -0.0739 \\ 0.0556 & -0.0560 & -0.0739 & 5.4767 \end{bmatrix},$$

$$\mathbf{E} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 \end{bmatrix}.$$

For this system, the resulting steady-state gains are given by

$$\overline{\mathbf{K}} = \text{diag}(\mathbf{K}_1, \mathbf{K}_2, \mathbf{K}_0),$$

where

$$\mathbf{K}_1 = \begin{bmatrix} -0.2613 & 0 & 0.0103 & 0 \\ 0 & 0.0603 & 0 & 0 \\ 0 & 0 & 0.1030 & 0 \\ 0.0406 & -0.0180 & 0 & 0 \\ 0.1485 & 0.0006 & 0 & 0.0438 \end{bmatrix}, \mathbf{K}_2 = \begin{bmatrix} 0.1473 & 0 & 0.0112 & 0 \\ 0 & 0.1371 & 0 & 0 \\ 0 & 0 & 0.0795 & 0 \\ -0.0041 & -0.1601 & 0 & 0 \\ 0.1536 & 0.0356 & 0 & 0.0547 \end{bmatrix},$$

$$\mathbf{K}_0 = \begin{bmatrix} 0.1173 & 0 & 0.1029 & 0 \\ 0 & -0.1834 & 0 & 0 \\ 0 & 0 & 0.0976 & 0 \\ -0.0938 & -0.2118 & 0 & 0 \\ 0.1250 & -0.0384 & 0 & -0.0225 \end{bmatrix}.$$

The order of the Kalman gain blocks follows the same order as the output matrix $\overline{\mathbf{C}}$, defined in (11). Figure 1 shows the trace of the covariance, obtained by propagating the covariance equations (3) and (5), that results when these periodic gains are applied to the periodic filter (2) and (4).

Remark 1. It is important to stress that the one-step algorithm proposed herein compares well, from a computational point of view, with the centralized counterpart, as shown in Reference 32. This is so because it inherits the properties of the one-step algorithm for LTI system proposed in Reference 9, only here the system has Tn states instead of n states. However, that is exactly the same for the centralized counterpart.

4.3 | Finite-horizon method

This method, rather than optimizing the gains at each step, solves the finite-horizon problem (17), repeated here for clarity

$$\begin{aligned} &\underset{\overline{\mathbf{K}}_i, i=1,\dots,W}{\text{minimize}} \quad \sum_{k=1}^{W} \text{tr}(\overline{\mathbf{P}}_{k|k}), \\ &\text{subject to} \quad \overline{\mathbf{K}}_i \in \text{Sparse}(\overline{\mathbf{E}}), \quad i = 1, 2, \dots, W. \end{aligned} \tag{21}$$

The idea behind this formulation is to approximate the steady-state behavior by making the window W large enough, thus obtaining a gain that outperforms the one computed through the one-step method. However, this problem formulation is nonconvex, since the objective function is polynomial in the optimization variables $\overline{\mathbf{K}}_k$. This can be verified by propagating the covariances $\overline{\mathbf{P}}_{k|k}$ in terms of $\overline{\mathbf{P}}_{0|0}$ and $\overline{\mathbf{K}}_k$ using (14) and (16). On the other hand, by fixing all of the gains as constants, except for one $\overline{\mathbf{K}}_k$, a partial problem can be written as

$$\begin{aligned} &\underset{\overline{\mathbf{K}}_k}{\text{minimize}} \quad \sum_{i=1}^{W} \text{tr}(\overline{\mathbf{P}}_{i|i}), \\ &\text{subject to} \quad \overline{\mathbf{K}}_k \in \text{Sparse}(\overline{\mathbf{E}}). \end{aligned} \tag{22}$$

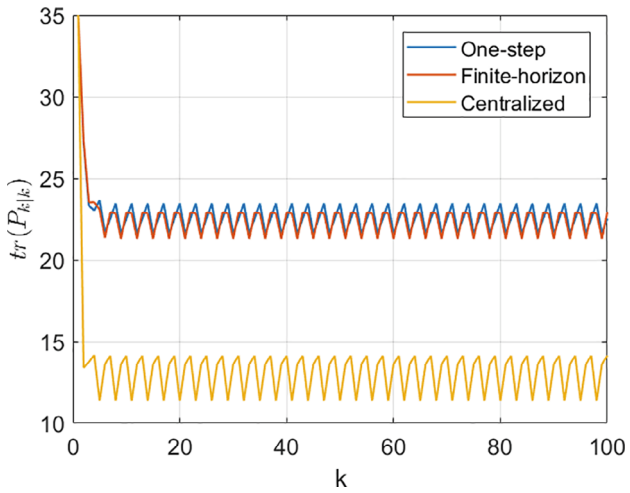

FIGURE 1 Evolution of the trace of the covariance of the finite-horizon, one-step, and centralized solutions

This way, the partial problem (22) becomes quadratic, since every gain except $\overline{\mathbf{K}}_k$ is constant. A closed-form solution to this partial problem for LTI systems has already been demonstrated in Reference 9 and an iterative algorithm that approximates the optimal solution to the full problem is presented in the same paper. For the sake of completeness, the solution is adapted in the following theorem for the time-lifted system. Notice that the difference lies on the system matrices, which correspond to the time-lifted formulation that was previously introduced.

Theorem 2. *Define a matrix* $\mathbf{Z}$ *such that the vector* $\mathbf{Z}\text{vec}(\overline{\mathbf{K}}_k)$ *contains the nonzero elements of* $\overline{\mathbf{K}}_k$ *according to the desired sparsity pattern. The closed-form solution of (22) is given by*

$$\text{vec}(\overline{\mathbf{K}}_k) = \mathbf{Z}^T(\mathbf{Z}(\overline{\mathbf{S}}_k \otimes \mathbf{\Lambda}_{k+1})\mathbf{Z}^T)^{-1}\mathbf{Z}vec(\mathbf{\Lambda}_{k+1}\overline{\mathbf{P}}_{k|k-1}\overline{\mathbf{C}}^T), \tag{23}$$

where $\overline{\mathbf{S}}_k$ *is computed as in (20), and*

$$\mathbf{\Lambda}_{k+1} = \mathbf{I}_n + \sum_{i=k+1}^{W} \mathbf{\Gamma}_{k+1,i}^T \mathbf{\Gamma}_{k+1,i},$$

with

$$\mathbf{\Gamma}_{k_i,k_f} = \prod_{j=k_i}^{k_f} (\mathbf{I}_n - \overline{\mathbf{K}}_{k_i+k_f-j}\overline{\mathbf{C}})\overline{\mathbf{A}}. \tag{24}$$

When $k_i > k_f$, *the convention* $\mathbf{\Gamma}_{k_i,k_f} = \mathbf{I}$ *is followed.*

To provide an example, this method was applied to the same system used in Section 4.2. Figure 1 depicts the evolution of the trace of the covariance with the finite-horizon gains. The steady-state gains obtained using this method are given by

$$\overline{\mathbf{K}} = \text{diag}(\mathbf{K}_1, \mathbf{K}_2, \mathbf{K}_0),$$

where

$$\mathbf{K}_1 = \begin{bmatrix} -0.2691 & 0 & 0.0053 & 0 \\ 0 & 0.0494 & 0 & 0 \\ 0 & 0 & 0.0966 & 0 \\ 0.0364 & -0.0139 & 0 & 0 \\ 0.1361 & 0.0044 & 0 & 0.0512 \end{bmatrix}, \quad \mathbf{K}_2 = \begin{bmatrix} 0.1461 & 0 & 0.0064 & 0 \\ 0 & 0.1364 & 0 & 0 \\ 0 & 0 & 0.0700 & 0 \\ -0.0060 & -0.1608 & 0 & 0 \\ 0.1483 & 0.0331 & 0 & 0.0498 \end{bmatrix},$$

$$\mathbf{K}_0 = \begin{bmatrix} 0.0964 & 0 & 0.0938 & 0 \\ 0 & -0.1655 & 0 & 0 \\ 0 & 0 & 0.1239 & 0 \\ -0.0657 & -0.2030 & 0 & 0 \\ 0.2564 & -0.0177 & 0 & -0.0143 \end{bmatrix}.$$

The finite-horizon algorithm clearly outperforms the one-step method, at the cost of higher computational load.

4.4 | Performance comparison

In the simulations, the randomly generated system in Section 4.2 was considered. The input matrices, with $m = 2$, follow

$$\mathbf{B}_0 = \begin{bmatrix} 0 & -0.8637 \\ -0.1649 & 0.0774 \\ 0 & -1.2141 \\ 0 & -1.1135 \\ 1.1093 & 0 \end{bmatrix}, \quad \mathbf{B}_1 = \begin{bmatrix} 0 & -1.2571 \\ -1.1201 & -0.8655 \\ 2.5260 & -0.1765 \\ 0 & 0.7914 \\ 0 & -1.3320 \end{bmatrix}, \quad \mathbf{B}_2 = \begin{bmatrix} 0 & 1.2503 \\ -0.0549 & 0.9298 \\ 0 & 0.2398 \\ 0 & -0.6904 \\ 0.3502 & -0.6516 \end{bmatrix}.$$

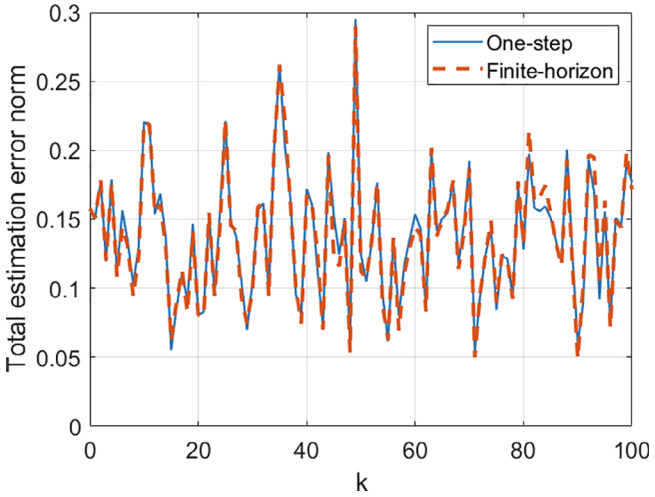

FIGURE 2 Total estimation error norm for the one-step and finite-horizon methods

TABLE 1 Sum of the projected steady-state performances for the one-step and finite-horizon methods over one period $(\mathrm{tr}(\mathbf{P}_\infty) + \cdots + \mathrm{tr}(\mathbf{P}_{\infty+T-1}))$

One-step	Finite-horizon
67.6257	67.0842

TABLE 2 Sum of the measured steady-state performances for the one-step and finite-horizon methods over one period, averaged over 1000 simulations

One-step	Finite-horizon
67.7447	67.1648

The initial condition for the state was set to $\mathbf{x}_0 = [4\ 1\ 2\ 1\ 3]^T$ and the input is given by a constant signal $\mathbf{u}_k = [0.5\ 1]^T$. For this setting, to better evaluate the performance of the proposed solution, the Monte Carlo method was applied with 1000 simulations, each carried out with different, randomly generated noise signals.

The initial estimate of the filter $\hat{\mathbf{x}}_0$ is randomly generated in each Monte Carlo run, considering a Gaussian distribution with mean $\mathbf{x}_0$ and standard deviation equal to the square root of the diagonal of $\mathbf{P}_{0|0}$, which was set at $\mathbf{P}_{0|0} = 7\mathbf{I}$. The process and measurement noises are zero-mean Gaussian processes with standard deviation given by the square root of the diagonal of the covariance matrices, $\mathbf{Q}$ and $\mathbf{R}$, respectively. The one-step and finite-horizon methods lead to very similar filtering performance in this case. This can be observed in Figure 2, which depicts the total norm of the estimation error of all state variables for the one-step (solid line) and the finite-horizon (dashed-line) methods. To supplement the graphical data, Table 1 compares the theoretical steady-state performance of the system between both methods. The performance value is obtained by summing the trace of the steady-state covariance over one period $\mathrm{tr}(\mathbf{P}_\infty) + \cdots + \mathrm{tr}(\mathbf{P}_{\infty+T-1})$. The measured performance over a period, averaged over 1000 simulations, is detailed in Table 2. The finite-horizon method outperforms the one-step method. However, given the heavier computational load for computing the finite-horizon gains, the one-step method would be preferable in most situations, unless very fine estimates are required or the difference in performance is significant.

5 | CONTROLLER GAINS

To compute the controller gains, the lifted dynamics are considered. First, the cost function (9) is reformulated as a quadratic cost of the lifted system. Subsequently, an equivalent problem is derived that consists in the minimization of the steady-state solution of a difference matrix equation. Finally, the one-step algorithm is presented as a solution

to this problem. This algorithm optimizes the solution of said matrix equation at each time-step, which converges to well-performing steady-state feedback gains.

5.1 | Cost function reformulation

Consider the lifted system

$$\begin{cases} \overline{\mathbf{x}}_{k+1} = \overline{\mathbf{A}}\overline{\mathbf{x}}_k + \overline{\mathbf{B}}\overline{\mathbf{u}}_k, \\ \overline{\mathbf{u}}_k = -\overline{\mathbf{K}}\overline{\mathbf{x}}_k, \end{cases} \tag{25}$$

where $\overline{\mathbf{x}}$, $\overline{\mathbf{u}}$, $\overline{\mathbf{A}}$, and $\overline{\mathbf{B}}$ are defined as in (11). The infinite-horizon quadratic cost (9) can be reformulated, for the lifted dynamics, as

$$\overline{J} = \sum_{k=0}^{+\infty} (\overline{\mathbf{x}}_k^T \overline{\mathbf{Q}} \overline{\mathbf{x}}_k + \overline{\mathbf{u}}_k^T \overline{\mathbf{R}} \overline{\mathbf{u}}_k), \tag{26}$$

where $\overline{\mathbf{Q}} = \text{diag}(\mathbf{Q}, \dots, \mathbf{Q}) \in \mathbb{R}^{nT \times nT}$ and $\overline{\mathbf{R}} = \text{diag}(\mathbf{R}, \dots, \mathbf{R}) \in \mathbb{R}^{mT \times mT}$. Now that the cost $\overline{J}$ is written for an invariant system, it is possible to follow the procedure described in Reference 30. The decentralized control problem can then be posed as

$$\begin{aligned} &\underset{\overline{\mathbf{K}}}{\text{minimize}} \lim_{k \to \infty} \text{tr}(\overline{\mathbf{P}}_k), \\ &\text{subject to } \overline{\mathbf{P}}_{k+1} = (\overline{\mathbf{A}} - \overline{\mathbf{B}}\,\overline{\mathbf{K}})^T \overline{\mathbf{P}}_k (\overline{\mathbf{A}} - \overline{\mathbf{B}}\,\overline{\mathbf{K}}) + \overline{\mathbf{Q}} + \overline{\mathbf{K}}^T \overline{\mathbf{R}}\,\overline{\mathbf{K}}, \\ &\qquad \overline{\mathbf{P}}_0 \succ 0, \\ &\qquad \overline{\mathbf{K}} \in \text{Sparse}(\overline{\mathbf{E}}_c), \end{aligned} \tag{27}$$

where the sparsity pattern $\overline{\mathbf{E}}_c \in \mathbb{R}^{mT \times nT}$ follows the structure

$$\overline{\mathbf{E}}_c = \begin{bmatrix} \mathbf{0} & \dots & \mathbf{0} & \mathbf{E}_c \\ \mathbf{0} & \dots & \mathbf{0} & \mathbf{E}_c \\ \vdots & & \vdots & \vdots \\ \mathbf{0} & \dots & \mathbf{0} & \mathbf{E}_c \end{bmatrix}. \tag{28}$$

To see how the sparsity pattern is derived, see the Appendix, and for a detailed explanation on how (27) is derived, see Reference 30. This problem formulation can now be used to develop an algorithm that computes well-performing steady-state feedback gains with which the lifted input can be computed.

5.2 | One-step method

Consider the difference matrix equation with a time-varying gain.

$$\overline{\mathbf{P}}_{k+1} = (\overline{\mathbf{A}} - \overline{\mathbf{B}}\,\overline{\mathbf{K}}_{k+1})^T \overline{\mathbf{P}}_k (\overline{\mathbf{A}} - \overline{\mathbf{B}}\,\overline{\mathbf{K}}_{k+1}) + \overline{\mathbf{Q}} + \overline{\mathbf{K}}_{k+1}^T \overline{\mathbf{R}}\,\overline{\mathbf{K}}_{k+1}. \tag{29}$$

The idea behind the one-step method is to minimize the trace of $\overline{\mathbf{P}}_{k+1}$ at each step by solving the optimization problem

$$\begin{aligned} &\underset{\overline{\mathbf{K}}_{k+1}}{\text{minimize}} \quad \text{tr}(\overline{\mathbf{P}}_{k+1}), \\ &\text{subject to} \quad \overline{\mathbf{K}}_{k+1} \in \text{Sparse}(\overline{\mathbf{E}}_c). \end{aligned} \tag{30}$$

Notice that $\overline{\mathbf{P}}_{k+1}$ depends on $\overline{\mathbf{P}}_k$. As such, this method amounts to propagating the difference equation $\overline{\mathbf{P}}_{k+1}$ until steady-state is achieved, similarly to the one-step method for computing the observer gains introduced in Section 4.2, which instead propagates the covariance equations. In the unconstrained case, this method yields the optimal steady-state gain. However, for this case, where the gain is subject to a sparsity constraint, the gain obtained is suboptimal. Nonetheless, since (30) is a quadratic optimization problem subject to a linear constraint, it allows for the computation of well-performing steady-state gains in a very computationally efficient way. To speed up the algorithm, the following theorem, adapted from Reference 30, details the closed-form solution of (30).

Theorem 3. *Consider a sparsity pattern $\overline{\mathbf{E}}_c$. Let $\mathbf{l}_i$ denote a column vector, whose entries are all set to zero except for the ith one, which is set to 1, and define $\boldsymbol{\mathcal{L}}_i := \text{diag}(\mathbf{l}_i)$. Define a vector $\boldsymbol{m}_j$ to encode the nonzero entries in the jth column of $\overline{\mathbf{K}}_{k+1}$, following*

$$\begin{cases} \boldsymbol{m}_j(i) = 0 \text{ if } [\overline{\mathbf{E}}_c]_{ij} = 0, \\ \boldsymbol{m}_j(i) = 1 \text{ if } [\overline{\mathbf{E}}_c]_{ij} \neq 0, \end{cases}$$

and let $\boldsymbol{\mathcal{M}}_j = \text{diag}(\boldsymbol{m}_j)$. Then, the optimal one-step gain is given by

$$\overline{\mathbf{K}}_{k+1} = \sum_{j=1}^{n} (\mathbf{I} - \boldsymbol{\mathcal{M}}_j + \boldsymbol{\mathcal{M}}_j \overline{\mathbf{S}}_{k+1} \boldsymbol{\mathcal{M}}_j)^{-1} \boldsymbol{\mathcal{M}}_j \overline{\mathbf{B}}^T \overline{\mathbf{P}}_k \overline{\mathbf{A}} \boldsymbol{\mathcal{L}}_j, \tag{31}$$

where $\overline{\mathbf{S}}_{k+1}$ is given by

$$\overline{\mathbf{S}}_{k+1} = \overline{\mathbf{B}}^T \overline{\mathbf{P}}_k \overline{\mathbf{B}} + \overline{\mathbf{R}}. \tag{32}$$

To provide a simple example, this method was applied to a randomly generated system with $n = 4$, $m = 3$, and $T = 2$. The relevant system matrices are given by

$$\mathbf{A}_0 = \begin{bmatrix} -0.1334 & 0.0179 & 0.4882 & 0.3194 \\ -0.1266 & -0.3831 & 0.3269 & -0.1444 \\ 0.5410 & 0.2399 & 0.7147 & -0.0521 \\ -0.1789 & 0.2931 & 0.0880 & -0.2500 \end{bmatrix}, \ \mathbf{A}_1 = \begin{bmatrix} 0.6545 & 0.0417 & -0.1223 & -0.4364 \\ 0.0417 & 0.2136 & 0.0895 & -0.7524 \\ -0.1223 & 0.0895 & 0.7871 & -0.0958 \\ -0.4364 & -0.7524 & -0.0958 & -0.3829 \end{bmatrix},$$

$$\mathbf{B}_0 = \begin{bmatrix} 1.4022 & 0.0660 & 0.9287 \\ -1.3677 & 0 & -0.4908 \\ -0.2925 & -0.3222 & 0 \\ 1.2708 & 0 & 0.5907 \end{bmatrix}, \ \mathbf{B}_1 = \begin{bmatrix} 0 & -0.3257 & -2.1182 \\ 0 & 0 & 0.7071 \\ 0.7898 & 1.3096 & -1.0434 \\ -0.8012 & 0.1604 & 1.0682 \end{bmatrix},$$

$$\mathbf{E} = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}, \ \mathbf{Q} = \mathbf{R} = \mathbf{I}.$$

This system was then lifted by applying the methodology introduced in Section 3. It is then straightforward to apply the one-step method to the lifted dynamics. Figure 3 depicts the trace of $\overline{\mathbf{P}}_k$. As it can be seen, a stable steady-state was achieved and the control feedback gains obtained are given by

$$\mathbf{K}_0 = \begin{bmatrix} -0.0789 & 0.1482 & 0 & -0.0174 \\ -0.1858 & -0.1168 & -0.2385 & 0 \\ 0 & 0 & 0.0985 & 0.0715 \end{bmatrix},$$

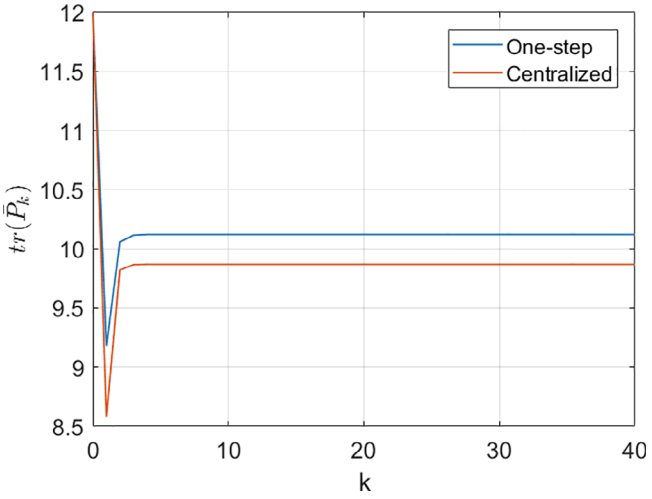

FIGURE 3 Trace of $\overline{\mathbf{P}}_k$ of the one-step and centralized solutions

$$\mathbf{K}_1 = \begin{bmatrix} -0.0051 & -0.0561 & 0 & -0.1209 \\ 0.2165 & 0.1508 & 0.1558 & 0 \\ 0 & 0 & -0.1787 & -0.0808 \end{bmatrix}.$$

6 | SIMULATION RESULTS FOR THE QTP

This section details the application of the proposed solution to the case of a QTP.[31] The observer is implemented as a continuous-discrete filter for the linearized dynamics of the QTP. The controllers are designed for the linearized dynamics augmented with additional integral states that accumulate the output tracking error, thus reduce the steady-state error to zero. Then, a controller is implemented at each pump. The performance of these solutions is assessed via simulation results.

6.1 | Quadruple-tank process

Consider a system of four interconnected tanks, such as the one depicted in Figure 4. The nonlinear, continuous-time dynamics of this system are given by

$$\begin{cases} \dot{x}_1 = -\frac{r_1}{R_1}\sqrt{2gx_1} + \frac{r_3}{R_1}\sqrt{2gx_3} + \frac{\gamma_1 k_1}{R_1}u_1, \\ \dot{x}_2 = -\frac{r_2}{R_2}\sqrt{2gx_2} + \frac{r_4}{R_2}\sqrt{2gx_4} + \frac{\gamma_2 k_2}{R_2}u_2, \\ \dot{x}_3 = -\frac{r_3}{R_3}\sqrt{2gx_3} + \frac{(1-\gamma_2)k_2}{R_3}u_2, \\ \dot{x}_4 = -\frac{r_4}{R_4}\sqrt{2gx_4} + \frac{(1-\gamma_1)k_1}{R_4}u_1, \end{cases} \tag{33}$$

where x_i is the water level of tank i, R_i is the cross-section, and r_i is the cross-section of the outlet hole for $i = 1, 2, 3, 4$. The voltage applied to pump i, $i = 1, 2$, is u_i, and the corresponding flow is $k_i u_i$. This flow is controlled by the valves $\gamma_i \in (0, 1)$, where a higher value of γ_i means more flow is directed toward the bottom tanks, while a lower value corresponds to more flow being directed to the upper tanks. The acceleration of gravity is denoted g. The parameter values used in the simulations are presented in Table 3.

It is assumed that the water level of each tank can be measured. This way, the state-space representation of (33) is given by

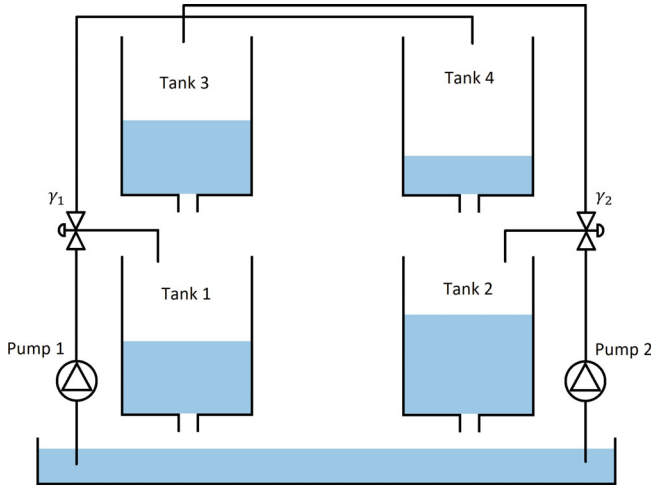

FIGURE 4 Quadruple-tank process

TABLE 3 Parameter values of the QTP used in the simulations

Parameter (unit)	Value
R_1, R_3 (cm^2)	28
R_2, R_4 (cm^2)	32
r_1, r_3 (cm^2)	0.071
r_2, r_4 (cm^2)	0.057
γ_1	0.7
γ_2	0.6
g (cm/s^2)	981

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}), \\ \mathbf{y} = \mathbf{x}, \end{cases} \tag{34}$$

where $\mathbf{x}$ is the state, $\mathbf{u}$ is the input, and $\mathbf{y}$ is the output. The function $\mathbf{f}(\mathbf{x}, \mathbf{u})$ represents the nonlinear dynamics, which can be directly inferred from (33).

6.2 | Linearization and discretization

Let $\mathbf{x}_{\text{nom}}$, $\mathbf{u}_{\text{nom}}$, and $\mathbf{y}_{\text{nom}}$ be a nominal periodic trajectory for the state, input, and output, respectively. Now, introduce the perturbation variables $\Delta\mathbf{x} = \mathbf{x} - \mathbf{x}_{\text{nom}}$, $\Delta\mathbf{u} = \mathbf{u} - \mathbf{u}_{\text{nom}}$, and $\Delta\mathbf{y} = \mathbf{y} - \mathbf{y}_{\text{nom}}$. The dynamics of the QTP linearized about its nominal trajectory are given by

$$\mathbf{A}(t) = \left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right|_{\mathbf{x}_{\text{nom}}} = \begin{bmatrix} -\frac{1}{T_1} & 0 & \frac{R_3}{R_1 T_3} & 0 \\ 0 & -\frac{1}{T_2} & 0 & \frac{R_4}{R_2 T_4} \\ 0 & 0 & -\frac{1}{T_3} & 0 \\ 0 & 0 & 0 & -\frac{1}{T_4} \end{bmatrix}, \quad \mathbf{B} = \left.\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right|_{\mathbf{u}_{\text{nom}}} = \begin{bmatrix} \frac{\gamma_1 k_1}{A_1} & 0 \\ 0 & \frac{\gamma_2 k_2}{A_2} \\ 0 & \frac{(1-\gamma_2) k_2}{A_3} \\ \frac{(1-\gamma_1) k_1}{A_4} & 0 \end{bmatrix}, \tag{35}$$

where the time constants T_i are

$$T_i = \frac{R_i}{r_i}\sqrt{\frac{2x_{i_{\text{nom}}}}{g}}, \quad i = 1, \dots, 4. \tag{36}$$

The state-space representation of the linearized system can then be written as

$$\begin{cases} \Delta\dot{\mathbf{x}} = \mathbf{A}(t)\Delta\mathbf{x} + \mathbf{B}\Delta\mathbf{u}, \\ \Delta\mathbf{y} = \mathbf{C}\Delta\mathbf{x}, \end{cases} \tag{37}$$

where $\mathbf{C} = \mathbf{I}$. To simplify the notation, from this point forward, a signal at step k will be denoted by $\mathbf{v}(k)$. To obtain a discrete-time model of the system, consider a constant sampling interval T_s. The discrete-time dynamics, assuming zero-order hold of the input and that $\mathbf{A}(t_k)$ is invertible,[33(pp26-27)] are described by

$$\begin{cases} \Delta\mathbf{x}(k+1) = \mathbf{A}_d(k)\Delta\mathbf{x}(k) + \mathbf{B}_d(k)\Delta\mathbf{u}(k), \\ \Delta\mathbf{y}(k) = \mathbf{C}_d\Delta\mathbf{x}(k), \end{cases} \tag{38}$$

where

$$\begin{cases} \mathbf{A}_d(k) = e^{\mathbf{A}(t_k)T_s}, \\ \mathbf{B}_d(k) = \mathbf{A}_d(k)(\mathbf{I} - e^{-\mathbf{A}(t_k)T_s})\mathbf{A}^{-1}(t_k)\mathbf{B}, \\ \mathbf{C}_d = \mathbf{C}. \end{cases} \tag{39}$$

6.3 | Filter implementation

Following the previous discussion, it is possible to design local state observers by decoupling the discrete LTP dynamics (38). Let $\mathbf{A}^{(i,j)}, i, j = 1, \dots, 4$, denote the element on the ith row and jth column of $\mathbf{A}$. The predicted linearized estimates of each tank are given by

$$\begin{cases} \Delta\hat{x}_1(k+1|k) = \mathbf{A}_d(k)^{(1,1)}\Delta\hat{x}_1(k|k) + \mathbf{A}_d(k)^{(1,3)}\Delta\hat{x}_3(k|k) + \mathbf{B}_d(k)^{(1,1)}\Delta u_1(k) + \mathbf{B}_d(k)^{(1,2)}\Delta u_2(k), \\ \Delta\hat{x}_2(k+1|k) = \mathbf{A}_d(k)^{(2,2)}\Delta\hat{x}_2(k|k) + \mathbf{A}_d(k)^{(2,4)}\Delta\hat{x}_4(k|k) + \mathbf{B}_d(k)^{(2,1)}\Delta u_1(k) + \mathbf{B}_d(k)^{(2,2)}\Delta u_2(k), \\ \Delta\hat{x}_3(k+1|k) = \mathbf{A}_d(k)^{(3,3)}\Delta\hat{x}_3(k|k) + \mathbf{B}_d(k)^{(3,2)}\Delta u_2(k), \\ \Delta\hat{x}_4(k+1|k) = \mathbf{A}_d(k)^{(4,4)}\Delta\hat{x}_4(k|k) + \mathbf{B}_d(k)^{(4,1)}\Delta u_1(k). \end{cases} \tag{40}$$

Notice that the state equations of tanks 1 and 2 depend on the estimates of tanks 3 and 4, respectively. It is assumed that these estimates are obtained by means of communication.

When the output sample $k+1$ is available, the filtered linearized estimate is determined by

$$\Delta\hat{x}_i(k+1|k+1) = \Delta\hat{x}_i(k+1|k) + \mathbf{K}(k+1)^{(i,i)}(\Delta y_i(t_{k+1}) - \mathbf{C}_d^{(i,i)}\Delta\hat{x}_i(k+1|k)). \tag{41}$$

Finally, the actual state estimates can be recovered with

$$\hat{x}_i(k+1|k+1) = \Delta\hat{x}_i(k+1|k+1) + x_{i_{\text{nom}}}. \tag{42}$$

To compute the observer gains, the methodology introduced in Section 4 can be applied to the discrete-time dynamics (40). The following section provides a detailed example of this and simulation results are presented to assess the performance of the filter.

6.4 | Filter simulation

To obtain a nominal periodic trajectory for the system, the nonlinear system (34) was simulated with a periodic input signal $\mathbf{u} = \mathbf{u}_{\text{nom}}$. Figure 5 depicts the input signal used and the resulting response of the system can be seen in Figure 6. As it can be observed, for a given constant section of $\mathbf{u}$, the system reaches an equilibrium after some time. The various equilibrium points form a periodic trajectory for the state, and their values are depicted in Table 4. The discretized system matrices, linearized about the equilibria and considering a sampling time of $T_s = 1$ s, follow

$$\mathbf{A}_d(0) = \begin{bmatrix} 0.8890 & 0 & 0.3846 & 0 \\ 0 & 0.9054 & 0 & 0.1718 \\ 0 & 0 & 0.5903 & 0 \\ 0 & 0 & 0 & 0.8192 \end{bmatrix}, \quad \mathbf{A}_d(1) = \begin{bmatrix} 0.9664 & 0 & 0.0984 & 0 \\ 0 & 0.9754 & 0 & 0.0635 \\ 0 & 0 & 0.8999 & 0 \\ 0 & 0 & 0 & 0.9357 \end{bmatrix},$$

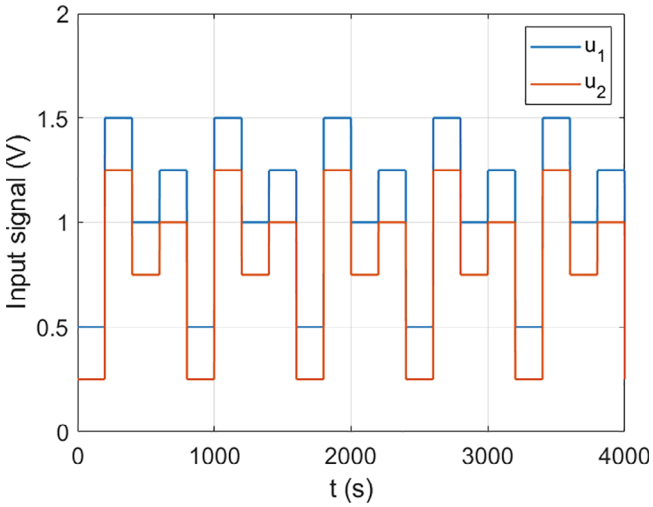

FIGURE 5 Periodic input signal used in the simulations

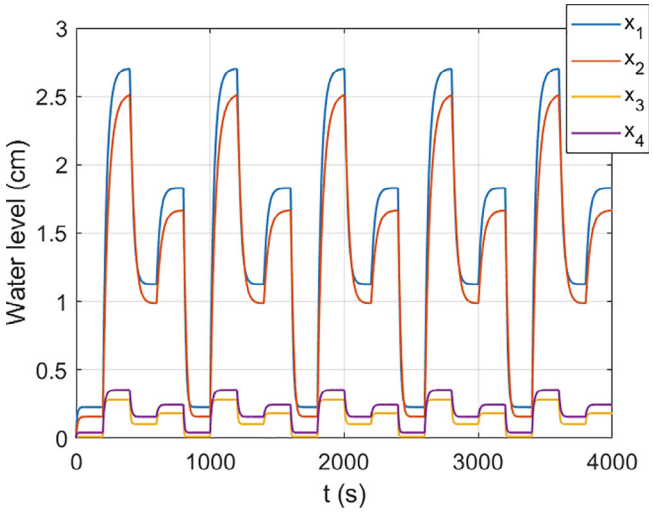

FIGURE 6 Response of the nonlinear system

TABLE 4 Equilibrium points for the input signal represented in Figure 5

Water level (cm)	$\mathbf{Eq_0}$ ($t = 195$ s)	$\mathbf{Eq_1}$ ($t = 395$ s)	$\mathbf{Eq_2}$ ($t = 595$ s)	$\mathbf{Eq_3}$ ($t = 795$ s)
x_1	0.2276	2.7012	1.1253	1.8292
x_2	0.1575	2.5067	0.9868	1.6640
x_3	0.0113	0.2837	0.1021	0.1815
x_4	0.0391	0.3523	0.1566	0.2446

$$\mathbf{A}_d(2) = \begin{bmatrix} 0.9484 & 0 & 0.1568 & 0 \\ 0 & 0.9611 & 0 & 0.0930 \\ 0 & 0 & 0.8388 & 0 \\ 0 & 0 & 0 & 0.9051 \end{bmatrix}, \quad \mathbf{A}_d(3) = \begin{bmatrix} 0.9593 & 0 & 0.1209 & 0 \\ 0 & 0.9699 & 0 & 0.0755 \\ 0 & 0 & 0.8765 & 0 \\ 0 & 0 & 0 & 0.9233 \end{bmatrix},$$

$$\mathbf{B}_d(0) = \begin{bmatrix} 0.0785 & 0.0102 \\ 0.0028 & 0.0598 \\ 0 & 0.0372 \\ 0.0283 & 0 \end{bmatrix}, \mathbf{B}_d(1) = \begin{bmatrix} 0.0818 & 0.0024 \\ 0.0010 & 0.0620 \\ 0 & 0.0454 \\ 0.0302 & 0 \end{bmatrix}, \quad \mathbf{B}_d(2) = \begin{bmatrix} 0.0811 & 0.0039 \\ 0.0015 & 0.0616 \\ 0 & 0.0439 \\ 0.0297 & 0 \end{bmatrix}, \mathbf{B}_d(3) = \begin{bmatrix} 0.0815 & 0.0030 \\ 0.0012 & 0.0619 \\ 0 & 0.0448 \\ 0.0300 & 0 \end{bmatrix},$$

leading to a LTP system with period $T = 4$. In reality, since each constant input section is simulated for 200 samples, the period of the system is actually $T = 800$, where the system dynamics follow

$$\begin{aligned} &(\mathbf{A}_d(0), \mathbf{B}_d(0)), \ k = 0, \ \dots, 199, \\ &(\mathbf{A}_d(1), \mathbf{B}_d(1)), \ k = 200, \ \dots, 399, \\ &(\mathbf{A}_d(2), \mathbf{B}_d(2)), \ k = 400, \ \dots, 599, \\ &(\mathbf{A}_d(3), \mathbf{B}_d(3)), \ k = 600, \ \dots, 799, \\ &(\mathbf{A}_d(0), \mathbf{B}_d(0)), \ k = 800, \ \dots, 999, \end{aligned}$$

and so forth. Nonetheless, to compute the observer gains, it will be shown via simulation results that it is reasonable to assume that the system has period $T = 4$.

Regarding sensor noise, the measurements $\Delta\mathbf{y}(k+1)$ were corrupted by additive white Gaussian noise $\mathbf{v}(k+1)$ with zero-mean and covariance given by

$$\mathbf{R} = \begin{bmatrix} 0.1 & 0.05 & 0.05 & 0.05 \\ 0.05 & 0.1 & 0.05 & 0.05 \\ 0.05 & 0.05 & 0.1 & 0.05 \\ 0.05 & 0.05 & 0.05 & 0.1 \end{bmatrix}.$$

The process noise covariance was set to $\mathbf{Q} = 0.01\mathbf{I}$. Given that it was considered that there is no process noise in the system model, the covariance $\mathbf{Q}$ was used purely as a tuning parameter for the filter. Notice that it can also be used to take into account linearization errors. In regards to the measurement topology, it was assumed that each tank only has access to its own measurements. This way, the sparsity pattern is given by $\mathbf{E} = \mathbf{I}$. A simulation of the filter, implemented with the one-step gains computed with $T = 4$, can be seen in Figure 7. The total estimation error norm is depicted in Figure 8, and a detailed view of the estimation error for each of the states can be seen in Figure 9. The simulations of the filters with the one-step gains with $T = 800$ and with the finite-horizon gains are not presented, as the results are very similar. A more detailed analysis, that showcases the difference in filtering performance, is presented in the following section.

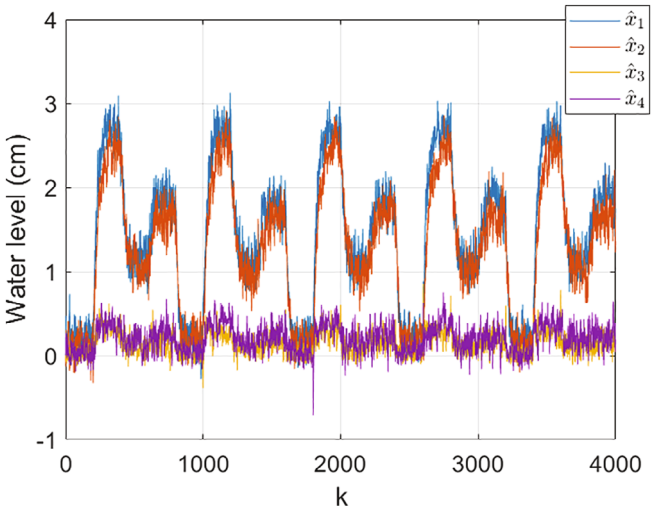

FIGURE 7 Filter simulation with the one-step gains computed with $T = 4$

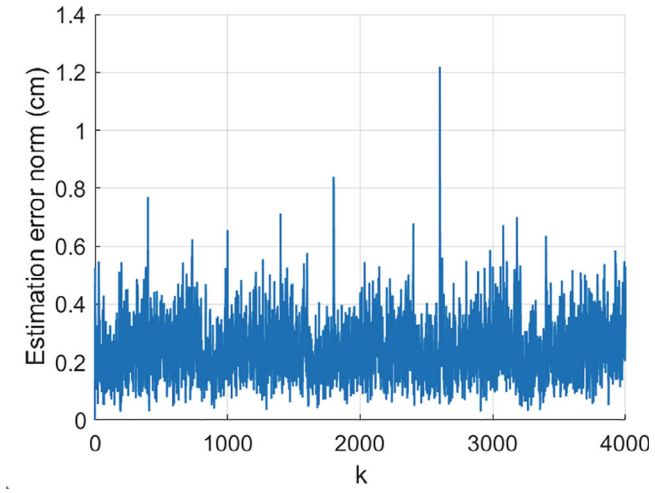

FIGURE 8 Estimation error norm for the filter implemented with the one-step gains computed with $T = 4$

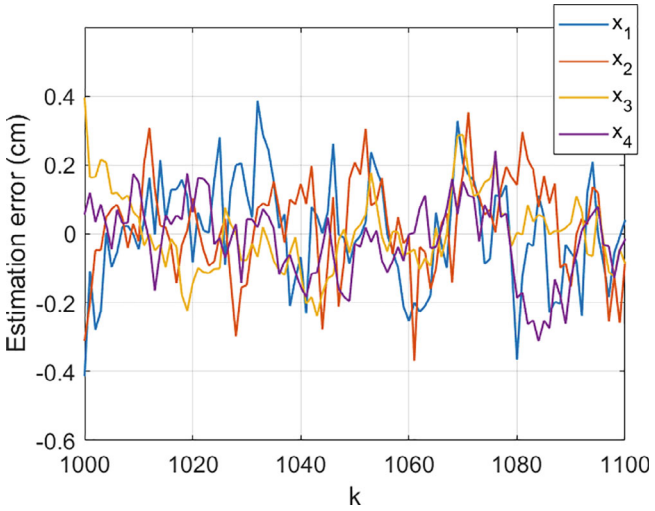

FIGURE 9 Detailed view of the estimation error for the one-step filter ($T = 4$)

6.5 | Monte Carlo analysis

To compare the filtering performance, similarly to Section 4.4, 1000 simulations were carried out, each with different, randomly generated noise signals. Figures 10 and 11 depict the total estimation error norm, averaged over 1000 simulations, of the filters (40)–(42) implemented using the one-step and the finite-horizon methods for computing the observer gains, respectively. The one-step method was used to compute the gains assuming that the system has period $T = 4$ and $T = 800$, while the finite-horizon method was only applied for $T = 4$ due to the higher computational requirements associated with this algorithm. The measured performance (sum of the variances over one period) is also shown in Table 5. The measured results suggest that the finite-horizon method leads to the best filtering performance, followed by the one-step method for $T = 4$, and lastly the one-step method for $T = 800$. While it may seem counterintuitive that considering a higher period leads to worse filtering performance, recall the discussion presented in Section 4.1. Due to the suboptimality of the lifted filter, using such a high period to compute the gains might end up introducing a significant amount of error when applying these gains to the LTP filter. Notice that, in all cases, the estimation error norm peaks when the equilibrium, and consequently the system dynamics, changes. This result is expected, as the transient sections before the system reaches its steady-state behavior are not accurately described by the linearized model. To further cement this argument, Figure 12 presents a detailed view of estimation error of each state for the observer implemented with the finite-horizon gains. During the transient phase, the estimation error for tanks 1 and 2 is higher compared to tanks 3 and 4, as the variations around the equilibrium are larger.

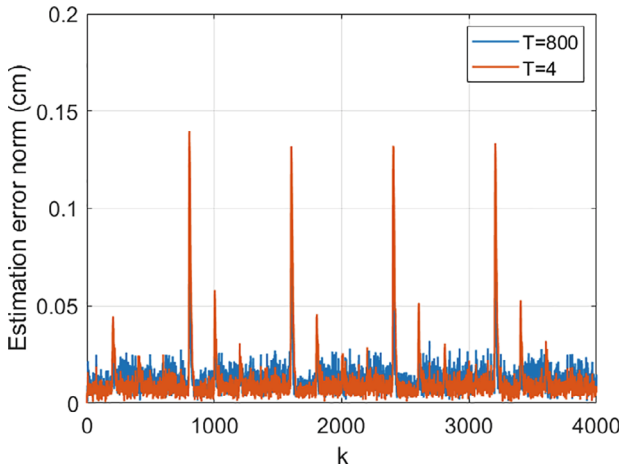

FIGURE 10 Total estimation error norm of the filter with the one-step gains ($T = 800$ and $T = 4$) averaged over 1000 simulations

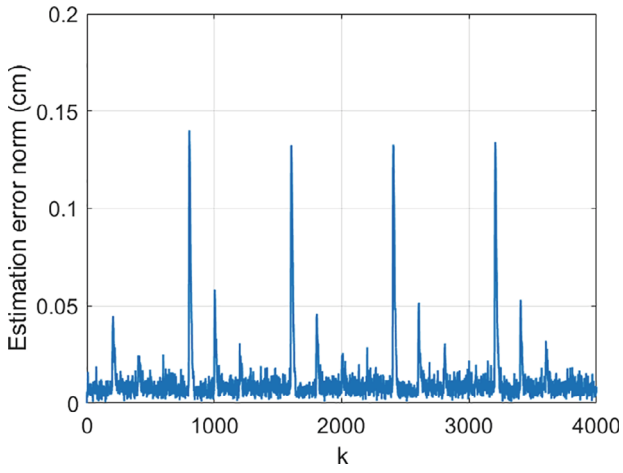

FIGURE 11 Total estimation error norm of the filter with the finite-horizon gains averaged over 1000 simulations

TABLE 5 Measured performance (sum of variances over one period) of the filters averaged over 1000 simulations

One-step ($T = 4$)	One-step ($T = 800$)	Finite-horizon
56.4021	110.5661	55.5212

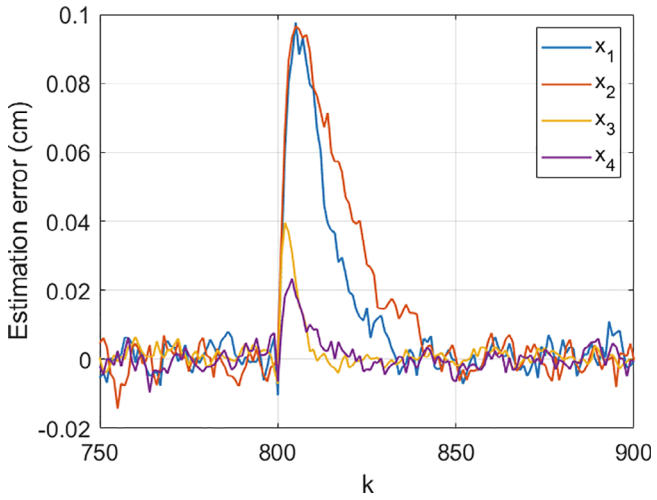

FIGURE 12 Detailed view of the estimation error of the filter implemented with the finite-horizon gains averaged over 1000 simulations

6.6 | Control

The objective of the controllers designed in this section is to control the water level in tanks 1 and 2. Thus, only x_1 and x_2 are assumed to be measured and, consequently, the output matrix is now given by

$$\mathbf{C} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}.$$

The system was then linearized about the equilibrium points depicted in Table 6, yielding a system with period $T = 3$. Following this, the system was discretized with a sampling time $T_s = 1$ s. To follow a constant reference signal $\mathbf{r}(k)$, two additional state variables $\mathbf{q}(k) = [q_1(k)\ q_2(k)]$ that accumulate the output tracking error were included, as given by

$$\mathbf{q}(k+1) = \mathbf{q}(k) + (\Delta\mathbf{y}(k) - \mathbf{r}(k)).$$

This yields the LTP system

$$\begin{cases} \begin{bmatrix} \Delta\mathbf{x}(k+1) \\ \mathbf{q}(k+1) \end{bmatrix} = \begin{bmatrix} \mathbf{A}_d(k) & \mathbf{0} \\ \mathbf{C}_d & \mathbf{I} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{x}(k) \\ \mathbf{q}(k) \end{bmatrix} + \begin{bmatrix} \mathbf{B}_d(k) & \mathbf{0} \\ \mathbf{0} & -\mathbf{I} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{u}(k) \\ \mathbf{r}(k) \end{bmatrix}, \\ \Delta\mathbf{y}(k) = \mathbf{C}_d\Delta\mathbf{x}(k). \end{cases} \tag{43}$$

It is then possible to compute the lifted controller gains by using the one-step method on the lifted system, obtained by applying the time-lifting technique introduced in Section 3 to (43). In the simulations, a local controller is implemented at each pump. For Pump 1, the controller uses the measured output $x_1(t)$ and its integral state $q_1(k)$, while for Pump 2 the controller has access to $x_2(t)$ and $q_2(k)$. Consequently, the sparsity pattern $\mathbf{E}_c$ is given by

TABLE 6 Equilibrium points

	$\mathbf{Eq_0}$	$\mathbf{Eq_1}$	$\mathbf{Eq_2}$
x_1 (cm)	1.3626	12.2594	5.4503
x_2 (cm)	1.4204	12.7404	5.6845
x_3 (cm)	0.1815	1.6339	0.7262
x_4 (cm)	0.1566	1.4090	0.6262
u_1 (V)	1	3	2
u_2 (V)	1	3	2

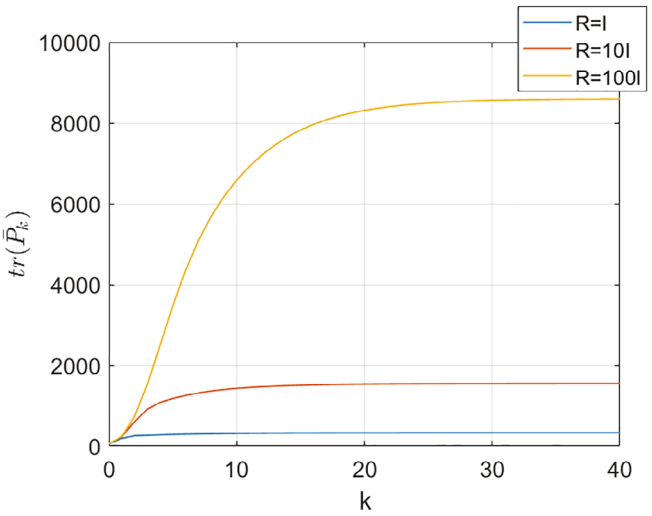

FIGURE 13 Trace of $\overline{\mathbf{P}}_k$

$$\mathbf{E}_c = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \tag{44}$$

It is important to note that the controllers compute the lifted input signal

$$\overline{\mathbf{u}}(k) = -\overline{\mathbf{K}}\overline{\mathbf{x}}(k),\ \overline{\mathbf{K}} \in \text{Sparse}(\overline{\mathbf{E}}_c),$$

and each component is then applied to the system at the correct time-step. For the simulations, the weight matrices were set to $\mathbf{Q} = \mathbf{I}$ and $\mathbf{R} = \mathbf{I}, 10\mathbf{I}, 100\mathbf{I}$. Figure 13 depicts the trace of $\overline{\mathbf{P}}(k)$. Figure 14 depicts the response of the controller with the gains computed for $\mathbf{R} = \mathbf{I}$, implemented in Pump 1, to a sequence of periodic steps, and Figure 15 is the corresponding input signal generated. A detailed view that compares the results with the different weights can be seen in Figures 16 and 17. As it can be seen, with a higher weight the controller generates an input signal of smaller magnitude, with the trade-off of having a slower response to the steps.

7 | SIMULATION RESULTS FOR THE N-TANK PROCESS

In this section, further simulation results are presented, now for the case of a system of N interconnected water tanks. Given that this system is an extension of the QTP, the process of designing the local state observers and local controllers is very similar. The dynamics, filter design, and controller design are briefly presented for the general case of N tanks, and then, simulations results are shown for the specific case of $N = 40$.

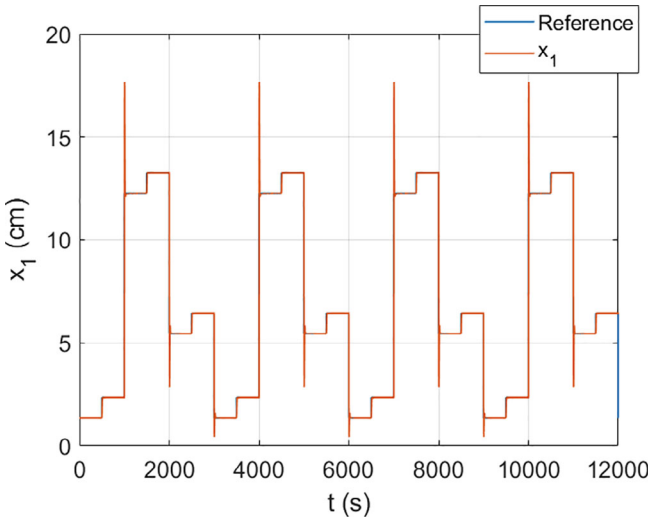

FIGURE 14 Response of the closed-loop system to a sequence of periodic steps

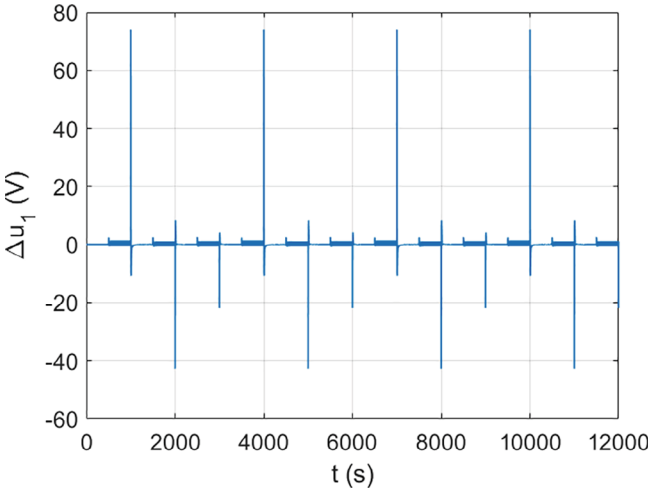

FIGURE 15 Input signal sent to Pump 1

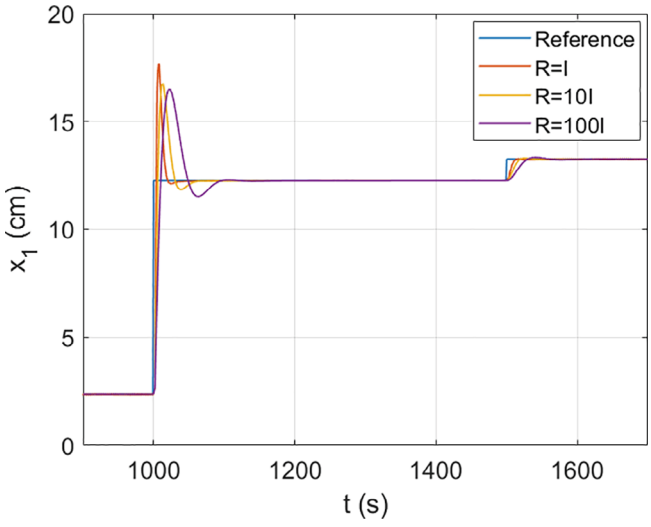

FIGURE 16 Comparison of the closed-loop system response with different weights

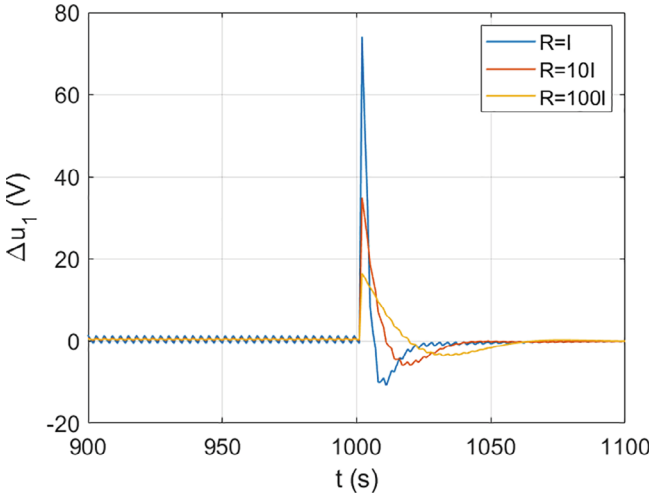

FIGURE 17 Comparison of the input signals generated by the controllers with different weights

7.1 | Dynamics of the N-tank process

Consider the system of N interconnected water tanks depicted in Figure 18. The nonlinear dynamics of this system are given by

$$\begin{cases} \dot{x}_1 = -\frac{r_1}{R_1}\sqrt{2gx_1} + \frac{r_{\frac{N}{2}+1}}{R_1}\sqrt{2gx_{\frac{N}{2}+1}} + \frac{\gamma_1 k_1}{R_1}u_1, \\ \dot{x}_2 = -\frac{r_2}{R_2}\sqrt{2gx_2} + \frac{r_{\frac{N}{2}+2}}{R_2}\sqrt{2gx_{\frac{N}{2}+2}} + \frac{\gamma_2 k_2}{R_2}u_2, \\ \vdots \\ \dot{x}_{\frac{N}{2}} = -\frac{r_{\frac{N}{2}}}{R_{\frac{N}{2}}}\sqrt{2gx_{\frac{N}{2}}} + \frac{r_N}{R_{\frac{N}{2}}}\sqrt{2gx_N} + \frac{\gamma_{\frac{N}{2}} k_{\frac{N}{2}}}{R_{\frac{N}{2}}}u_{\frac{N}{2}}, \\ \dot{x}_{\frac{N}{2}+1} = -\frac{r_{\frac{N}{2}+1}}{R_{\frac{N}{2}+1}}\sqrt{2gx_{\frac{N}{2}+1}} + \frac{(1-\gamma_{\frac{N}{2}})k_{\frac{N}{2}}}{R_{\frac{N}{2}+1}}u_{\frac{N}{2}}, \\ \dot{x}_{\frac{N}{2}+2} = -\frac{r_{\frac{N}{2}+2}}{R_{\frac{N}{2}+2}}\sqrt{2gx_{\frac{N}{2}+2}} + \frac{(1-\gamma_1)k_1}{R_{\frac{N}{2}+2}}u_1, \\ \vdots \\ \dot{x}_N = -\frac{r_N}{R_N}\sqrt{2gx_N} + \frac{(1-\gamma_{\frac{N}{2}-1})k_{\frac{N}{2}-1}}{R_N}u_{\frac{N}{2}-1}, \end{cases} \tag{45}$$

where x_i is the water level of tank i, R_i is the cross-section, and r_i is the cross-section of the outlet hole for $i = 1, \dots, N$. The voltage applied to pump i, $i = 1, \dots, \frac{N}{2}$, is u_i, and the corresponding flow is $k_i u_i$. This flow is controlled by the valves $\gamma_i \in (0, 1)$. The parameter values used in the simulations are presented in Table 7. The linearized system matrices follow

$$\mathbf{A}(t) = \begin{bmatrix} -\frac{1}{T_1} & 0 & \dots & 0 & \frac{R_{\frac{N}{2}+1}}{R_1 T_{\frac{N}{2}+1}} & 0 & \dots & 0 \\ 0 & \ddots & \ddots & & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & & \ddots & \ddots & 0 \\ \vdots & & \ddots & \ddots & \ddots & & \ddots & \frac{R_N}{R_{\frac{N}{2}} T_N} \\ \vdots & & & \ddots & \ddots & \ddots & & 0 \\ \vdots & & & & \ddots & \ddots & \ddots & \vdots \\ \vdots & & & & & \ddots & \ddots & 0 \\ 0 & \dots & \dots & \dots & \dots & \dots & 0 & -\frac{1}{T_N} \end{bmatrix},$$

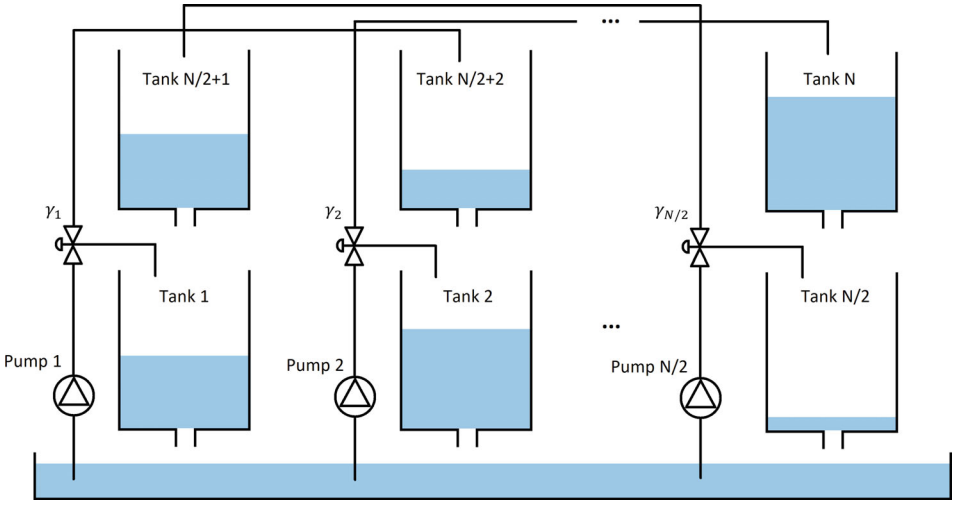

FIGURE 18 N-tank process

TABLE 7 Parameter values of the N-tank process used in the simulations

Parameter (unit)	Value
R_i (cm^2)	28
r_i (cm^2)	0.071
γ_i	0.7
g (cm/s^2)	981

$$\mathbf{B} = \begin{bmatrix} \frac{\gamma_1 k_1}{R_1} & 0 & \dots & \dots & 0 \\ 0 & \ddots & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ \vdots & & & 0 & \frac{\gamma_{\frac{N}{2}} k_{\frac{N}{2}}}{R_{\frac{N}{2}}} \\ 0 & \dots & \dots & 0 & \frac{(1-\gamma_{\frac{N}{2}})k_{\frac{N}{2}}}{R_{\frac{N}{2}+1}} \\ \frac{(1-\gamma_1)k_1}{R_{\frac{N}{2}+2}} & 0 & \dots & \dots & 0 \\ 0 & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \vdots \\ \vdots & \ddots & & & \vdots \\ 0 & \dots & 0 & \frac{(1-\gamma_{\frac{N}{2}-1})k_{\frac{N}{2}-1}}{R_N} & 0 \end{bmatrix},$$

where T_i is computed as in (36), for $i = 1, \dots, N$. The discrete-time dynamics $\mathbf{A}_d(k), \mathbf{B}_d(k)$, and $\mathbf{C}_d$ can be obtained by applying the transformation (39).

7.2 | Filter implementation

The local state observers are obtained by decoupling the system dynamics, similarly to the approach introduced in Section 6.3. The prediction step is given by

TABLE 8 Measured performance of the filters (sum of variances over one period) averaged over 1000 simulations

One-step ($T = 4$)	One-step ($T = 800$)	Finite-horizon
465.0487	624.7562	454.9105

$$\begin{cases} \Delta\hat{x}_1(k+1|k) = \mathbf{A}_d(k)^{(1,1)}\Delta\hat{x}_1(k|k) + \mathbf{A}_d(k)^{(1,\frac{N}{2}+1)}\Delta\hat{x}_{\frac{N}{2}+1}(k|k) \\ \qquad\qquad + \mathbf{B}_d(k)^{(1,1)}\Delta u_1(k) + \mathbf{B}_d(k)^{(1,\frac{N}{2})}\Delta u_{\frac{N}{2}}(k), \\ \vdots \\ \Delta\hat{x}_{\frac{N}{2}}(k+1|k) = \mathbf{A}_d(k)^{(\frac{N}{2},\frac{N}{2})}\Delta\hat{x}_{\frac{N}{2}}(k|k) + \mathbf{A}_d(k)^{(\frac{N}{2},N)}\Delta\hat{x}_N(k|k) \\ \qquad\qquad + \mathbf{B}_d(k)^{(\frac{N}{2},\frac{N}{2})}\Delta u_{\frac{N}{2}}(k) + \mathbf{B}_d(k)^{(\frac{N}{2},\frac{N}{2}-1)}\Delta u_{\frac{N}{2}-1}(k), \\ \Delta\hat{x}_{\frac{N}{2}+1}(k+1|k) = \mathbf{A}_d(k)^{(\frac{N}{2}+1,\frac{N}{2}+1)}\Delta\hat{x}_{\frac{N}{2}+1}(k|k) + \mathbf{B}_d(k)^{(\frac{N}{2}+1,\frac{N}{2})}\Delta u_{\frac{N}{2}}(k), \\ \vdots \\ \Delta\hat{x}_N(k+1|k) = \mathbf{A}_d(k)^{(N,N)}\Delta\hat{x}_N(k|k) + \mathbf{B}_d(k)^{(N,\frac{N}{2}-1)}\Delta u_{\frac{N}{2}-1}(k), \end{cases} \tag{46}$$

and the filtered state estimate follows

$$\Delta\hat{x}_i(k+1|k+1) = \Delta\hat{x}_i(k+1|k) + \mathbf{K}(k+1)^{(i,i)}(\Delta y_i(t_{k+1}) - \mathbf{C}_d^{(i,i)}\Delta\hat{x}_i(k+1|k)). \tag{47}$$

Then, the actual state estimates can be recovered with

$$\hat{x}_i(k+1|k+1) = \Delta\hat{x}_i(k+1|k+1) + x_{i_{\text{nom}}}. \tag{48}$$

7.3 | Filter simulation and Monte Carlo analysis

The simulations presented here were performed for the case of 40 tanks. Analogously to Section 6.4, a nominal trajectory was obtained by simulating the nonlinear dynamics (45) with a periodic input $\mathbf{u} = \mathbf{u}_{\text{nom}}$. Each component $u_i, i = 1, \dots, 20$, of the input signal was composed of 4 constant sections, each with a duration of 200 s. This results in a system with period $T = 800$, and 4 equilibrium points. The sampling time was, once again, set to $T_s = 1$ s. The measurements were corrupted by additive white Gaussian noise with zero-mean and covariance given by

$$\mathbf{R} = \begin{bmatrix} 0.1 & 0.05 & \dots & 0.05 \\ 0.05 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0.05 \\ 0.05 & \dots & 0.05 & 0.1 \end{bmatrix}.$$

The process noise covariance, which was used as a tuning parameter, was set to $\mathbf{Q} = 0.01\mathbf{I}$ and the sparsity pattern is given by $\mathbf{E} = \mathbf{I}$. To compare the performance of the competing solutions, 1000 simulations were carried out. Figure 19 depicts the total estimation error norm for the filter with the one-step gains, computed assuming a period of $T = 4$ and $T = 20$. A higher period was not considered due to the heavier computational requirements associated with this larger system. Nonetheless, the results obtained suggest that using a larger period to compute the gains leads to worse filter performance, which is in accordance with the results obtained in Section 6.5. The total estimation error norm for the filter with finite-horizon gains can be seen in Figure 20. The measured performance of the three competing solutions is shown in Table 8. These results further cement the finite-horizon method as the superior choice when filtering performance is the main priority. The one-step method also performs very well and, given the computational efficiency of this method, it is still a solid choice in most situations.

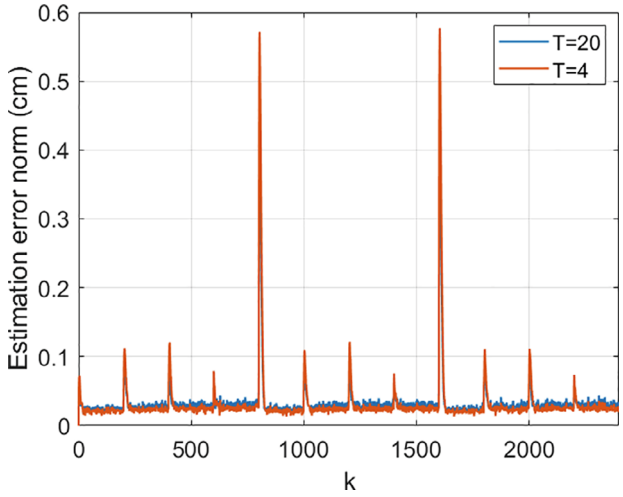

FIGURE 19 Total estimation error norm for the filter with the one-step gains ($T = 20$ and $T = 4$) averaged over 1000 simulations

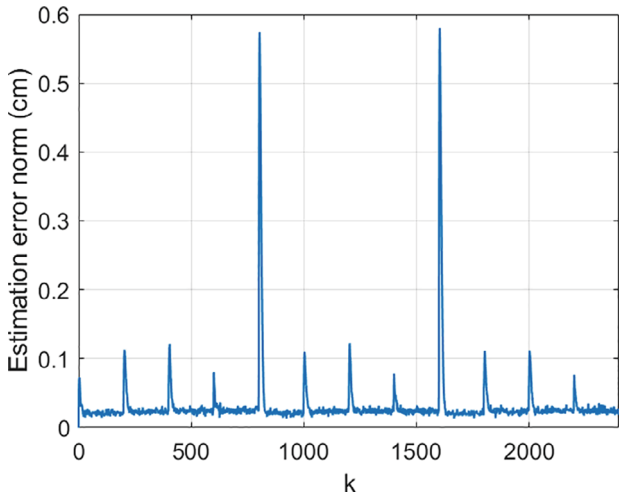

FIGURE 20 Total estimation error norm for the filter with finite-horizon gains averaged over 1000 simulations

7.4 | Control

To control the water level of the bottom tanks, a local controller is implemented at each pump. Since only $x_1, \ldots, x_{\frac{N}{2}}$ are relevant, the output matrix is reduced to

$$\mathbf{C} = \begin{bmatrix} \mathbf{I}_{\frac{N}{2}} & \mathbf{0}_{\frac{N}{2}} \end{bmatrix},$$

where $\mathbf{I}_{\frac{N}{2}}$ and $\mathbf{0}_{\frac{N}{2}}$ represent the identity and zero matrices of size $\frac{N}{2}$, respectively. This system was linearized about the equilibria depicted in Table 9 and then discretized with a sampling time of $T_s = 1$ s. Similarly to the approach introduced for the QTP in Section 6.6, the system was augmented with $\frac{N}{2}$ integral states. The resulting system has the same structure as (43), and is repeated here for clarity

$$\begin{cases} \begin{bmatrix} \Delta\mathbf{x}(k+1) \\ \mathbf{q}(k+1) \end{bmatrix} = \begin{bmatrix} \mathbf{A}_d(k) & \mathbf{0} \\ \mathbf{C}_d & \mathbf{I} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{x}(k) \\ \mathbf{q}(k) \end{bmatrix} + \begin{bmatrix} \mathbf{B}_d(k) & \mathbf{0} \\ \mathbf{0} & -\mathbf{I} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{u}(k) \\ \mathbf{r}(k) \end{bmatrix}, \\ \Delta\mathbf{y}(k) = \mathbf{C}_d \Delta\mathbf{x}(k). \end{cases} \tag{49}$$

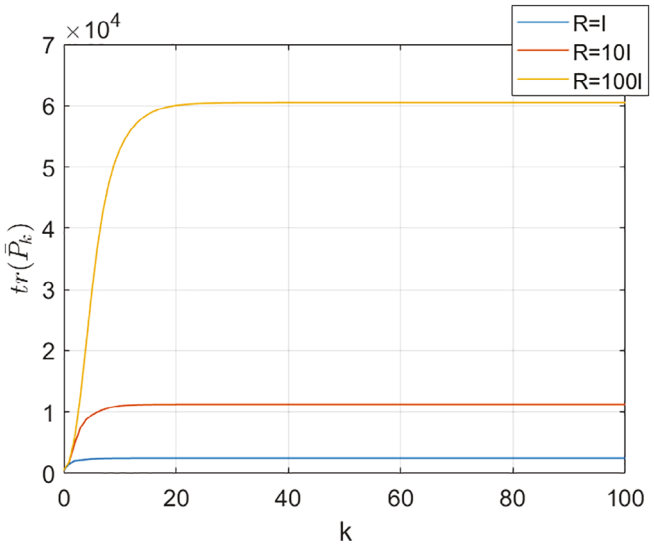

FIGURE 21 Trace of $\overline{\mathbf{P}}_k$

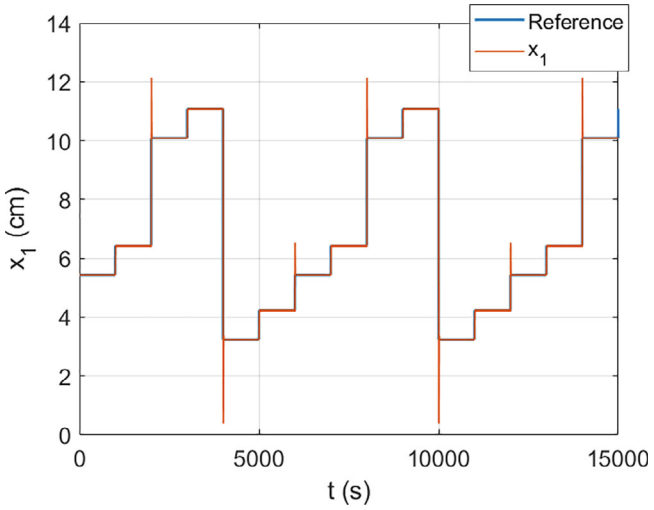

FIGURE 22 Response of the closed-loop system to a sequence of periodic steps

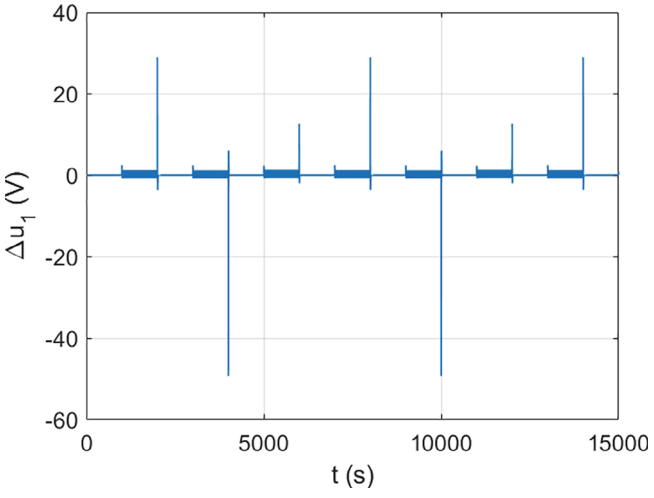

FIGURE 23 Input signal sent to Pump 1

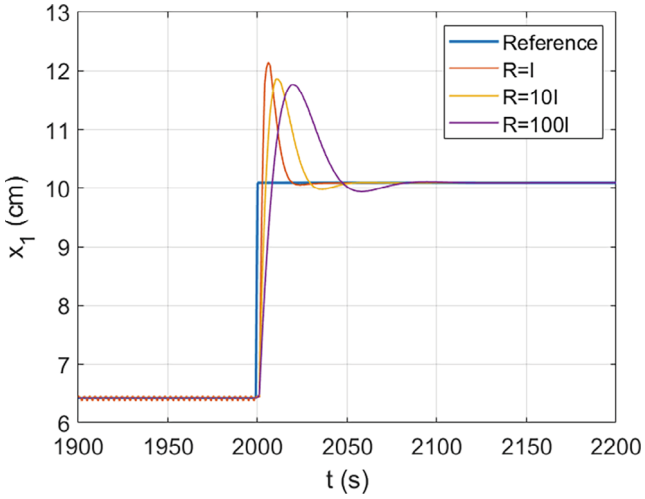

FIGURE 24 Comparison of the closed-loop system response with different weights

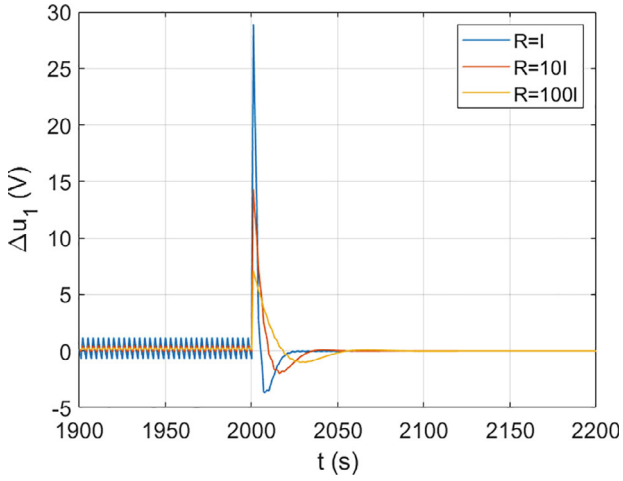

FIGURE 25 Comparison of the input signals generated by the controllers with different weights

TABLE 9 Equilibrium points

	$\mathbf{Eq_0}$	$\mathbf{Eq_1}$	$\mathbf{Eq_2}$
$x_i, i = 1, \dots, \frac{N}{2}$ (cm)	5.4265	10.0906	3.2402
$x_i, i = \frac{N}{2} + 1, \dots, N$ (cm)	0.4884	0.9082	0.2916
$u_i, i = 1, \dots, \frac{N}{2}$ (V)	2.2	3	1.7

Given that each local controller has access only to the water level x_i and its respective integral state q_i, the sparsity pattern follows

$$\mathbf{E}_c = \begin{bmatrix} \mathbf{I}_{\frac{N}{2}} & \mathbf{0}_{\frac{N}{2}} & \mathbf{I}_{\frac{N}{2}} \\ \mathbf{0}_{\frac{N}{2}} & \mathbf{0}_{\frac{N}{2}} & \mathbf{0}_{\frac{N}{2}} \end{bmatrix}.$$

In the simulations that follow, $N = 40$ tanks were considered. Figure 21 represents the trace of $\overline{\mathbf{P}}_k$ associated with the gains computed using one-step method, with $\mathbf{Q} = \mathbf{I}$ and $\mathbf{R} = \mathbf{I}, 10\mathbf{I}, 100\mathbf{I}$. Figures 22 and 23 depict the evolution of x_1 and Δu_1 in response to a sequence of periodic steps, for the controller implemented with the gains computed for $\mathbf{R} = \mathbf{I}$. A more detailed view can be seen in Figures 24 and 25, also showcasing the effect of varying the value of $\mathbf{R}$.

8 | CONCLUSIONS

This article addressed the problems of decentralized control and state estimation for LTP systems. The estimation problem was formulated as a discrete-time Kalman filter with sparsity constraints on the gains and, based on a LTI reformulation of the periodic dynamics, two methods were presented. The first method is an adaptation of the classic Kalman filter equations for LTI systems to decentralized LTP systems, making this a well-performing and very computationally efficient algorithm. The second method solves an optimization problem over a finite-time window to approximate steady-state behavior and compute well-performing gains. The control problem was formulated as an infinite-horizon cost optimization for the lifted dynamics, and was then reformulated as the minimization of a difference matrix equation, subject to a sparsity constraint. The one-step method, which computes the gains very efficiently, by minimizing the trace of the difference equation, was presented. The proposed algorithms were particularized to the case of a system of four interconnected water tanks, and simulation results were presented to assess the performance of the proposed solutions. Further simulation results were presented for an extended case where 40 interconnected tanks were considered. In regards to state estimation, it was shown that the finite-horizon method achieves better filtering performance than the one-step method, at the cost of being computationally heavier. The control solution that was proposed performs adequately and it was shown that the controller gains can be manually optimized by adjusting the weight matrices of the cost function.

ACKNOWLEDGMENTS

This work was supported by the Fundação para a Ciência e a Tecnologia (FCT) through LARSyS - FCT Project UIDB/50009/2020, IDMEC through LAETA FCT project UIDB/50022/2020, and through the FCT project DECENTER [LISBOA-01-0145-FEDER-029605], funded by the Lisboa 2020 and PIDDAC programs; and by the Macau Science and Technology Development Fund under Grants FDCT/0146/2019/A3 and FDCT/0031/2020/AFJ.

CONFLICT OF 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 article.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

ORCID

Pedro Batista https://orcid.org/0000-0001-6079-0436

REFERENCES

1. Bittanti S, Colaneri P. *Periodic Systems: Filtering and Control*. Springer Science & Business Media; 2009.
2. Genta G. *Dynamics of Rotating Systems*. Springer Science & Business Media; 2007.
3. Allen M, Ginsberg JH. Floquet modal analysis to detect cracks in a rotating shaft on anisotropic supports. Proceedings of the 24th International Modal Analysis Conference (IMAC XXIV); 2006.
4. Patel Y. *The Design and Analysis of Multirate Control Systems*. Ph.D dissertation. University of York; 1991.
5. Martel F, Pal P, Psiaki M. Active magnetic control system for gravity gradient stabilized spacecraft. Proceedings of the 2nd Annual AIAA/USU Conference on Small Satellites; 1988.
6. Assimakis N, Adam M. Steady state Kalman filter for periodic models: a new approach. *Int J Contemp Math Sci*. 2009;4(5):201-218.
7. Hashemipour HR, Roy S, Laub AJ. Decentralized structures for parallel Kalman filtering. *IEEE Trans Automat Contr*. 1988;33(1):88-94.
8. Oruc S, Sijs S, Van den Bosch P. Optimal decentralized Kalman filter. Proceedings of the 2009 17th Mediterranean Conference on Control and Automation; 2009:803-808; IEEE.
9. Viegas D, Batista P, Oliveira P, Silvestre C. Discrete-time distributed Kalman filter design for formations of autonomous vehicles. *Control Eng Pract*. 2018;75:55-68.
10. Olfati-Saber R. Kalman-consensus filter: optimality, stability, and performance. Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference; 2009:7036-7042; IEEE.
11. Li X, Caimou H, Haoji H. Distributed filter with consensus strategies for sensor networks. *J Appl Math*. 2013;2013:683249.
12. Lee SH, West M. Performance comparison of the distributed extended Kalman filter and Markov chain distributed particle filter (MCDPF). *IFAC Proc Vol*. 2010;43(19):151-156.
13. Cattivelli FS, Sayed AH. Distributed nonlinear Kalman filtering with applications to wireless localization. Proceedings of the 2010 IEEE International Conference on Acoustics, Speech and Signal Processing; 2010:3522-3525; IEEE.

14. Wu J, Li L, Ugrinovskii V, Allgower F. Distributed filter design for cooperative ho-type estimation. Proceedings of the 2015 IEEE Conference on Control Applications (CCA); 2015:1373-1378; IEEE.
15. Li Z, Yin X, Yin X, Xie Y, Wang C. Distributed h_∞ filtering over multiple-channel sensor networks with Markovian channel switching and time-varying delays. Proceedings of the 2015 54th IEEE Conference on Decision and Control (CDC); 2015:7410-7415; IEEE.
16. Enayat M, Khorasani K. Semi-decentralized nonlinear cooperative control strategies for a network of heterogeneous autonomous underwater vehicles. *Int J Robust Nonlinear Control*. 2017;27(16):2688-2707.
17. Wu B, Wang D, Poh E. Decentralized sliding-mode control for attitude synchronization in spacecraft formation. *Int J Robust Nonlinear Control*. 2013;23(11):1183-1197.
18. Garcia E, Antsaklis PJ. Model-based control using a lifting approach. Proceedings of the 18th Mediterranean Conference on Control and Automation, MED'10; 2010:105-110; IEEE.
19. Garcia E, Antsaklis P. Model-based control of networked distributed systems with multi-rate state feedback updates. *Int J Control*. 2013;86(9):1503-1517.
20. Khargonekar P, Ozguler A. Decentralized control and periodic feedback. *IEEE Trans Automat Contr*. 1994;39(4):877-882.
21. Weihua Z, Go TH. Robust decentralized formation flight control. *Int J Aerosp Eng*. 2011;2011:157590.
22. Grosso JM, Ocampo-Martinez C, Puig V. A distributed predictive control approach for periodic flow-based networks: application to drinking water systems. *Int J Syst Sci*. 2017;48(14):3106-3117.
23. Massioni P, Keviczky T, Gill E, Verhaegen M. A decomposition-based approach to linear time-periodic distributed control of satellite formations. *IEEE Trans Control Syst Technol*. 2010;19(3):481-492.
24. Ayatollahi M, Majd VJ, Nasrabadi MM. An optimal cooperative control design for state consensus of periodic multiagent systems. *Math Probl Eng*. 2018;2018:3183278.
25. Blondel V, Tsitsiklis J. A survey of computational complexity results in systems and control. *Automatica*. 2000;36(9):1249-1274.
26. Witsenhausen H. A counterexample in stochastic optimum control. *SIAM J Control*. 1968;66(1):131-147.
27. Lessard L, Lall S. Convexity of decentralized controller synthesis. *IEEE Trans Automat Contr*. 2015;61(10):3122-3127.
28. Bittanti S, Colaneri P. Invariant representations of discrete-time periodic systems. *Automatica*. 2000;36(12):1777-1793.
29. Yang Y. An efficient LQR design for discrete-time linear periodic system based on a novel lifting method. *Automatica*. 2018;87:383-388.
30. Viegas D, Batista P, Oliveira P, Silvestre C. Distributed controller design and performance optimization for discrete-time linear systems. *Opt Control Appl Methods*. 2020; (in press).
31. Johansson KH. The quadruple-tank process: A multivariable laboratory process with an adjustable zero. *IEEE Trans Control Syst Technol*. 2000;8(3):456-465.
32. Pedroso L, Batista P. Efficient algorithm for the computation of the solution to a sparse matrix equation in distributed control theory. *Mathematics*. 2021;9(13):1497.
33. Simon D. *Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches*. John Wiley & Sons; 2006:26-27.

How to cite this article: Andrushka I, Batista P, Oliveira P, Silvestre C. Decentralized control and state estimation of linear time-periodic systems. *Int J Robust Nonlinear Control*. 2023;33(1):102-133. doi: 10.1002/rnc.6130

APPENDIX . PROOF OF THE LIFTED SPARSITY PATTERN

Consider the LTP system with period $T = 3$ given by

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{A}_k\mathbf{x}_k + \mathbf{B}_k\mathbf{u}_k, \\ \mathbf{u}_k = -\mathbf{K}_k\mathbf{x}_k. \end{cases} \tag{A1}$$

The control signal can be expanded as follows

$$\mathbf{u}_0 = -\mathbf{K}_0\mathbf{x}_0,$$

$$\mathbf{u}_1 = -\mathbf{K}_1\mathbf{x}_1,$$

$$\mathbf{u}_2 = -\mathbf{K}_2\mathbf{x}_2,$$

$$\mathbf{u}_3 = -\mathbf{K}_0\mathbf{x}_3,$$

$$\mathbf{u}_4 = -\mathbf{K}_1\mathbf{x}_4,$$

$$\mathbf{u}_5 = -\mathbf{K}_2\mathbf{x}_5,$$

and so forth. Now doing the same for the lifted input yields

$$\begin{bmatrix}\mathbf{u}_0\\ \mathbf{u}_1\\ \mathbf{u}_2\end{bmatrix} = -\begin{bmatrix}\mathbf{K}_0^{11} & \mathbf{K}_0^{12} & \mathbf{K}_0^{13}\\ \mathbf{K}_1^{21} & \mathbf{K}_1^{22} & \mathbf{K}_1^{23}\\ \mathbf{K}_2^{31} & \mathbf{K}_2^{32} & \mathbf{K}_2^{33}\end{bmatrix}\begin{bmatrix}\mathbf{0}\\ \mathbf{0}\\ \mathbf{x}_0\end{bmatrix},$$
$$\begin{bmatrix}\mathbf{u}_3\\ \mathbf{u}_4\\ \mathbf{u}_5\end{bmatrix} = -\begin{bmatrix}\mathbf{K}_0^{11} & \mathbf{K}_0^{12} & \mathbf{K}_0^{13}\\ \mathbf{K}_1^{21} & \mathbf{K}_1^{22} & \mathbf{K}_1^{23}\\ \mathbf{K}_2^{31} & \mathbf{K}_2^{32} & \mathbf{K}_2^{33}\end{bmatrix}\begin{bmatrix}\mathbf{x}_1\\ \mathbf{x}_2\\ \mathbf{x}_3\end{bmatrix},$$

and so forth. To further simplify the discussion, consider only the first period. The lifted input can be worked out to

$$\begin{aligned}\mathbf{u}_0 &= -\mathbf{K}_0^{13}\mathbf{x}_0,\\ \mathbf{u}_1 &= -\mathbf{K}_1^{23}\mathbf{x}_0,\\ \mathbf{u}_2 &= -\mathbf{K}_2^{33}\mathbf{x}_0.\end{aligned}$$

Comparing the lifted and periodic inputs makes it obvious that they are computed differently, given that the lifted input is computed using only $\mathbf{x}_0$. However, it is important to note that the gains that minimize the control optimization problem depend on the lifted matrices $\overline{\mathbf{A}}$ and $\overline{\mathbf{B}}$, which encode the system dynamics and input over a full period. With that in mind, and due to the special structure of the dynamics matrix $\overline{\mathbf{A}}$, it becomes apparent that the gains $\mathbf{K}_1^{23}$ and $\mathbf{K}_2^{33}$ are computed with the dynamics and input associated with following time-steps, thus having the state transition effect embedded in their design. To enforce that the following periods are consistent with the first, the lifted gain matrix must have the structure

$$\overline{\mathbf{K}} = \begin{bmatrix}\mathbf{0} & \mathbf{0} & \mathbf{K}_0^{13}\\ \mathbf{0} & \mathbf{0} & \mathbf{K}_1^{23}\\ \mathbf{0} & \mathbf{0} & \mathbf{K}_2^{33}\end{bmatrix}.$$

For the centralized case, this lifted structure leads to a lifted control law equivalent to the periodic control law, as shown in Reference 29. This result can be further extended to the decentralized case by enforcing that the lifted sparsity pattern has the structure

$$\overline{\mathbf{E}}_c = \begin{bmatrix}\mathbf{0} & \mathbf{0} & \mathbf{E}_c\\ \mathbf{0} & \mathbf{0} & \mathbf{E}_c\\ \mathbf{0} & \mathbf{0} & \mathbf{E}_c\end{bmatrix}.$$