Set-membership-based distributed moving horizon estimation of large-scale systems

Segovia, Pablo; Puig, Vicenç; Duviella, Eric

DOI
10.1016/j.isatra.2021.10.036

Publication date
2022

Document Version
Final published version

Published in
ISA Transactions

ISA Transactions 128 (2022) 402–413

Contents lists available at ScienceDirect

ISA Transactions

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

Research article

Set-membership-based distributed moving horizon estimation of large-scale systems

Pablo Segovia [a,*], Vicenç Puig [b,c], Eric Duviella [d]

[a] Department of Maritime and Transport Technology, Delft University of Technology, Delft, The Netherlands
[b] Research Center for Supervision, Safety and Automatic Control (CS2AC), Universitat Politècnica de Catalunya (UPC), Rambla Sant Nebridi 22, 08222 Terrassa, Spain
[c] Institut de Robòtica i Informàtica Industrial, CSIC-UPC, Llorens i Artigas 4-6, 08028 Barcelona, Spain
[d] CERI Digital Systems, IMT Nord Europe, Univ. Lille, F-59000 Lille, France

ARTICLE INFO

Article history:
Received 28 July 2021
Received in revised form 27 October 2021
Accepted 29 October 2021
Available online 15 November 2021

Keywords:
Large-scale systems
Distributed state estimation
Moving horizon estimation
Set-membership
Optimality condition decomposition
Community detection

ABSTRACT

This work is concerned with the design of a two-step distributed state estimation scheme for large-scale systems in the presence of unknown-but-bounded disturbances and noise. The set-membership approach is employed to construct a compact set containing the states consistent with system measurements and bounded noise and disturbances. The tightened feasible region is then provided to a moving horizon estimator that determines the optimal state estimates. Partitioning of the overall problem and coordination of the resulting subproblems are achieved using decomposition of the optimality conditions and community detection. The proposed strategy is tested on a case study based on a reactor–separator system widely used in the literature. Its performance is compared to those of centralized and distributed (without set-membership) implementations, allowing to highlight its effectiveness.

1. Introduction

State estimation is of paramount importance in engineering, and has received considerable attention from the research community for many decades. Research on the topic builds upon the seminal results on Kalman filtering, which uses probabilistic assumptions on disturbances and noise to minimize the error variance of the state estimate [1,2]. Nevertheless, these assumptions might be unrealistic and difficult to validate in real applications [3]. In this context, the set-membership approach does not resort to assumptions on statistical properties, and norm-bounded uncertainty can be considered instead [4]. Its main principle consists in building compact sets that bound the system states consistent with the norm-bounded uncertainty and the measurements [5]. On the other hand, Kalman filtering is not concerned with the issue of constraints, thus not making use of available physical and operational insight [6]. Moving horizon estimation (MHE), an approach that is deemed to be the dual of model predictive control (MPC), provides a framework that can deal with constraints in a straightforward manner [7]. The main principle consists in using an estimation window of constant size, which is shifted in time, so as to process only the most recent information and thus keep the problem computationally tractable [8,9].

Regardless of the choice of state estimation approach, technological developments have resulted in large-scale systems, which usually consist of multiple interacting subsystems and are characterized by a growing complexity of operation [10]. Therefore, centralized implementations present several important drawbacks in the case of large-scale systems, such as reliability (single point of failure) and non-scalability [11]. However, as a result of advances in information and communication technologies, non-centralized strategies are possible nowadays, thus improving traditional plant-wide model-based control and monitoring [12].

Non-centralized approaches assume that a set of agents is deployed, each in charge of a subsystem, or part of the overall system. Moreover, they are usually categorized into two main groups, depending on information availability and interactions among the local agents. On the one hand, decentralized approaches ignore interactions among subsystems, which might yield poor overall performances [13]. On the other hand, distributed approaches account for interactions in the design of the subproblems, introducing cooperation and negotiation mechanisms so as to achieve optimal global performance [14]. Given the superior performance of distributed over decentralized approaches, the former are usually preferred when the degree of coupling is not negligible, although this is achieved at the expense

* Corresponding author.
E-mail addresses: p.segoviacastillo@tudelft.nl (P. Segovia), vicenc.puig@upc.edu (V. Puig), eric.duviella@imt-nord-europe.fr (E. Duviella).

https://doi.org/10.1016/j.isatra.2021.10.036

of computation time. Distributed state estimation has been extensively studied, considering both set membership [15–18] and MHE [19–22] approaches.

Distributed state estimation approaches require, first of all, to split the problem into smaller subproblems. However, decomposition is a sensitive issue, as different decompositions can lead to rather different overall performances and computation times [23]. In this regard, graph theory and network science have recently examined the property of community structure in networked systems, where systems with higher community structures exhibit nodes in tight clusters among which there are fewer links [24]. Metrics to quantify this property have been defined, and algorithms to generate system decompositions that maximize such metrics have been devised. These tools can be of interest to decompose state estimation problems, as the resulting subproblems are loosely coupled, thus keeping information exchanges to a minimum.

Once the subproblems have been generated, these can be solved in a distributed fashion. There exist multiple distributed optimization techniques, which may be categorized into two main classes: on the one hand, approaches based on augmented Lagrangian decomposition; on the other hand, techniques that employ the decentralized solution of the Karush–Kuhn–Tucker (KKT) optimality conditions [25]. Examples of the former and the latter are the alternating direction method of multipliers (ADMM) [26] and the optimality condition decomposition (OCD) method [27], respectively. The main differences between these two approaches reside in the construction of the Lagrangian function and the management of coupled variables. On the one hand, the Lagrangian function used in ADMM is augmented with supplementary terms that are linked to the constraint residuals [26], while OCD keeps complicating constraints, i.e., constraints including coupled variables, in the assigned subproblem but also relaxes them in the coupled subproblems [28]. On the other hand, ADMM creates copies of coupled variables in all coupled subproblems while adding equality constraints to guarantee solution compatibility among subproblems, thus requiring a central coordinator. In contrast, each variable is assigned in OCD to exactly one subproblem and optimizes its value in that subproblem, thus overriding the need for a central coordinator [25]. Despite this, the literature review reveals that ADMM has been more widely applied than OCD to solve distributed state estimation problems. Indeed, while the former has been used in, e.g., [22,29–31], no reference making an explicit use of the latter has been found, although it shares some similarities with the method reported in [32].

The combination of the aforementioned set-membership and MHE allows for robust state estimation, a problem has been addressed from multiple different perspectives. The authors of [33] made use of multi-parametric methods to generate the dynamic equations and estimation error bounds for a linear constrained MHE. An initial high-gain observer was proposed in [34] to compute confidence regions that contain the actual system state, feeding this information to an MHE. The authors of [35] designed an MHE that uses the convex hull to compute guaranteed bounds for the state estimates. The design of an MHE with real-time adaptive update of error variances was tackled in [36].

The results reported in the previous references were designed considering a centralized architecture. Non-centralized implementations of robust state estimators based on MHE can also be found in the literature, although it must be noted that none employs the set-membership approach. A robust distributed MHE (DMHE) for nonlinear constrained systems was proposed in [37], adding a consensus term to propagate information regarding local estimates. The authors of [29] designed a DMHE for joint identification of corrupted SCADA measurements (as a result of unintentional metering faults) and state estimation solving the relaxed ℓ_1-norm problem. Another approach considering the ℓ_1-norm was devised in [31], also under the assumption of compressed sensing. The authors of [38] designed local MHE featuring complementary nonlinear observers to track the nominal states. A partitioned MHE using the method of largest normalized residuals was conceived in [30] to deal with outliers. Another DMHE with guaranteed robustness to outliers introduced in cyberattacks was developed in [22] using total-variation denoising and ℓ_1 trend filtering.

Summary of the paper and contribution

This work reports the derivation of a two-step distributed state estimation algorithm considering an unknown-but-bounded description of disturbances and noise. As a first step, set-membership is used to tighten the physical and operational bounds on states, making use of information regarding measurements and uncertainty. These results are then provided to the MHE, which benefits from the reduced feasible region computed in the previous step, and whose solution yields the optimal state estimates. Moreover, a community detection algorithm and OCD are used to generate the subproblems and coordinate the solutions. The performance is then tested on a case study based on a large-scale reactor–separator plant widely used in the literature.

Contributions of the paper with regard to the state of the art are detailed next:

- The current paper builds on previous results reported in [39] concerning the design of a distributed control approach using OCD and community detection. Indeed, the main findings are used to consider a different problem, i.e., state estimation. To the best knowledge of the authors, the combined use of OCD and community detection for distributed state estimation is a novel approach.
- Moreover, the effect of disturbances and noise was not studied in [39]. In contrast, this issue is explicitly addressed in this paper, using the set-membership approach to tighten the feasible region considering only the states that are consistent with measurements and the level of uncertainty. This facilitates the task of the MHE afterward. The review of the literature confirms that a robust distributed state estimation approach based on the combination of set-membership and moving horizon estimation has not been proposed before.
- While the goal of bound tightening is shared with other works [40–43], these references consider the output feedback control problem. Therefore, results on bound tightening are used to estimate the future evolution of the system by means of a Luenberger observer, which introduces the additional issue of appropriately tuning an observer gain [44]. In contrast, this paper makes use of bound tightening results for a different purpose, i.e., state estimation using an MHE, which overrides the need to tune the observer gain.
- Furthermore, the previous references propose centralized implementations. Conversely, this is carried out in a distributed manner in this work, which gives rise to a distributed set-membership-based MHE algorithm, hereinafter referred to as DMHE-SM.

The structure of the paper is as follows: Section 2 describes the problem and presents the rationale behind the proposed approach. Section 3 details the overall problem decomposition and coordination approach, which allows to devise the two-step distributed state estimation algorithm in Section 4. Finally, Section 5 introduces the case study that is used to test and validate the approach, thus allowing to derive conclusions and elaborate on possible research directions in Section 6.

Notation

Let $\mathbb{Z}_{\geq 0}$, $\mathbb{R}^n$ and $\mathbb{R}^{n\times m}$ denote the set of natural non-negative scalars, the space of n-dimensional real column vectors and the space of n-by-m real matrices, respectively. Moreover, $\mathbf{A}_i$ denotes the ith row of matrix $\mathbf{A}$. Scalars, vectors and matrices are represented by either lowercase or uppercase letters, bold lowercase letters and bold uppercase letters, respectively, while sets are denoted with calligraphic symbols. Furthermore, let $\mathcal{G} = (\mathcal{N}, \mathcal{E})$ represent an undirected and connected graph, where $\mathcal{N} = \{1, 2..., N\}$ and $\mathcal{E} = \{(i, j) : i, j \in \mathcal{N}\} \subseteq \mathcal{N} \times \mathcal{N}$ denote the sets of nodes and edges, respectively. Then, the set of neighbors of node i is denoted by $\mathcal{N}^{(i)} = \{j : (i, j) \in \mathcal{E}\}$.

2. Problem statement

The problem tackled in this work regards the class of discrete-time invariant linear uncertain systems

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

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{v}_k, \tag{1b}$$

where $\mathbf{x}_k \in \mathbb{R}^{n_x}$, $\mathbf{u}_k \in \mathbb{R}^{n_u}$ and $\mathbf{y}_k \in \mathbb{R}^{n_y}$ represent the state, input and output vectors, respectively, with $k \in \mathbb{Z}_{\geq 0}$ the sample time. Moreover, $\mathbf{w}_k \in \mathbb{R}^{n_x}$ and $\mathbf{v}_k \in \mathbb{R}^{n_y}$ denote disturbance and noise vectors, respectively, and are considered to be unknown but bounded by known compact sets $\mathbf{w}_k \in \mathcal{W}$ and $\mathbf{v}_k \in \mathcal{V}$ that contain the respective origins. Furthermore, the state-space matrices $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$ are of appropriate dimensions.

Given (1) and a set of input–output data, the goal consists in designing a state estimation strategy that allows to fully reconstruct the vector of states at each time instant, as they are seldom completely available for measurement. It should be noted that data generation is assumed to be out of the scope of the problem. Then, a centralized MHE (CMHE) can be designed for (1) as follows:

$$\min_{\{\hat{\mathbf{x}}_{i|k}\}_{i=k-N+1}^{k}} \left(\hat{\mathbf{x}}_{k-N+1|k} - \mathbf{x}_{k-N+1}\right)^{\intercal} \mathbf{P}^{-1} \left(\hat{\mathbf{x}}_{k-N+1|k} - \mathbf{x}_{k-N+1}\right) + \sum_{i=k-N+1}^{k-1} \left(\mathbf{w}_{i|k}^{\intercal}\mathbf{Q}^{-1}\mathbf{w}_{i|k} + \mathbf{v}_{i|k}^{\intercal}\mathbf{R}^{-1}\mathbf{v}_{i|k}\right) \tag{2a}$$

subject to

$$\mathbf{w}_{i|k} = \hat{\mathbf{x}}_{i+1|k} - \left(\mathbf{A}\hat{\mathbf{x}}_{i|k} + \mathbf{B}\mathbf{u}_i\right), \ i \in \{k-N+1, \ldots, k\}, \tag{2b}$$

$$\mathbf{v}_{i|k} = \mathbf{y}_i - \mathbf{C}\hat{\mathbf{x}}_{i|k}, \ i \in \{k-N+1, \ldots, k\}, \tag{2c}$$

$$\hat{\mathbf{x}}_{i|k} \in \mathcal{X}, \ i \in \{k-N+1, \ldots, k\}, \tag{2d}$$

where $\left\{\hat{\mathbf{x}}_{i|k}\right\}_{i=k-N+1}^{k}$ denotes the sequence of state estimates[1] for (1) that are most consistent with the provided input–output data[2] $\{(\mathbf{u}_i, \mathbf{y}_i)\}_{i=k-N+1}^{k}$, with N the length of the moving estimation window. Moreover, $\mathbf{x}_{k-N+1}$ denotes the most reasonable initial state (which may be selected based on knowledge of the system), while $\mathcal{X}$ represents the feasible set of states according to operational and physical constraints. Furthermore, $\mathbf{P}^{-1}$, $\mathbf{Q}^{-1}$ and $\mathbf{R}^{-1}$ are weighting matrices inverses, and indicate confidence in the initial state, quality of the model and the measurements, respectively [7].

The last component of the optimal sequence $\left\{\hat{\mathbf{x}}_{i|k}\right\}_{i=k-N+1}^{k}$, i.e., $\hat{\mathbf{x}}_{k|k}$, is retained, and the rest are discarded. The window is then shifted forward in time to utilize updated information, hence converting the initial open-loop approach into a closed-loop one.

[1] $\left\{\hat{\mathbf{x}}_{i|k}\right\}_{i=k-N+1}^{k} \triangleq \left\{\hat{\mathbf{x}}_{k-N+1|k}, \hat{\mathbf{x}}_{k-N+2|k}, \ldots, \hat{\mathbf{x}}_{k|k}\right\}$

[2] $\{(\mathbf{u}_i, \mathbf{y}_i)\}_{i=k-N+1}^{k-1} \triangleq \{(\mathbf{u}_{k-N+1}, \mathbf{y}_{k-N+1}), (\mathbf{u}_{k-N+2}, \mathbf{y}_{k-N+2}), \ldots, (\mathbf{u}_k, \mathbf{y}_k)\}$

Despite the fact that (1) describes a broad class of systems, state estimation of large-scale systems is specifically addressed. Then, and owing to the particular features of large-scale systems, non-centralized implementations are preferred over their centralized counterparts. In particular, distributed schemes allow for interactions among local agents, each in charge of only a part of the overall system. These local agents exchange information with each other in an iterative manner until convergence.

Distributed schemes offer an interesting alternative to centralized approaches given the minor loss of performance that comes at the benefit of reduced computation times. Indeed, performing a single iteration in a distributed approach is usually much faster than solving the overall problem. As the subproblems can be solved in parallel, the duration of a single iteration amounts to the time required to solve the largest subproblem. However, the choice of the convergence error threshold has implications on the amount of iterations needed. Although this number could be reduced by increasing the threshold, this is not desirable as it leads to a degraded performance. Instead, the physical feasible region can be further constrained by eliminating solutions that are inconsistent with the measured outputs and the noise bounds. Such strategy is known as the set-membership approach.

With all this, the proposed solution consists in designing a DMHE enhanced with a set-membership scheme that computes tightened bounds at each time instant (DMHE-SM). Nevertheless, overall problem decomposition into subproblems and definition of the coordination policy need to be carried out first. Hence, these are discussed in the next section. The derivation of the DMHE-SM will be tackled afterward.

3. Decomposition and coordination via OCD and community detection

3.1. Optimality condition decomposition

The OCD is considered to be a particular Lagrangian relaxation implementation. As its name suggests, it makes use of the optimality conditions of a problem to divide it into smaller subproblems. Moreover, not only does it allow for overall system partitioning, but also determines the coordination policy among subproblems [25,45].

The centralized estimation problem (2) is restated for convenience:

$$\min_{\hat{\mathbf{x}}} \ f(\hat{\mathbf{x}}) \tag{3a}$$

subject to

$$\mathbf{a}\left(\hat{\mathbf{x}}\right) = \mathbf{0}, \tag{3b}$$

$$\mathbf{b}\left(\hat{\mathbf{x}}\right) \leq \mathbf{0}, \tag{3c}$$

where $\hat{\mathbf{x}} \in \mathbb{R}^{n_x}$, $f\left(\hat{\mathbf{x}}\right) : \mathbb{R}^{n_x} \to \mathbb{R}$, $\mathbf{a}\left(\hat{\mathbf{x}}\right) : \mathbb{R}^{n_x} \to \mathbb{R}^{n_a}$ and $\mathbf{b}\left(\hat{\mathbf{x}}\right) : \mathbb{R}^{n_x} \to \mathbb{R}^{n_b}$.

Without loss of generality, (3) can be further simplified by only retaining the inequality constraints. Indeed, those cases featuring equality constraints can be dealt with in an analogous fashion [45]. The problem is restated as shown below:

$$\min_{\hat{\mathbf{x}}} \ f(\hat{\mathbf{x}}) \tag{4a}$$

subject to

$$\mathbf{b}\left(\hat{\mathbf{x}}\right) \leq \mathbf{0}. \tag{4b}$$

The OCD assumes that (4) can be divided into subproblems. To illustrate this, the following reformulation is introduced:

$$\min_{\{\hat{\mathbf{x}}^{(i)}\}_{i=1}^{L}} \sum_{i=1}^{L} f^{(i)}\left(\hat{\mathbf{x}}^{(i)}\right) \tag{5a}$$

subject to

$$\mathbf{h}\left(\hat{\mathbf{x}}^{(1)},\ldots,\hat{\mathbf{x}}^{(L)}\right)\leq\mathbf{0}, \tag{5b}$$

$$\mathbf{g}^{(i)}\left(\hat{\mathbf{x}}^{(i)}\right)\leq\mathbf{0}, \qquad i\in\{1,\ldots,L\}, \tag{5c}$$

where L represents the total number of subproblems into which (4) decomposes, $\hat{\mathbf{x}}^{(i)}$ represents the subset of states to be estimated in the ith subproblem and $f^{(i)}(\hat{\mathbf{x}}^{(i)})$ denotes the cost function associated with the ith subproblem. Furthermore, (5b) comprises the *complicating constraints*, i.e., constraints that feature variables pertaining to different subproblems, and whose existence prevents the overall problem from decomposing into L completely independent subproblems.

The method of Lagrange multipliers can be used to simplify the resolution of a problem. Therefore, (5) can be relaxed as follows:

$$\min_{\{\hat{\mathbf{x}}^{(i)}\}_{i=1}^{L}}\sum_{i=1}^{L}f^{(i)}\left(\hat{\mathbf{x}}^{(i)}\right)+\sum_{i=1}^{L}\boldsymbol{\lambda}^{(i)}\mathbf{h}^{(i)}\left(\hat{\mathbf{x}}^{(1)},\ldots,\hat{\mathbf{x}}^{(L)}\right) \tag{6a}$$

subject to

$$\mathbf{h}^{(i)}\left(\hat{\mathbf{x}}^{(1)},\ldots,\hat{\mathbf{x}}^{(L)}\right)\leq\mathbf{0}, \qquad i\in\{1,\ldots,L\}, \tag{6b}$$

$$\mathbf{g}^{(i)}\left(\hat{\mathbf{x}}^{(i)}\right)\leq\mathbf{0}, \qquad i\in\{1,\ldots,L\}, \tag{6c}$$

where $\boldsymbol{\lambda}^{(i)}$ is the vector of Lagrange multipliers associated to $\mathbf{h}^{(i)}$, with $i=1,\ldots,L$.

The relaxed overall problem (6) can be decomposed in L subproblems if the values of those variables pertaining to the rest of subproblems are fixed. Then, the ith subproblem can be posed in the following manner:

$$\begin{aligned}\min_{\hat{\mathbf{x}}^{(i)}}\ & f^{(i)}\left(\hat{\mathbf{x}}^{(i)}\right)+\sum_{j=1,j\neq i}^{L}f^{(j)}\left(\bar{\hat{\mathbf{x}}}^{(j)}\right) \\ &+\sum_{j=1,j\neq i}^{L}\bar{\boldsymbol{\lambda}}^{(j)}\mathbf{h}^{(j)}\left(\bar{\hat{\mathbf{x}}}^{(1)},\ldots,\bar{\hat{\mathbf{x}}}^{(i-1)},\hat{\mathbf{x}}^{(i)},\bar{\hat{\mathbf{x}}}^{(i+1)},\ldots,\bar{\hat{\mathbf{x}}}^{(L)}\right)\end{aligned} \tag{7a}$$

subject to

$$\mathbf{h}^{(i)}\left(\bar{\hat{\mathbf{x}}}^{(1)},\ldots,\bar{\hat{\mathbf{x}}}^{(i-1)},\hat{\mathbf{x}}^{(i)},\bar{\hat{\mathbf{x}}}^{(i+1)},\ldots,\bar{\hat{\mathbf{x}}}^{(L)}\right)\leq\mathbf{0}, \tag{7b}$$

$$\mathbf{g}^{(i)}\left(\hat{\mathbf{x}}^{(i)}\right)\leq\mathbf{0}, \tag{7c}$$

and the variables with an overline represent fixed values. Moreover, note that different distributions of complicating constraints among the subproblems yield the same solution.

The OCD achieves coordination among subproblems by relaxing those complicating constraints of the jth block that include variables from the ith subproblem in the ith subproblem. In this way, the effect of the solution of the ith subproblem on the jth subproblem is directly accounted for in the cost function of the ith subproblem. Thus, coordination of subproblems regarding complicating constraints is straightforward, and is accomplished by performing the multiplier updates as

$$\boldsymbol{\lambda}_i\leftarrow\boldsymbol{\lambda}_i+\alpha\mathbf{h}_i,\ i\in\{1,\ldots,L\}, \tag{8}$$

with α an appropriate constant and $\mathbf{h}_i$ is evaluated using the latest solution.

The methodology performs decomposition of the global problem by means of the manipulation of the KKT conditions, which can be expressed as [46]:

$$\nabla_{\hat{\mathbf{x}}^{(i)}}f_i\left(\hat{\mathbf{x}}_*^{(i)}\right)+\sum_{i=1}^{L}\nabla_{\hat{\mathbf{x}}^{(i)}}^{\mathsf{T}}\mathbf{h}^{(i)}\left(\hat{\mathbf{x}}_*^{(1)},\ldots,\hat{\mathbf{x}}_*^{(L)}\right)\boldsymbol{\lambda}_*^{(i)} \tag{9a}$$

$$+\sum_{i=1}^{L}\nabla_{\hat{\mathbf{x}}^{(i)}}^{\mathsf{T}}\mathbf{g}^{(i)}\left(\hat{\mathbf{x}}_*^{(i)}\right)\boldsymbol{\nu}_*^{(i)}=0, \qquad i\in\{1,\ldots,L\},$$

$$\mathbf{h}^{(i)}\left(\hat{\mathbf{x}}_*^{(1)},\ldots,\hat{\mathbf{x}}_*^{(L)}\right)\leq\mathbf{0}, \qquad i\in\{1,\ldots,L\}, \tag{9b}$$

$$\left(\mathbf{h}^{(i)}\left(\hat{\mathbf{x}}_*^{(1)},\ldots,\hat{\mathbf{x}}_*^{(L)}\right)\right)^{\mathsf{T}}\boldsymbol{\lambda}_*^{(i)}=0, \qquad i\in\{1,\ldots,L\}, \tag{9c}$$

$$\boldsymbol{\lambda}_*^{(i)}\geq\mathbf{0}, \qquad i\in\{1,\ldots,L\}, \tag{9d}$$

$$\mathbf{g}^{(i)}\left(\hat{\mathbf{x}}_*^{(i)}\right)\leq\mathbf{0}, \qquad i\in\{1,\ldots,L\}, \tag{9e}$$

$$\left(\mathbf{g}^{(i)}\left(\hat{\mathbf{x}}_*^{(i)}\right)\right)^{\mathsf{T}}\boldsymbol{\nu}_*^{(i)}=0, \qquad i\in\{1,\ldots,L\}, \tag{9f}$$

$$\boldsymbol{\nu}_*^{(i)}\geq\mathbf{0}, \qquad i\in\{1,\ldots,L\}, \tag{9g}$$

where $_*$ indicates optimal value.

All in all, OCD is a decomposition and coordination approach that resorts to the generation and manipulation of the KKT matrix associated to the overall problem. As a result, a suitable set of subproblems can be determined. However, the OCD does not yield the optimal decomposition; nor is it concerned with identifying the subproblems. Hence, a complementary method to the OCD that provides the optimal partitioning is discussed next.

3.2. Community-detection-based optimal partitioning

The KKT matrix contains information regarding the system structure and connections among variables. The same information can be expressed by means of the graph $\mathcal{G}=(\mathcal{N},\mathcal{E})$. In the case of an estimation problem, $\mathcal{G}$ captures the connections among outputs, system states and unmeasurable system states [47].

Partitioning the KKT matrix can thus be approached using graph theory techniques. Traditional methods have generally aimed at determining block-diagonal or block-triangular structures [23]. A different approach known as community detection has emerged more recently, and is concerned with identifying community structures within networks, i.e., clusters of densely connected nodes which are more sparsely connected to other clusters [24]. A metric called modularity is employed to assess this property for a certain network decomposition, and is defined as [48]

$$M=\frac{1}{2m}\sum_{i,j}\left(A_{ij}-\frac{k_ik_j}{2m}\right)\delta\left(c_i,c_j\right), \tag{10}$$

where M quantifies the resulting decomposition modularity, m denotes the sum of weights of the edges, A_{ij} is the weight of the edge connecting the ith and jth nodes, k_i and k_j denote the sum of weights of all edges that connect the ith and jth nodes with the remaining nodes, respectively, c_i and c_j indicate the communities that the ith and jth nodes belong to, respectively, and $\delta\left(c_i,c_j\right)$ is the Kronecker delta function:

$$\delta\left(c_i,c_j\right)=\begin{cases}1 & \text{if the } i\text{th and } j\text{th belong to identical communities,}\\ 0 & \text{otherwise.}\end{cases} \tag{11}$$

In the light of the above information, community detection seeks to determine the decomposition for which a maximal modularity is attained [49]. Although modularity maximization is an NP-hard problem [50], various approaches that provide near-optimal performances have been developed, such as fast unfolding [51] and spectral clustering [52]. Fast unfolding is selected in this work to determine communities into which the overall system decomposes. It consists of two phases that are reiterated until no further improvement is possible, hence generating the optimal partitioning. An initial partition consisting of a single node per community is considered. The first phase (modularity optimization) requires to compute the modularity gain obtained if the ith node is assigned to a neighboring community. This is

Algorithm 1 The fast unfolding algorithm

Require: graph $\mathcal{G}$

1: Initialization: assign each node in $\mathcal{N}$ to a different community

2: Evaluate modularity gain ΔM resulting from allocating the ith node in the jth community $\forall i, j \in \mathcal{N}$ and $j \in \mathcal{N}^{(i)}$, as

$$\Delta M = \left(\frac{\sum_{in} + k_{i,in}}{2m} - \left(\frac{\sum_{tot} + k_i}{2m} \right)^2 \right) - \left(\frac{\sum_{in}}{2m} - \left(\frac{\sum_{tot}}{2m} \right)^2 - \left(\frac{k_i}{2m} \right)^2 \right),$$

with $\sum_{in}$ the total weight of edges in the community of destination, $\sum_{tot}$ the total weight of edges connected with the community of destination, k_i the sum of weights of edges incident to the ith node, $k_{i,in}$ the total weight of edges linking the ith node to the jth community and m the total weight of all edges in the network

3: Place ith node in the community for which ΔM is maximum

4: Repeat 2–3 for all nodes until no additional improvement is possible

5: Build a new network with identified communities as nodes

6: Repeat 2–5 until communities remain invariable and a maximum M is achieved

Algorithm 2 OCD-based decomposition and coordination scheme

Require: overall MHE problem (2)

1: Formulate KKT matrix for (2) using (9)

2: Determine graph $\mathcal{G}$ equivalent to the KKT matrix

3: Execute Algorithm 1

4: Rearrange matrix of KKT conditions according to final communities

5: Formulate MHE local subproblems based on rearranged KKT matrix

done for each node in the graph. The second phase (community aggregation) involves adding each node to the neighboring community that yields the maximal modularity gain. This procedure is repeated iteratively on the resulting network, in which the communities formed in the prior phase are the new nodes, and the new weights of the edges can be obtained as the sums of all edges among communities. Algorithm 1 summarizes these steps and provides additional insight.

Finally, the partitioning approach is incorporated into the final OCD-based decomposition and coordination scheme sketched in Algorithm 2.

4. Distributed state estimation

The approach described in the previous section allows to decompose (2) into a set of minimally coupled subproblems and address coordination of the solutions. Building on these results, a DMHE-SM algorithm is introduced to deal with the subproblems.

The overall state estimation strategy can be thought of as a two-step approach. In a first stage, the feasible region (according to physical and operational constraints) of state estimates is tightened by using the consistency between the system outputs and the disturbance and noise bounds. Then, the DMHE solves a distributed estimation problem with tightened bounds that yields the optimal state estimates.

4.1. Bound tightening using the set-membership approach

Physical bounds on the states can be tightened considering the measured system outputs and known noise bounds. The set-membership approach constructs a compact set that bounds the system states that are compatible with the measurements and the noise bounds [5]. These sets can be represented using different geometric figures, e.g., ellipsoids, polytopes and zonotopes [53]. While general polytopes yield tighter enclosures than ellipsoids, efficient results can only be obtained in those cases with a reasonable amount of observations and order of the system [4]. Conversely, the use of ellipsoids might yield rough enclosures of the consistent state sets [54]. Zonotopes constitute an interesting trade-off, as they are characterized by superior compactness over ellipsoids, and offer reduced complexity with respect to polytopes [3].

Zonotopes are described by a center and a generator matrix (also referred to as matrix of segments), and can be used to represent sets $\mathcal{W}$ and $\mathcal{V}$ introduced in Section 2 as follows:

$$\mathcal{W} = \langle \mathbf{c}_w, \mathbf{E}_w \rangle, \tag{12a}$$

$$\mathcal{V} = \langle \mathbf{c}_v, \mathbf{E}_v \rangle, \tag{12b}$$

where $\mathbf{c}_w \in \mathbb{R}^{n_x}$ and $\mathbf{c}_v \in \mathbb{R}^{n_y}$ are the centers, and $\mathbf{E}_w \in \mathbb{R}^{n_x \times n_x}$ and $\mathbf{E}_v \in \mathbb{R}^{n_y \times n_y}$ are the generator matrices. Moreover, $\mathbf{E}_w$ and $\mathbf{E}_v$ are diagonal matrices whose entries correspond to disturbance and noise bounds, respectively.

The zonotopic set-membership approach employed in this work follows the design reported in [4]. The main steps performed at each time instant are summarized in Algorithm 3. It should be noted that, with some abuse of notation, the second step is carried out by successively updating the center and segments considering one measurement from the output vector at a time. Moreover, it can be observed that the computation of $\mathbf{R}_k^{(c)}$ results in matrices of increasing dimensions. To this end, the reduction operations given by [54, Eqs. (8)–(10)] must be performed.

Algorithm 3 needs to be adapted so that it can provide an appropriate input to the DMHE. Indeed, states are estimated in each subproblem for a certain input–output data window, hence bounds should be computed for the same whole window. To this end, some ideas presented in [17] to compute tightened bounds in a distributed manner are used. Then, Algorithm 4 provides the integration of the set-membership approach within the DMHE framework, sketching the steps followed by the lth agent at every time instant within the window, with $l = 1, \ldots, L$, and L is the total number of subproblems. The result is a set of tightened bounds for all subproblems for the corresponding window.

4.2. The DMHE-SM algorithm

The final solution combines the CMHE (2), Algorithms 1 and 2 for decomposition of (2) and determination of the coordination policy for the identified subproblems, and Algorithms 3 and 4 for bound tightening using the set-membership approach. Note that Algorithms 1 and 2 can be solved offline, and their solution should be available before the start of online computations.

The DMHE problem solved by the lth agent can be formulated based on (2) and (7) as follows:

$$\min_{\{\hat{\mathbf{x}}_{i|k}^{(l)}\}_{i=k-N+1}^{k}} \; J^{(l)}\left(\hat{\mathbf{x}}_{i|k}^{(l)}\right) + \sum_{m \in \mathcal{N}^{(l)}} J^{(m)}\left(\hat{\mathbf{x}}_{i|k}^{(m)}\right) + \tag{13a}$$

$$\boldsymbol{\lambda}^{(m)}\left[\left(\mathbf{w}_{i|k}^{(m)} - \left(\hat{\mathbf{x}}_{i+1|k}^{(m)} - \mathbf{A}^{(m)}\hat{\mathbf{x}}_{i|k}^{(m)} - \mathbf{B}^{(m)}\mathbf{u}_i^{(m)}\right)\right) +$$

Algorithm 3 The zonotopic set-membership approach [4]

Require: $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{E}_w$, $\mathbf{E}_v$, $\mathbf{c}_{k-1}^{(c)}$, $\mathbf{R}_{k-1}^{(c)}$, input–output data

1: *Step 1 (prediction).* Use the corrected zonotope $\left\langle \mathbf{c}_{k-1}^{(c)}, \mathbf{R}_{k-1}^{(c)} \right\rangle$ at the previous time instant to compute

$$\mathbf{c}_k^{(p)} = \mathbf{A}\mathbf{c}_{k-1}^{(c)} + \mathbf{B}\mathbf{u}_{k-1},$$

$$\mathbf{R}_k^{(p)} = [\mathbf{A}\mathbf{R}_{k-1}^{(c)} \quad \mathbf{E}_w],$$

where $\mathbf{c}_k^{(p)}$ and $\mathbf{R}_k^{(p)}$ are the predicted center and segments of the zonotope at current time instant k.

2: *Step 2 (update).* Update $\left\langle \mathbf{c}_k^{(p)}, \mathbf{R}_k^{(p)} \right\rangle$ using the ith output as follows:

3: **for** $i = 1 : n_y$ **do**

4: Compute the free vector $\boldsymbol{\lambda}_i \in \mathbb{R}^{n_x}$ as

$$\boldsymbol{\lambda}_i = \left(\mathbf{R}_{k-1}^{(c)} \left(\mathbf{R}_{k-1}^{(c)}\right)^{\intercal} \mathbf{C}_i^{\intercal}\right) \left(\mathbf{C}_i \mathbf{R}_{k-1}^{(c)} \left(\mathbf{R}_{k-1}^{(c)}\right)^{\intercal} \mathbf{C}_i^{\intercal} + \mathbf{E}_{v_i} \mathbf{E}_{v_i}^{\intercal}\right)^{-1}$$

such that $\boldsymbol{\lambda}_i$ minimizes the Frobenius norm of the zonotope.

5: Correct the zonotope as

$$\mathbf{c}_k^{(c)} = \mathbf{c}_k^{(p)} + \boldsymbol{\lambda}_i \left(y_i - \mathbf{C}_i \mathbf{c}_k^{(p)}\right),$$

$$\mathbf{R}_k^{(c)} = [(\mathbf{I} - \boldsymbol{\lambda}_i \mathbf{C}_i)\, \mathbf{R}_k^{(p)} \quad -\boldsymbol{\lambda}_i \mathbf{E}_{v_i}],$$

and construct tightened bounds at time instant k using $\left\langle \mathbf{c}_k^{(c)}, \mathbf{R}_k^{(c)} \right\rangle$.

6: **end for**

Algorithm 4 Set-membership bound tightening in the DMHE framework

Require: parameters in Algorithm 3

1: **for** $i = (k - N + 1) : k$ **do**

2: Send $\left\langle \mathbf{c}_{i-1}^{(c,l)}, \mathbf{R}_{i-1}^{(c,l)} \right\rangle$ to all neighbors

3: Receive $\left\langle \mathbf{c}_{i-1}^{(c,m)}, \mathbf{R}_{i-1}^{(c,m)} \right\rangle$, $\forall m \in \mathcal{N}^{(l)}$

4: Obtain input–output data pair $\left(\mathbf{u}_k^{(l)}, \mathbf{y}_i^{(l)}\right)$

5: Compute $\left\langle \mathbf{c}_i^{(c,l)}, \mathbf{R}_i^{(c,l)} \right\rangle$ using Algorithm 3 and [17, Eqs. (6) and (23)]

6: **end for**

7: Obtain tightened bounds $\mathcal{X}_{SM}^{(l)}$ for the considered time window

$$\left(\mathbf{v}_{i|k}^{(m)} - \left(\mathbf{y}_i^{(m)} - \mathbf{C}^{(m)} \hat{\mathbf{x}}_{i|k}^{(m)}\right)\right)\Bigg]$$

subject to

$$\mathbf{w}_{i|k}^{(l)} = \hat{\mathbf{x}}_{i+1|k}^{(l)} - \left(\mathbf{A}^{(l)} \hat{\mathbf{x}}_{i|k}^{(l)} + \mathbf{B}^{(l)} \mathbf{u}_i^{(l)}\right), \; i \in \{k - N + 1, \ldots, k - 1\}, \tag{13b}$$

$$\mathbf{v}_{i|k}^{(l)} = \mathbf{y}_i^{(l)} - \mathbf{C}^{(l)} \hat{\mathbf{x}}_{i|k}^{(l)}, \; i \in \{k - N + 1, \ldots, k - 1\}, \tag{13c}$$

$$\hat{\mathbf{x}}_{j|k}^{(l)} \in \mathcal{X}_{SM}^{(l)}, \; j \in \{k - N + 1, \ldots, k\}, \tag{13d}$$

where l and m represent information pertaining to the lth and mth subsystems, respectively. It is recalled that the values of the variables with the superscript m are determined in the mth subsystem, and are thus regarded as parameters in the lth subsystem. Note also that the lth and mth problems are coupled (or equivalently, neighbors) if one or more variables appear in the equations of both subproblems, which may be expressed as $m \in \mathcal{N}^{(l)}$. Moreover, $J^{(l)}$ and $J^{(m)}$ are as in (2a), but adapted to the lth and mth subsystems, respectively.

Algorithm 5 Online DMHE-SM algorithm

Require: Subproblems (13); $\left\{\left(\mathbf{u}_i^{(l)}, \mathbf{y}_i^{(l)}\right)\right\}$, $l = 1, \ldots, L$

1: **while** $k \leq t_{sim}$ **do**

2: Initialize Lagrange multipliers in all subproblems

3: Provide $\left\{\left(\mathbf{u}_i^{(l)}, \mathbf{y}_i^{(l)}\right)\right\}_{i=k-N+1}^{k}$ to the lth subproblem, $l = 1, \ldots, L$

4: Execute Algorithm 4 and obtain tightened bounds

5: Perform one iteration for each subproblem

6: **while** stop criterion not satisfied **do**

7: Exchange last solution among coupled subproblems

8: Update Lagrange multipliers using (8)

9: Perform a new iteration for each subproblem

10: **end while**

11: Extract $\hat{\mathbf{x}}_{k|k}^{(l)}$, $l = 1, \ldots, L$

12: $k \leftarrow k + 1$

13: **end while**

With all this, Algorithm 5 describes the main steps that must be followed to carry out the online DMHE-SM approach. As mentioned before, the solutions of Algorithms 1 and 2 need to be available before online computations can start.

4.3. Convergence analysis of the DMHE-SM

Algorithm 5 can be employed for the purpose of state estimation for any system that can be described by (1), provided that input–output data is available. Then, convergence of the proposed approach can be examined in two steps:

1. The first part is concerned with the set-membership approach. The methodology devised in [4] is followed, which determines a guaranteed bound of the uncertain system trajectory at every sampling instant [4, Property 3].
2. The second part examines the convergence of the DMHE, which can be analyzed with independence of the set-membership results. Indeed, the set-membership approach only tightens the original bounds, hence the MHE formulation remains invariable. Then, its convergence can be discussed at two different levels:
   - 2.1. Convergence of the decomposition algorithm (OCD) is guaranteed by [45, Eqs. (5.97)–(5.100)].
   - 2.2. The choice and update of certain weighting matrices guarantees convergence of the DMHE [19, Theorem 1]. In particular, the initial penalty matrix, which is denoted with $\mathbf{P}^{-1}$ in this paper, can be updated as described in [19, Eqs. (26)–(28)]. Another possibility consists in using the tightened bounds (computed in the set-membership step detailed in Algorithm 4) to update the initial penalty matrix as in [54, Definition 4], where a procedure to generate this covariance matrix from the zonotope generator matrix is provided.

5. Case study

The efficacy of the proposed distributed state estimation methodology is assessed by means of a typical reactor–separator process. This benchmark case study has been widely employed in the literature to illustrate many control and state estimation approaches [12,38,55–63].

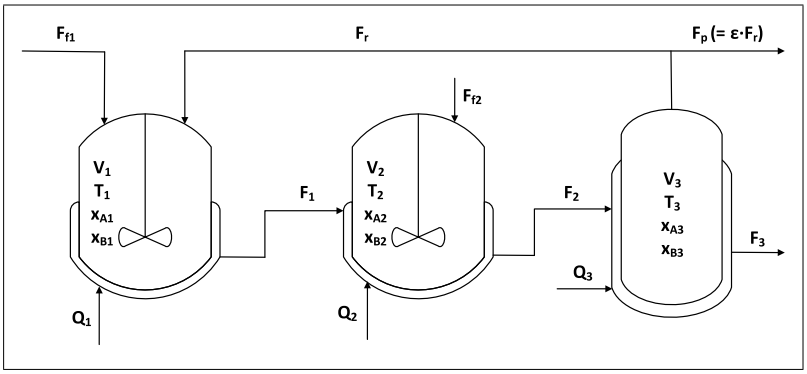

Fig. 1. Schematic representation of the reactor–separator process.

5.1. System description

The reactor–separator system, depicted in Fig. 1, features two continuously stirred tank reactors and a vapor–liquid separator [12,57]. Two streams of pure reactant A, denoted by F_{f_1} and F_{f_2}, are fed to both reactors and transformed into desired product B according to the first-order reaction $A \xrightarrow{r_1} B$. A second parallel first-order reaction $B \xrightarrow{r_2} C$ causes B to be lost to side product C, where r_1 and r_2 represent the reaction rates [56]. The stream leaving the first reactor, i.e., F_1, is fed to the second reactor, and contains all components A, B and C. In turn, the outlet stream of the second reactor, i.e., F_2, is supplied to the separator, which works at equilibrium conditions to isolate B from the mixture [62]. A recycle stream F_r is supplied by the separator to the first reactor, while simultaneously purging a small ratio, denoted by ϵ, to prevent accumulation of C [58]. Furthermore, the heat flows Q_1, Q_2 and Q_3 are supplied by the jackets to the two reactors and the separator, respectively.

The subsequent model characterizes the holdup, temperature and concentration dynamics of species A and B:

$$\frac{dV_1}{dt} = F_{f_1} + F_r - F_1, \tag{14a}$$

$$\frac{dV_2}{dt} = F_{f_2} + F_1 - F_2, \tag{14b}$$

$$\frac{dV_3}{dt} = F_2 - (1+\epsilon)\,F_r - F_3, \tag{14c}$$

$$\frac{dT_1}{dt} = \frac{F_{f_1}}{V_1}(T_0 - T_1) + \frac{F_r}{V_1}(T_3 - T_1) + \frac{Q_1}{\rho C_p V_1} - \frac{\mu}{C_p}\left(\Delta H_{r_1} k_1^0 \exp\left(\frac{-E_1}{RT_1}\right)x_{A1} + \Delta H_{r_2} k_2^0 \exp\left(\frac{-E_2}{RT_1}\right)x_{B1}\right), \tag{14d}$$

$$\frac{dT_2}{dt} = \frac{F_{f_2}}{V_2}(T_0 - T_2) + \frac{F_1}{V_2}(T_1 - T_2) + \frac{Q_2}{\rho C_p V_2} - \frac{\mu}{C_p}\left(\Delta H_{r_1} k_1^0 \exp\left(\frac{-E_1}{RT_2}\right)x_{A2} + \Delta H_{r_2} k_2^0 \exp\left(\frac{-E_2}{RT_2}\right)x_{B2}\right), \tag{14e}$$

$$\frac{dT_3}{dt} = \frac{F_2}{V_3}(T_2 - T_3) + \frac{Q_3}{\rho C_p V_3}, \tag{14f}$$

$$\frac{dx_{A1}}{dt} = \frac{F_r}{V_1}\left(\frac{\alpha_A x_{A3}}{\alpha_A x_{A3} + \alpha_B x_{B3} + \alpha_C\left(1 - x_{A_3} - x_{B_3}\right)} - x_{A1}\right) + \frac{F_{f_1}}{V_1}(x_{A0} - x_{A1}) - k_1^0 \exp\left(\frac{-E_1}{RT_1}\right)x_{A1}, \tag{14g}$$

$$\frac{dx_{B1}}{dt} = \frac{F_r}{V_1}\left(\frac{\alpha_B x_{B3}}{\alpha_A x_{A3} + \alpha_B x_{B3} + \alpha_C\left(1 - x_{A_3} - x_{B_3}\right)} - x_{B1}\right) - \frac{F_{f_1}}{V_1}x_{B1} + k_1^0 \exp\left(\frac{-E_1}{RT_1}\right)x_{A1} - k_2^0 \exp\left(\frac{-E_2}{RT_1}\right)x_{B1}, \tag{14h}$$

$$\frac{dx_{A2}}{dt} = \frac{F_{f_2}}{V_2}(x_{A0} - x_{A2}) + \frac{F_1}{V_2}(x_{A1} - x_{A2}) - k_1^0 \exp\left(\frac{-E_1}{RT_2}\right)x_{A2}, \tag{14i}$$

$$\frac{dx_{B2}}{dt} = \frac{F_1}{V_2}(x_{B1} - x_{B2}) - \frac{F_{f_2}}{V_2}x_{B2} + k_1^0 \exp\left(\frac{-E_1}{RT_2}\right)x_{A2} - k_2^0 \exp\left(\frac{-E_2}{RT_2}\right)x_{B2}, \tag{14j}$$

$$\frac{dx_{A3}}{dt} = \frac{F_2}{V_3}(x_{A2} - x_{A3}) - \frac{(1+\epsilon)F_r}{V_3}\left(\frac{\alpha_A x_{A3}}{\alpha_A x_{A3} + \alpha_B x_{B3} + \alpha_C\left(1 - x_{A_3} - x_{B_3}\right)} - x_{A3}\right), \tag{14k}$$

$$\frac{dx_{B3}}{dt} = \frac{F_2}{V_3}(x_{B2} - x_{B3}) - \frac{(1+\epsilon)F_r}{V_3}\left(\frac{\alpha_B x_{B3}}{\alpha_A x_{A3} + \alpha_B x_{B3} + \alpha_C\left(1 - x_{A_3} - x_{B_3}\right)} - x_{B3}\right), \tag{14l}$$

where V, T, x_A and x_B are the system variables and denote volumetric holdup, temperature and mole fractions of A and B, respectively. Moreover, the subscripts 1, 2 and 3 denote magnitudes associated with the outlet stream of the first and second reactors and the separator, respectively. Furthermore, F_{f_1}, F_{f_2}, F_1, F_2, F_3, F_r, Q_1, Q_2 and Q_3 are the manipulated inputs. The remaining terms in (14) are physical parameters described in Table 1.

5.2. Experimental design

Testing the effectiveness of the set-membership based distributed state estimation approach requires input–output process data to be available. To this end, the output tracking problem designed in [62] is implemented. This regulation problem considers three tracking zones, which correspond to intermediate, low and high overall conversion of A into B. The reference values of the states and the corresponding steady state inputs are given in Tables 2 and 3, respectively. Moreover, the initial state values are provided in Table 4.

It should be noted that the output tracking problem is solved in the presence of measurement and process noise. Hence, white noise with a signal-to-noise ratio of 40 dB is added to each sample, following the approach presented in [62]. Note that this information allows to define numerical values for (12).

Once the input–output data has been generated for the whole duration of the simulation, appropriate truncated data sequences are supplied to the DMHE-SM at every sampling instant. However, only a subset of the states are assumed to be measurable, while the rest must be estimated. The choice of measured states in this work is aligned with that of [64], thus considering that only holdup volumes and temperatures can be measured online.

Table 1
Description of the process parameters.

Parameter	Description	Value	Unit
ρ	Flow stream density	1000	kg/m^3
C_p	Flow stream heat capacity	4.2	kJ/kgK
x_{A_0}	Mole fraction of A in pure reactant stream	1	–
T_0	Temperature of pure reactant streams	359.1	K
$\{k_1^0, k_2^0\}$	Pre-exponential factors for reactions 1, 2	$\{2.77, 2.5\} \times 10^3$	1/s
$\{E_1, E_2\}$	Activation energies of reactions 1, 2	$\{5, 6\} \times 10^4$	kJ/kmol
$\{\Delta H_1, \Delta H_2\}$	Enthalpies of reactions 1, 2	$-\{6, 7\} \times 10^4$	kJ/kmol
$\{\alpha_A, \alpha_B, \alpha_C\}$	Relative volatilities of A, B, C	$\{5, 1, 0.5\}$	–
R	Universal gas constant	8.314	kJ/kmolK
ϵ	Purge ratio	0.02	–
μ	Flow stream molality	0.00279	kmol/kg

Table 2
Reference values of each state.

State	Tracking zone I	Tracking zone II	Tracking zone III
V_1^{ref} [m^3]	1	1.6	1.2
V_2^{ref} [m^3]	0.5	0.8	0.6
V_3^{ref} [m^3]	1	1.4	1.1
T_1^{ref} [K]	432.4	410.2	447.1
T_2^{ref} [K]	427.1	407.5	442.3
T_3^{ref} [K]	432.1	411.0	447.4
$x_{A_1}^{ref}$	0.536	0.733	0.265
$x_{B_1}^{ref}$	0.448	0.264	0.657
$x_{A_2}^{ref}$	0.545	0.724	0.287
$x_{B_2}^{ref}$	0.438	0.272	0.636
$x_{A_3}^{ref}$	0.298	0.507	0.103
$x_{B_3}^{ref}$	0.670	0.485	0.765

Table 3
Steady-state input values.

Input	Tracking zone I	Tracking zone II	Tracking zone III
$F_{f_1}^{ss}$ [m^3/h]	5.04	8.06	4.03
$F_{f_2}^{ss}$ [m^3/h]	5.04	7.05	3.53
F_1^{ss} [m^3/h]	22.04	35.26	17.63
F_2^{ss} [m^3/h]	27.08	42.31	21.16
F_3^{ss} [m^3/h]	9.74	14.57	7.29
F_r^{ss} [m^3/h]	17	27.2	13.6
Q_1^{ss} [kJ/h]	715.3 $\times 10^3$	786.8 $\times 10^3$	572.2 $\times 10^3$
Q_2^{ss} [kJ/h]	579.8 $\times 10^3$	637.8 $\times 10^3$	463.8 $\times 10^3$
Q_3^{ss} [kJ/h]	568.7 $\times 10^3$	625.6 $\times 10^3$	455.0 $\times 10^3$

Table 4
Initial state values.

State	Value	State	Value
V_1^0 [m^3]	0.7	T_1^0 [K]	400
V_2^0 [m^3]	0.7	T_2^0 [K]	400
V_3^0 [m^3]	1.5	T_3^0 [K]	400
$x_{A_1}^0$	0.65	$x_{B_1}^0$	0.3
$x_{A_2}^0$	0.65	$x_{B_2}^0$	0.3
$x_{A_3}^0$	0.65	$x_{B_3}^0$	0.3

The estimation problem is solved considering the same sampling time and prediction horizon reported in [62] for the output tracking problem, i.e., $T_s = 180$ s and $N = 15$ samples. Simulations are then performed in a computer with an Intel Core i7-8665U processor running at 1.9 GHz with 8 GB RAM. Furthermore, simulation results are obtained in Matlab R2020b[3] using IBM ILOG CPLEX Optimization Studio V12.10.0[4] and the YALMIP toolbox [65].

5.3. Results and discussion

Before proceeding with the analysis of results, it should be mentioned that the issue of input–output data generation via MPC is not addressed in this paper. Indeed, the proposed approach assumes that suitable data sequences are provided at regular time intervals. The interested reader is advised to check [62] for a comprehensive control problem description and analysis of results.

As a first step, the CMHE problem for the reactor–separator system is formulated as in (2). Choice of physical bounds and weighting matrices is as follows [47]:

- Physical bounds on state estimates are chosen to be the same as those considered in the MPC. Therefore, a $\pm 80\%$ bound on the states (with respect to reference values) is selected.
- Diagonal weighting matrices are defined so as to compensate the different magnitudes of the states, thus assigning equal priorities. Therefore, inverse values of the corresponding reference values are selected.

The KKT conditions associated to the resulting CMHE can be formulated using (9). Since the objective function is completely separable, its associated graph can be constructed following the procedure reported in [60]. Then, the fast unfolding algorithm[5] allows to identify three communities, one per vessel, thus matching the results reported in [64]. More precisely, the ith community consists of states V_i, T_i, x_{A_i} and x_{B_i}, with $i = 1, 2, 3$. Hence, the following couplings among subsystems can be identified:

$$\psi_{1,2} = \{T_1, x_{A_1}, x_{B_1}\},$$
$$\psi_{2,3} = \{T_2, x_{A_2}, x_{B_2}\},$$
$$\psi_{3,1} = \{T_3, x_{A_3}, x_{B_3}\},$$

where $\psi_{i,j}$ represents the states that are associated with (and hence optimized in) the ith subproblem, and at the same time

[3] https://nl.mathworks.com/products/new_products/release2020b.html
[4] https://www.ibm.com/support/pages/downloading-ibm-ilog-cplex-optimization-studio-v12100
[5] https://perso.uclouvain.be/vincent.blondel/research/louvain.html

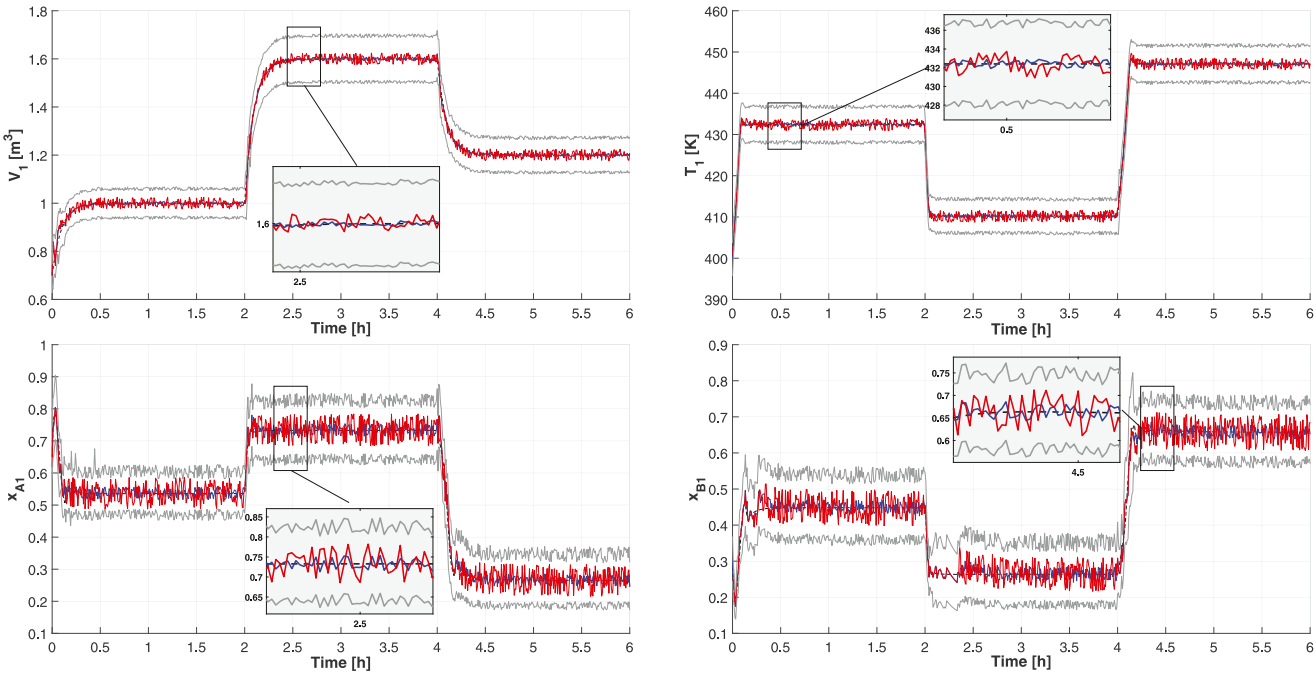

Fig. 2. State estimation results for the first subsystem: CMHE (blue solid line), DMHE (red solid line), SM bounds (gray solid line) and references (black dashed line).

influence the jth subproblem, with $i, j = 1, 2, 3$. Then, three local problems are derived from the overall problem using (13), and the DMHE-SM approach is executed as in Algorithm 5.

The solutions obtained using three different MHE schemes are presented and compared next, namely the centralized (CMHE) and two distributed approaches, one with (DMHE-SM) and another without set-membership bound tightening (thus removing line 4 in Algorithm 5). Then, the state estimates computed using each approach for the first, second and third subsystems are depicted in Figs. 2–4, respectively. It is worth noting that there is no significant difference in terms of performance between the DMHE-SM and the DMHE with physical bounds, which in turn does not seem to deviate much from the CMHE results. Moreover, it can be observed that the set-membership approach allows for bound tightening with respect to physical bounds (which are not even reproduced to offer better visualization of the results).

In the light of the results, the DMHE schemes appear to perform only slightly worse than the CMHE. To ascertain whether this is the case, centralized and distributed performances can be further compared using the cumulative cost over the entire simulation duration. This information is presented in Fig. 5, and allows to conclude that both DMHE lead to a minor decrease of performance, which amounts to less than 3% with respect to that of the CMHE. This issue is directly linked to the stop criterion mentioned in Algorithm 5, which is formulated following the ideas in [39,45] as

$$\sqrt{\sum_{l=1}^{3}\left(\max\left(\left|\mathbf{h}_{*}^{(l)}\right|\right)\right)^{2}} \leq 10^{-2}, \tag{15}$$

where $\mathbf{h}_{*}^{(l)}$ denotes the values of the complicating constraints once the solutions are substituted, and the threshold is selected bearing in mind the trade-off between accuracy of the solution and computational burden.

Even though centralized implementations result in optimal performances, distributed strategies can lead to lower computation times (among other advantages), thus offering an interesting trade-off. This issue is examined by determining the total computation times of the three approaches, which are depicted in Fig. 6. It can be noted that the distributed architectures require lower computation times with respect to the centralized scheme. Although the subproblems solved by the former approaches are smaller than the overall problem, several iterations are often needed to ensure convergence as defined in (15). Regarding the comparison between distributed schemes, bound tightening carried out in the DMHE-SM approach results in less iterations than the DMHE with physical bounds. Furthermore, additional overhead derived from computation of tightened bounds is compensated by the reduction of number of iterations.

A summary of computation times and required number of iterations is provided in Table 5. It can then be resolved that the DMHE-SM ultimately constitutes the most suitable approach, as it is almost 25% faster than the CMHE with a mere relative loss of performance below 3%.

6. Conclusions

This work has presented a two-step state estimation approach for large-scale systems, combining set-membership and moving horizon estimation. The former is used at the initial stage, tightening physical bounds on states using the consistency between the model, the measured outputs and the disturbance and noise bounds. An MHE is then employed to determine the optimal state estimates, benefiting from a tightened feasible region. Although not explicitly considered in this paper, the methodology could be further enhanced to tackle the case of systems characterized by parametric uncertainty, using results available in the literature [66,67].

The state estimation strategy is implemented and solved in a distributed manner. To this end, the set of subproblems is coordinated using the OCD. This is a particular implementation of Lagrangian relaxation, and consists in manipulating the KKT optimality conditions to yield subproblems whose solution converges to the optimal centralized solution. However, system partitioning needs to be determined separately, as the OCD addresses the coordination but not the decomposition. Hence, system partitioning is determined using community detection techniques, as close-to-optimal modularity decompositions are obtained. While this step

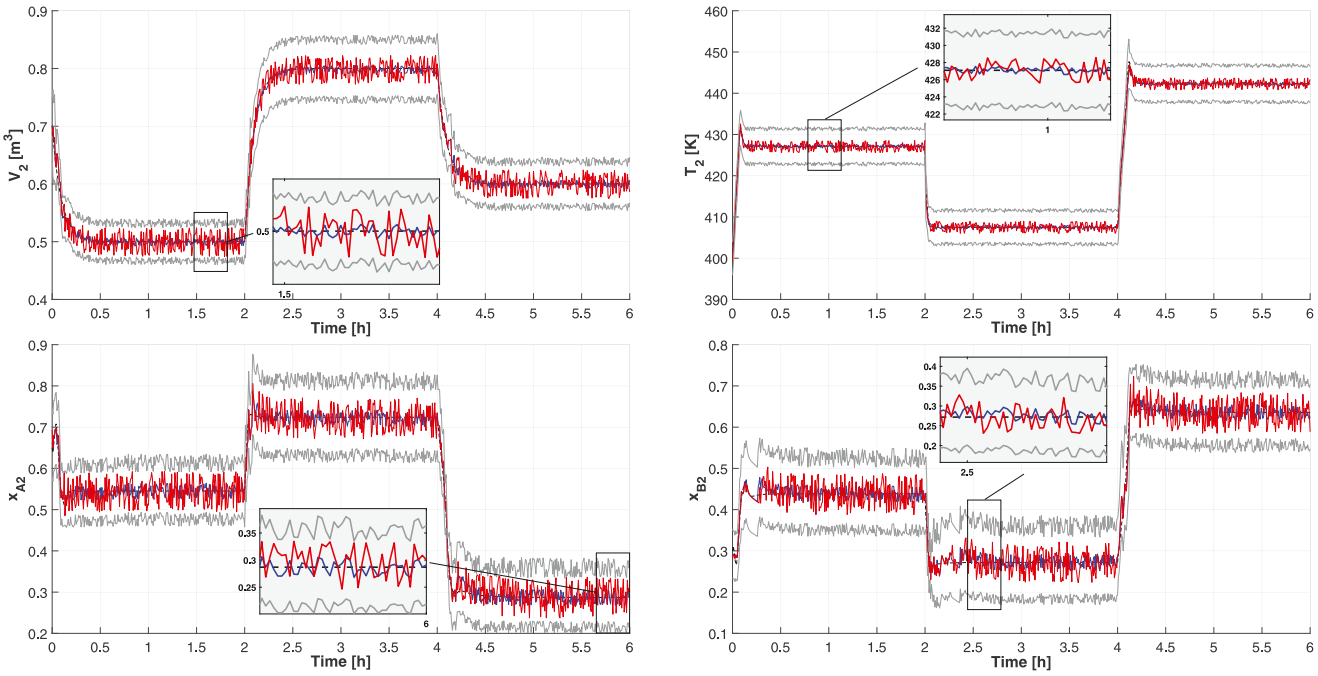

Fig. 3. State estimation results for the second subsystem: CMHE (blue solid line), DMHE (red solid line), SM bounds (gray solid line) and references (black dashed line).

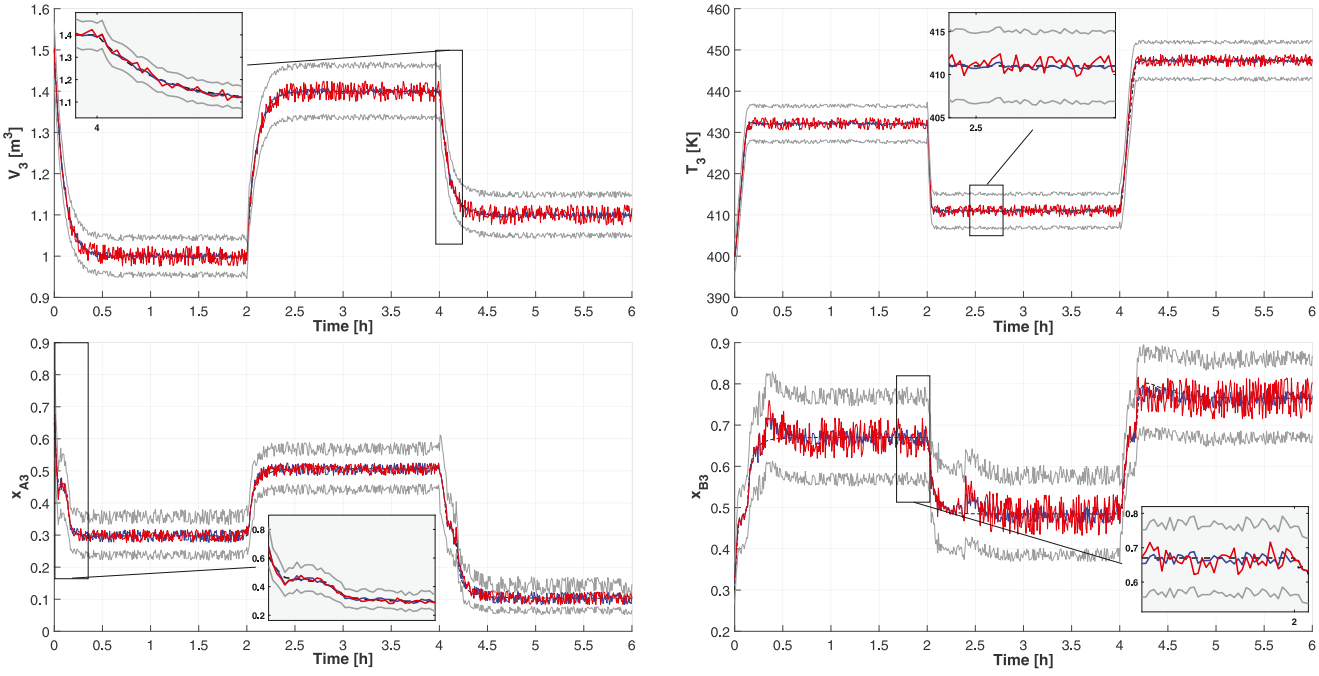

Fig. 4. State estimation results for the third subsystem: CMHE (blue solid line), DMHE (red solid line), SM bounds (gray solid line) and references (black dashed line).

Table 5
Summary of average computation times.

Approach	Nr. iterations	Bound tightening [s]	Total comp. time per sample [s]	Time reduction w.r.t. CMHE [%]
CMHE	1	–	0.1794	–
DMHE	≤ 4	–	0.1533	14.55
DMHE-SM	≤ 2	0.0586	0.1352	24.61

is carried out in an offline manner, future research could regard the issue of online re-partitioning. Then, the initial partitioning, which would still be computed offline, could evolve based on time-varying coupling conditions. The complete approach is then tested considering a typical reactor–separator system employed in the process industry, comparing its performance to those of a

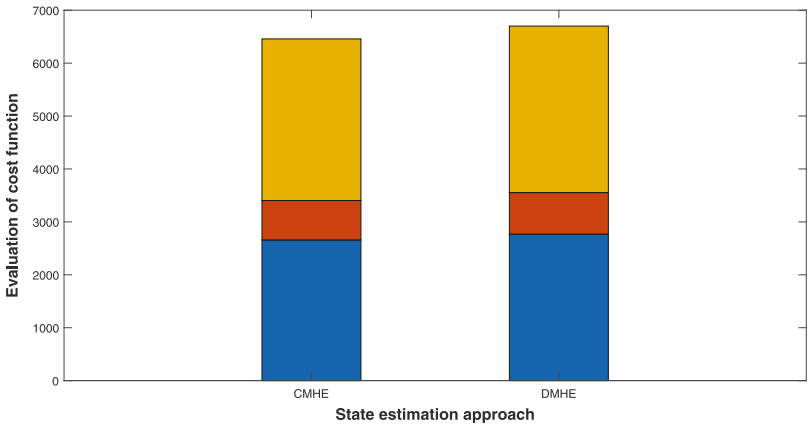

Fig. 5. Cost function evaluation for CMHE and DMHE (blue: zone I; red: zone II; yellow: zone III).

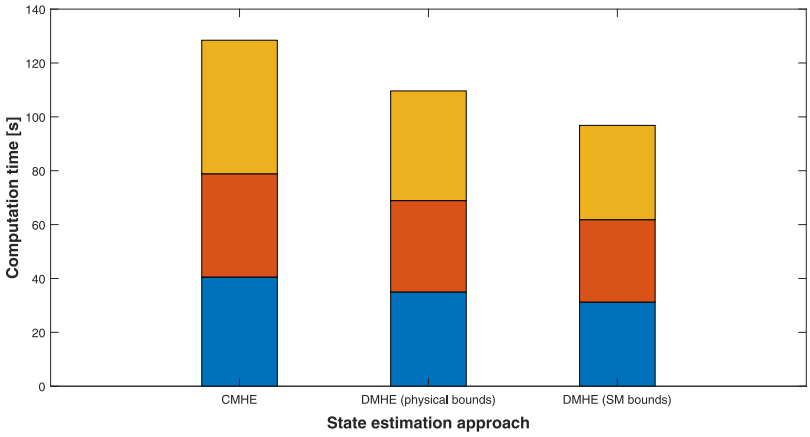

Fig. 6. Evaluation of total computation times for CMHE and DMHE (blue: zone I; red: zone II; yellow: zone III).

CMHE and a DMHE without set-membership. The results allow to validate the approach and highlight its performance, as it improves computational time while keeping loss of performance at a minimum with respect to the CMHE.

The distributed state estimation approach derived in this work complements and extends the results reported in [39], which mainly dealt with distributed control of large-scale systems without explicit consideration of the effect of disturbance and noise. The combination of the two algorithms would allow for robust distributed control and state estimation of large-scale systems, which is stated to be the main motivation behind the developments in [39]. Indeed, the aim is that of developing a methodology that enables to manage inland waterways in an optimal manner, and which is robust to natural and operational disturbances, e.g., unknown flows and lock operations, thus extending the centralized results provided in [68]. However, inland waterways models belong to the class of time-delay systems. Therefore, the application of the methodology to inland waterways would require an in-depth theoretical analysis, as it has been developed for systems that are not characterized by system delays. Another extension could regard the integration of the scheme within the hierarchical architecture presented in [69], thus allowing to tackle other features, e.g., the effect of tides and the existence of controlled infrastructure that can only apply actions from a set of discrete values.

Declaration of competing interest

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

References

[1] Kalman RE. A new approach to linear filtering and prediction problems. Trans ASME J Basic Eng 1960;82(1):35–45.
[2] Kalman RE, Bucy RS. New results in linear filtering and prediction theory. Trans ASME J Basic Eng 1961;83(1):95–108.
[3] Le VTH, Stoica C, Alamo T, Camacho EF, Dumur D. Zonotopic guaranteed state estimation for uncertain systems. Automatica 2013;49(11):3418–24.
[4] Alamo T, Bravo JM, Camacho EF. Guaranteed state estimation by zonotopes. Automatica 2005;41(6):1035–43.
[5] Alamo T, Bravo JM, Redondo MJ, Camacho EF. A set-membership state estimation algorithm based on DC programming. Automatica 2008;44(1):216–24.
[6] Rao CV. Moving horizon strategies for the constrained monitoring and control of nonlinear discrete-time systems. [Ph.D. thesis], University of Wisconsin–Madison; 2000.
[7] Rao CV, Rawlings JB, Mayne DQ. Constrained state estimation for nonlinear discrete-time systems: Stability and moving horizon approximations. IEEE Trans Automat Control 2003;48(2):246–58.
[8] Allgöwer F, Badgwell TA, Qin JS, Rawlings JB, Wright SJ. Nonlinear predictive control and moving horizon estimation — An introductory overview. In: Frank PM, editor. Advances in Control. London: Springer London; 1999, p. 391–449.
[9] Copp DA, Hespanha JP. Simultaneous nonlinear model predictive control and state estimation. Automatica 2017;77:143–54.
[10] Scattolini R. Architectures for distributed and hierarchical model predictive control - a review. J Process Control 2009;19(5):723–31.
[11] Ocampo-Martinez C, Bovo S, Puig V. Partitioning approach oriented to the decentralised predictive control of large-scale systems. J Process Control 2011;21(5):775–86.
[12] Christofides PD, Liu J, Muñoz de la Peña D. Networked and Distributed Predictive Control. London: Springer; 2011.
[13] Siljak DD. Decentralized Control of Complex Systems. Courier Corporation; 2011.

[14] Negenborn R, van Overloop PJ, Keviczky T, De Schutter B. Distributed model predictive control of irrigation canals. Netw Heterog Media 2009;4(2):359–80.
[15] Wang Y, Alamo T, Puig V, Cembrano G. Distributed zonotopic set-membership state estimation based on optimization methods with partial projection. IFAC-PapersOnLine 2017;50(1):4039–44, 20th IFAC World Congress.
[16] Orihuela L, Roshany-Yamchi S, García RA, Millán P. Distributed set-membership observers for interconnected multi-rate systems. Automatica 2017;85:221–6.
[17] Wang Y, Alamo T, Puig V, Cembrano G. A distributed set-membership approach based on zonotopes for interconnected systems. In: 2018 IEEE conference on decision and control. 2018. p. 668–73.
[18] Ierardi C, Orihuela L, Jurado I. A distributed set-membership estimator for linear systems with reduced computational requirements. Automatica 2021;132:109802.
[19] Farina M, Ferrari-Trecate G, Scattolini R. Distributed moving horizon estimation for linear constrained systems. IEEE Trans Automat Control 2010;55(11):2462–75.
[20] Haber A, Verhaegen M. Moving horizon estimation for large-scale interconnected systems. IEEE Trans Automat Control 2013;58(11):2834–47.
[21] Battistelli G. Distributed moving-horizon estimation with arrival-cost consensus. IEEE Trans Automat Control 2019;64(8):3316–23.
[22] Kim J, Kang JH, Bae J, Lee W, Kim KKK. Distributed moving horizon estimation via operator splitting for automated robust power system state estimation. IEEE Access 2021;9:90428–40.
[23] Allman A, Tang W, Daoutidis P. Towards a generic algorithm for identifying high-quality decompositions of optimization problems. In: 13th International Symposium on Process Systems Engineering. Computer Aided Chemical Engineering, vol. 44, Elsevier; 2018, p. 943–8.
[24] Girvan M, Newman MEJ. Community structure in social and biological networks. Proc Natl Acad Sci 2002;99(12):7821–6.
[25] Molzahn DK, Dörfler F, Sandberg H, Low SH, Chakrabarti S, Baldick R, Lavaei J. A survey of distributed optimization and control algorithms for electric power systems. IEEE Trans Smart Grid 2017;8(6):2941–62.
[26] Boyd S, Parikh N, Chu E, Peleato B, Eckstein J. Distributed optimization and statistical learning via the alternating direction method of multipliers. Found Trends Mach Learn 2011;3(1):1–122.
[27] Conejo AJ, Nogales FJ, Prieto FJ. A decomposition procedure based on approximate Newton directions. Math Program 2002;93(3):495–515.
[28] Wang Y, Wang S, Wu L. Distributed optimization approaches for emerging power systems operation: a review. Electr Power Syst Res 2017;144:127–35.
[29] Kekatos V, Giannakis GB. Distributed Robust power system state estimation. IEEE Trans Power Syst 2013;28(2):1617–26.
[30] Chen T, Foo YSE, Ling K, Chen X. Distributed state estimation using a modified partitioned moving horizon strategy for power systems. Sensors 2017;17(10).
[31] Zheng W, Wu W, Gomez-Exposito A, Zhang B, Guo Y. Distributed Robust bilinear state estimation for power systems with nonlinear measurements. IEEE Trans Power Syst 2017;32(1):499–509.
[32] Zavala VM, Laird CD, Biegler LT. A fast moving horizon estimation algorithm based on nonlinear programming sensitivity. J Process Control 2008;18(9):876–84.
[33] Voelker A, Kouramas K, Pistikopoulos EN. Moving horizon estimation: Error dynamics and bounding error sets for robust control. Automatica 2013;49(4):943–8.
[34] Ellis M, Zhang J, Liu J, Christofides PD. Robust moving horizon estimation based output feedback economic model predictive control. Systems Control Lett 2014;68:101–9.
[35] Sartipizadeh H, Vincent TL. Computationally tractable robust moving horizon estimation using an approximate convex hull. In: 2016 IEEE 55th conference on decision and control. 2016. p. 3757–62.
[36] Chen T. Robust state estimation for power systems via moving horizon strategy. Sustain Energy Grids Netw 2017;10:46–54.
[37] Farina M, Ferrari-Trecate G, Scattolini R. Distributed moving horizon estimation for nonlinear constrained systems. Internat J Robust Nonlinear Control 2012;22(2):123–43.
[38] Zhang J, Liu J. Distributed moving horizon state estimation for nonlinear systems with bounded uncertainties. J Process Control 2013;23(9):1281–95.
[39] Segovia P, Puig V, Duviella E, Etienne L. Distributed model predictive control using optimality condition decomposition and community detection. J Process Control 2021;99:54–68.
[40] Chisci L, Zappa G. Feasibility in predictive control of constrained linear systems: the output feedback case. Internat J Robust Nonlinear Control 2002;12(5):465–87.
[41] Mayne D, Raković S, Findeisen R, Allgöwer F. Robust output feedback model predictive control of constrained linear systems. Automatica 2006;42(7):1217–22.
[42] Goulart PJ, Kerrigan EC. Output feedback receding horizon control of constrained systems. Internat J Control 2007;80(1):8–20.
[43] Brunner FD, Müller MA, Allgöwer F. Enhancing output-feedback MPC with set-valued moving horizon estimation. IEEE Trans Automat Control 2018;63(9):2976–86.
[44] Sui D, Feng L, Hovd M. Robust output feedback model predictive control for linear systems via moving horizon estimation. In: 2008 American control conference. 2008. p. 453–58.
[45] Conejo AJ, Castillo E, Mínguez R, García-Bertrand R. Decomposition techniques in mathematical programming: engineering and science applications. Springer Science & Business Media; 2006.
[46] Boyd S, Vandenberghe L. Convex Optimization. Cambridge University Press; 2004.
[47] Pourkargar DB, Moharir M, Almansoori A, Daoutidis P. Distributed estimation and nonlinear model predictive control using community detection. Ind Eng Chem Res 2019;58(30):13495–507.
[48] Leicht EA, Newman MEJ. Community structure in directed networks. Phys Rev Lett 2008;100(11):118703.
[49] Daoutidis P, Tang W, Allman A. Decomposition of control and optimization problems by network structure: Concepts, methods, and inspirations from biology. AIChE J 2019;65(10):e16708.
[50] Allman A, Tang W, Daoutidis P. DeCODe: a community-based algorithm for generating high-quality decompositions of optimization problems. Opt Eng 2019;20(4):1067–84.
[51] Blondel VD, Guillaume JL, Lambiotte R, Lefebvre E. Fast unfolding of communities in large networks. J Stat Mech Theory Exp 2008;2008(10):P10008.
[52] Newman MEJ. Modularity and community structure in networks. Proc Natl Acad Sci 2006;103(23):8577–82.
[53] Pourasghar M, Puig V, Ocampo-Martinez C. Interval observer versus set-membership approaches for fault detection in uncertain systems using zonotopes. Internat J Robust Nonlinear Control 2019;29(10):2819–43.
[54] Combastel C. Zonotopes and Kalman observers: Gain optimality under distinct uncertainty paradigms and robust convergence. Automatica 2015;55:265–73.
[55] Liu J, Muñoz de la Peña D, Ohran BJ, Christofides PD, Davis JF. A two-tier architecture for networked process control. Chem Eng Sci 2008;63(22):5394–409.
[56] Stewart BT, Venkat AN, Rawlings JB, Wright SJ, Pannocchia G. Cooperative distributed model predictive control. Systems Control Lett 2010;59(8):460–9.
[57] Tippett MJ, Bao J. Distributed model predictive control based on dissipativity. AIChE J 2013;59(3):787–804.
[58] Pourkargar DB, Almansoori A, Daoutidis P. Impact of decomposition on distributed model predictive control: A process network case study. Ind Eng Chem Res 2017;56(34):9606–16.
[59] Yin X, Liu J. Distributed moving horizon state estimation of two-time-scale nonlinear systems. Automatica 2017;79:152–61.
[60] Tang W, Allman A, Pourkargar DB, Daoutidis P. Optimal decomposition for distributed optimization in nonlinear model predictive control through community detection. Comput Chem Eng 2018;111:43–54.
[61] Tang W, Daoutidis P. Network decomposition for distributed control through community detection in input–output bipartite graphs. J Process Control 2018;64:7–14.
[62] Pourkargar DB, Almansoori A, Daoutidis P. Comprehensive study of decomposition effects on distributed output tracking of an integrated process over a wide operating range. Chem Eng Res Des 2018;134:553–63.
[63] Zeng J, Liu J. Distributed state estimation based distributed model predictive control. Mathematics 2021;9(12).
[64] Yin X, Liu J. Subsystem decomposition of process networks for simultaneous distributed state estimation and control. AIChE J 2019;65(3):904–14.
[65] Löfberg J. YALMIP: a toolbox for modeling and optimization in MATLAB. In: IEEE international symposium on computer aided control systems design. 2004.
[66] El Ghaoui L, Calafiore G. Robust filtering for discrete-time systems with bounded noise and parametric uncertainty. IEEE Trans Automat Control 2001;46(7):1084–9.
[67] Jafari Fesharaki S, Kamali M, Sheikholeslam F. Adaptive tube-based model predictive control for linear systems with parametric uncertainty. IET Control Theory Appl 2017;11(17):2947–53.
[68] Segovia P, Rajaoarisoa L, Nejjari F, Duviella E, Puig V. Model predictive control and moving horizon estimation for water level regulation in inland waterways. J Process Control 2019;76:1–14.
[69] Segovia P, Duviella E, Puig V. Multi-layer model predictive control of inland waterways with continuous and discrete actuators. IFAC-PapersOnLine 2020;53(2):16624–9, 21th IFAC World Congress.