Distributed Partial Output Consensus Optimization for Constrained Chain Interconnected Systems

Zidong Liu[a], Dongya Zhao[a,*], Shuzhan Zhang[a], Xindong Wang[a], Sarah K. Spurgeon[b]

[a] *College of New Energy, China University of Petroleum (East China), Qingdao, 266580, Shandong, China*

[b] *Department of Electronic and Electrical Engineering, University College London, London, WC1E 7JE, United Kingdom*

Abstract

For chain interconnected systems with state and input constraints, a partial output consensus (POC) optimization problem is studied when the set-points are infeasible. In this case, outputs with and without consensus requirements cannot converge to the set-points achieved from real-time optimization. For this case, a novel set-point optimization method is developed, which is called distributed partial output consensus optimization. Based on this method, the set-points for two-part outputs i.e. having a part that must achieve consensus and a part that has a set-point, can be recalculated simultaneously and their feasibility can be ensured by using a distributed projection operator. The convergence of the strategy is then analyzed. From the results of both simulation and experimental testing, the effectiveness of the proposed method is validated.

Keywords: Partial output consensus, chain interconnected systems, infeasible set-point, distributed optimization

[*]Corresponding author

Email addresses: b20150004@s.upc.edu.cn (Zidong Liu), dyzhao@upc.edu.cn (Dongya Zhao), shuzhan@upc.edu.cn (Shuzhan Zhang), b21150002@s.upc.edu.cn (Xindong Wang), s.spurgeon@ucl.ac.uk (Sarah K. Spurgeon)

1. Introduction

In process control, interconnected systems composed of multiple subsystems have particular dynamic characteristics due to their complex internal structures Zhang et al. (2022). The chain interconnected systems as studied in this paper are formed from sequentially interconnected subsystems, resembling a ‘chain’. This is common in large-scale processes such as a fossil fuel power unit Liu et al. (2003), a continuous annealing line Yoshitani (1993) and a seawater distillation process Liu et al. (2023). In these scenarios, a common problem exists which is known as ‘Partial Output Consensus’ (POC), which can be illustrated by the following example. In the seawater distillation process shown in Figure 1, flashing chambers can be considered as subsystems, which have mass and energy interconnections. In this process, the three pressure signals need to maintain a descending relationship, and the temperature signals are determined independently. When designing the controller, the descending relationship can be modelled as a consensus requirement, where the goal is to achieve consensus among the pressures (defined as the ‘Consensus’ output) and make the temperatures (defined as the ‘Nonconsensus’ output) track their respective set-points. This problem was termed Partial Output Consensus (POC) in Liu et al. (2023), and then a distributed robust control algorithm was designed for chain interconnected systems. This work Liu et al. (2023) is based on the assumption that the set-points are in the interiors of the feasible regions, which cannot always be ensured. The set-points are obtained from real-time optimization, and their calculation models and constraints are independent of the underlying dynamic system. This means that the set-points may be infeasible for some subsystems Li et al. (2008); Wu et al. (2021). In Figure 1, once the infeasible set-points are directly used, the actual value of the pressures and temperatures may converge somewhere unknown, and the subsystems may violate the input and state constraints. This may cause interruption of the process and in the worst case can produce accidents. Hence, it is necessary to construct a set-point optimization layer to recalculate the feasible set-points and maintain the characteristics of POC.

For infeasible set-points, many set-point optimization methods have been developed based on constrained optimization and adaptive methods Li et al. (2008); Marchetti et al. (2014); Pang et al. (2015); Wu et al. (2021). However, these methods are typically designed for a single system, and may not be suitable for large-scale optimization and POC problems. Note that the

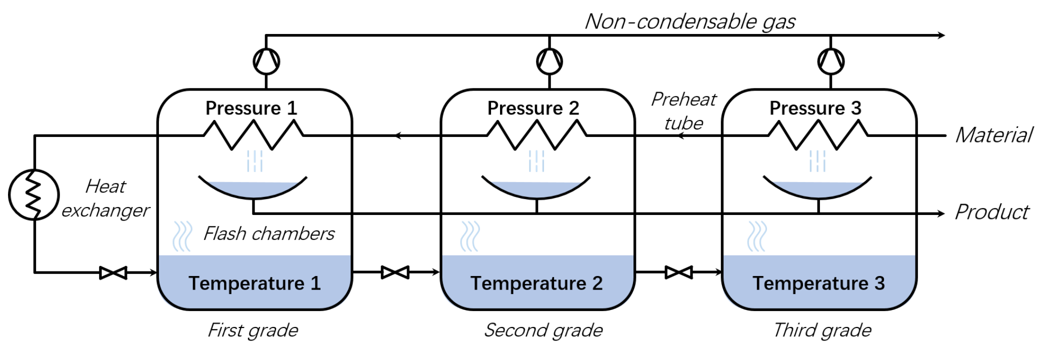


Figure 1: The process of seawater distillation.

issue of an infeasible set-point has already been considered in the work on distributed consensus optimization Liu et al. (2017); Meng et al. (2017); Fontan et al. (2020). However, these methods can only find a common feasible set-point for subsystems to achieve consensus and have not addressed the POC problem. Different from the consensus problem, the feasibility of 'Nonconsensus' set-points also needs to be ensured in POC. This has been studied in Xia et al. (2023) where a partial consensus matrix is designed and a decentralized algorithm is proposed to solve the optimization problem with partial consensus. This method can ensure the consensus of partial variables and the convergence of variables. However, it is only suitable for multi-agent systems whose subsystems have no mass and energy interconnections. It should be noted there is limited work on distributed partial output consensus optimization for chain interconnected systems. Although centralized optimization can also solve this problem the corresponding methods may have high computational cost Yangzhou et al. (2015) and Turan et al. (2021). Distributed consensus optimization has low computational cost and is an ideal solution but the following two challenging issues need to be addressed. The first is how to find the set-points to satisfy both 'Consensus' and 'Nonconsensus' requirements. The second is how to guarantee the feasibility of set-points for chain interconnected systems.

In this paper, a distributed partial output consensus optimization method has been developed for chain interconnected systems to solve the POC problem of infeasible set-points. The proposed approach cannot only achieve POC but also guarantee the feasibility of 'Consensus' and 'Nonconsensus' set-points.

The contributions of this paper are:

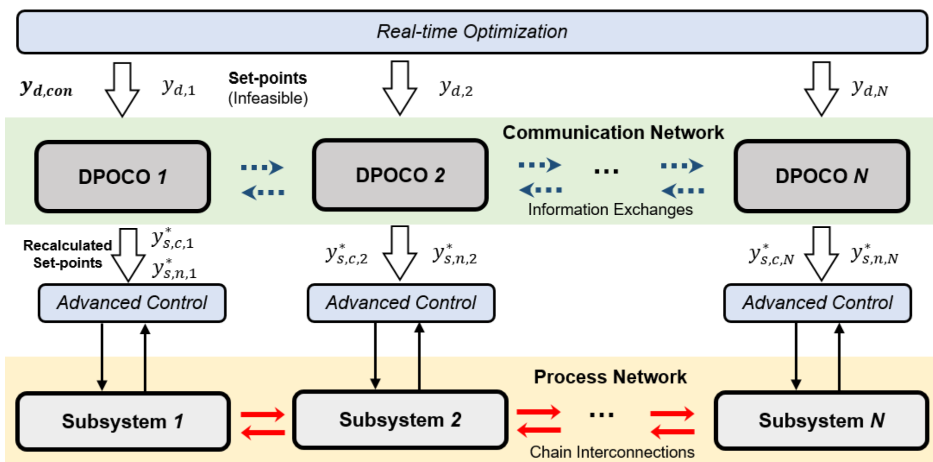
- A distributed partial output consensus method is developed to recalculate the set-points. A new iterative strategy is designed for both 'Consensus' and 'Nonconsensus' set-points. This is achieved by cooperation between subsystems.
- A distributed projection algorithm is derived to ensure the feasibility of the recalculated set-points. By redefining the steady-state variables and design of a consistency constraint, the feasible points for all subsystems can be solved in a distributed manner. Compared with the centralized projection algorithm, the computational burden can be greatly reduced.

The paper is structured as follows. In Section 2, chain interconnected systems are described, and the POC feasible set-point optimal problem is formulated. In section 3, the distributed partial output consensus optimization method is designed, and the algorithm analysis is completed. The results of numerical simulations and experiments are shown in section 4. Finally, section 5 summarizes this paper and gives conclusions.

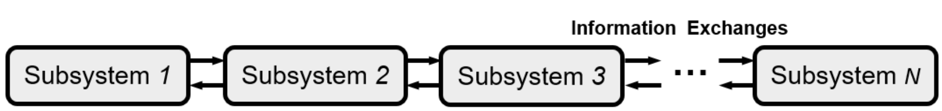
Notation 1. *Note P' and rank(P) denote the transpose and rank of P, respectively. $\mathbf{0}_n$ and $\mathbf{0}_{n\times m}$ represent n-dimensional zero vector and $n \times m$ dimensional zero matrix. The matrix diag$[S_i]_N$ denote the diagonal block matrix composed of $S_1, S_2, \cdots, S_N$. The quadratic norm with respect to a positive definite matrix $P = P'$ is denoted by $\|x\|_P^2 = x'Px$. $\|x\|$ and $\|x\|_\infty$ represent the 2-norm and ∞-norm of x respectively. Let $\mathrm{Proj}_\Omega(u)$ to be a projection operator from $u \in \mathcal{R}^n$ to $\Omega \subseteq \mathcal{R}^n$: $\mathrm{Proj}_\Omega(u) = \mathrm{argmin}_{v\in\Omega} \|v - u\|^2$, where Ω is a closed convex set. The maximum eigenvalue of P is denote by $\lambda_{max}(P)$. $\mathcal{R}$ and $\mathcal{Z}$ respectively represent the set of real numbers and integers, and $\mathcal{Z}_i^j = \{i, i+1, \cdots, j\}$ with $i < j \in \mathcal{Z}$. Denote col$(\cdot)$ as the column vector and define $E_i = col(0, \cdots, \underset{ith}{1}, \cdots, 0)$.*

2. Problem Formulation

The process network considered in this paper is composed of N subsystems with chain interconnections, in which the interconnections refer to the couplings in mass and energy between the subsystems. As shown in Figure 2(a), a typical framework consists of real-time optimization, set-point optimization and advanced control Ding (2017); Yang and Ding (2020), where set-point optimization is named as distributed partial output consensus optimization in this paper. Real time optimization can generate a 'Consensus' set-point $y_{d,con}$ (only assigned to subsystem 1) and a 'Nonconsensus' set-point $y_{d,i}$, $i = 1, \cdots, N$ (assigned to subsystem i), which may be infeasible and cannot be directly utilized for the advanced control operation. Given the set-points from real-time optimization, distributed partial output consensus optimization can recalculate set-points $y^*_{s,c,i}$, $y^*_{s,n,i}$ which are feasible and which can be used for advanced control allowing the subsystems to achieve POC, where $y^*_{s,c,i}$ is the feasible set point to achieve 'Consensus' and $y^*_{s,n,i}$ is the feasible set-point relating to the 'Nonconsensus' element. These set-points are obtained using distributed partial output consensus optimization to meet the partial consensus requirement. In addition, there is a chain communication network (see Figure 2(b)), which can be used to exchange information between the subsystems when the algorithm is running.



(a) The framework for the whole system.



(b) The communication network.

Figure 2: Distributed partial output consensus optimization for Chain interconnected systems.

Remark 1. *The models and constraints used for real-time optimization may be different from those used in advanced control. In this case the set-points* *obtained directly from real-time optimization may be infeasible for control. This may cause the outputs to converge somewhere unknown and subsystems may violate the state and input constraints. It is worth noting that the feasibility of $y_{d,con}$, $y_{d,1}$, $\cdots$, $y_{d,N}$ cannot be distinguished easily. Hence, the distributed partial output consensus optimization layer is designed to recognize the potential problem and recalculate the feasible set-points if necessary.*

Let t represent the discrete-time index. The dynamics of the subsystems can be represented by

$$\begin{aligned} x_i^{t+1} =& \sum_{j=i-1}^{i+1} A_{ij} x_j^t + B_i u_i^t, \\ y_i^t =& C_i x_i^t. \end{aligned} \tag{1}$$

where $i \in \mathcal{V} = \mathcal{Z}_1^N$, $x_i^t \subseteq \mathcal{X}_i \in \mathcal{R}^n$, $u_i^t \subseteq \mathcal{U}_i \in \mathcal{R}^m$, $y_i^t \in \mathcal{R}^p$ and $p \geq m$, represent the state, input and output of subsystem i. The matrices A_{ij}, B_i and C_i are known and have appropriate dimensions. The constraint sets $\mathcal{X}_i, \mathcal{U}_i$ are convex, compact polytopes whose interiors are not empty. Note that each subsystem is 'multiple-output', that is $p \geq 2$. For each subsystem, its output is composed of two parts

$$\begin{aligned} y_{con,i}^t =& T_c y_i^t \in \mathbb{R}^{p_c}, \\ y_{non,i}^t =& T_n y_i^t \in \mathbb{R}^{p_n}, \end{aligned} \tag{2}$$

where $y_{con,i}^t$ is the 'Consensus' output and $y_{non,i}^t$ contains the 'Nonconsensus' element. It follows that $p_c + p_n = p$. The transformation matrices $T_c \in \mathbb{R}^{p_c \times p}$, $T_n \in \mathbb{R}^{p_n \times p}$ are both row full rank and satisfy $col(T_c, T_n) = I_p$.

Then, the steady-state model for subsystem i can be formulated as

$$\begin{aligned} \mathbf{0}_n =& (A_{ii} - I) x_{s,i} + B_i u_{s,i} + \sum_{j \in \mathcal{N}_{c,i}} A_{ij} x_{s,j}, \\ y_{s,c,i} =& T_c C_i x_{s,i}, \\ y_{s,n,i} =& T_n C_i x_{s,i}, \end{aligned} \tag{3}$$

where $x_{s,i}$ and $u_{s,i}$ represent the steady-state state and input respectively, $y_{s,c,i}$ and $y_{s,n,i}$ represent the 'Consensus' and 'Nonconsensus' steady-state outputs respectively and $x_{s,j}$ is the steady-state state of the neighbours. The steady-state models are coupled and $\mathcal{N}_{c,i}$ represents the neighbours set where

$$\mathcal{N}_{c,i} = \begin{cases} \{2\}, i = 1 \\ \{i-1, i+1\}, i \in \mathcal{Z}_2^{N-1} \\ \{N-1\}, i = N \end{cases}.$$

The following assumptions are stated:

Assumption 1. *Farina et al. (2014) The parameters in* (1) *satisfy the following condition*

$$rank\left(\begin{bmatrix} I_{nN} - A & B \\ C & \mathbf{0}_{pN\times mN} \end{bmatrix}\right) = (n+p)\,N, \tag{4}$$

where $A = [A_{ij}]_{N\times N}$, $B = diag[B_i]_N$, $C = diag[C_i]_N$.

Assumption 2. *This determines the conditions which must be satisfied by the communication network.*

- *For any two subsystems, if there are interconnections between them, they can exchange information with each other, that is, the communication network is chained. The Laplace matrix is*

$$L = \begin{bmatrix} 1 & -1 & & & \mathbf{0} \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ \mathbf{0} & & & -1 & 1 \end{bmatrix}_{N\times N}. \tag{5}$$

- *Information can be sent and received at every time instant without packet loss and delay.*

Remark 2. *Assumption 2 concerning the communication network is reasonable. In process industry plants, communication among devices is usually wired, and the network structure is fixed, stable and reliable.*

In Liu et al. (2023), all the set-points from real-time optimization are assumed to be feasible, and then a distributed POC control is designed to steer the subsystems to achieve POC whereby:

$$\lim_{t\to\infty}\left\|y^t_{con,1}-y_{d,con}\right\| =0, \tag{6a}$$

$$\lim_{t\to\infty}\left\|y^t_{non,i}-y_{d,i}\right\| =0, \tag{6b}$$

$$\lim_{t\to\infty}\left\|y^t_{con,i}-y^t_{con,j}\right\| =0, j\in\mathcal{N}_{c,i}. \tag{6c}$$

In practice, since real-time optimization and advanced control depend on different models and constraints Wu et al. (2021), $y_{d,con}, y_{d,1}, \cdots, y_{d,N}$ may be infeasible for some subsystems, so the control in Liu et al. (2023) may not be effective. In this paper a distributed partial output consensus optimization is designed to recalculate feasible set-points $(y^*_{s,c,i}, y^*_{s,n,i})$, which meet the partial consensus requirement and can be utilized by the advanced control operation.

According to the partial consensus requirement, $y^*_{s,c,i}$, $y^*_{s,n,i}$ are formulated and the POC feasible set-point optimal problem is established. To guarantee the feasibility of $y^*_{s,c,i}, y^*_{s,n,i}$ for control, the feasible region $\mathcal{Y}_s$ for the whole system needs to be formulated. This can be derived from the the steady-state model in (3) and can be expressed as

$$\begin{aligned}\mathcal{Y}_s =&\{\left(y_{s,c,1},y_{s,n,1}\right),\cdots,\left(y_{s,c,N},y_{s,n,N}\right)\mid\\ &y_{s,c,i}=T_cC_ix_{s,i}, y_{s,n,i}=T_nC_ix_{s,i},\\ &\mathbf{0}_n=\left(A_{ii}-I\right)x_{s,i}+B_iu_{s,i}+\sum_{j\in\mathcal{N}_{c,i}}A_{ij}x_{s,j},\\ &i\in\mathcal{V}, x_{s,i}\in\mathcal{X}_{s,i}, u_{s,i}\in\mathcal{U}_{s,i}\}\\ =&\mathcal{Y}_{s,1}\times\mathcal{Y}_{s,2}\times\cdots\times\mathcal{Y}_{s,N},\end{aligned} \tag{7}$$

where $\mathcal{X}_{s,i}\subseteq\mathcal{X}_i$, $\mathcal{U}_{s,i}\subseteq\mathcal{U}_i$ are the steady-state state and input constraint sets and $\mathcal{Y}_{s,i}$ is the local feasible region of $(y_{s,c,i}, y_{s,n,i})$. It can be inferred that $\mathcal{Y}_{s,i}$ is coupled with $\mathcal{Y}_{s,j}$, $j\in\mathcal{N}_{c,i}$. When the set-points are within $\mathcal{Y}_s$, they are feasible and can be reached by the subsystems deploying the advanced control.

Based on this, $y^*_{s,c,i}$, $y^*_{s,n,i}$ can be formulated as

$$(y^*_{s,c,i}, y^*_{s,n,i})=\mathrm{Proj}_{\mathcal{Y}_{s,i}}(y_{d,con}, y_{d,i}), i\in\mathcal{V}, \tag{8a}$$

$$\lim_{k\to\infty}\left\|y^*_{s,c,i}-y^*_{s,c,j}\right\|=0, j\in\mathcal{N}_{c,i}, \tag{8b}$$

where (8a) means that $(y^*_{s,c,i}, y^*_{s,n,i})$ is the projection of $(y_{d,con}, y_{d,i})$ into $\mathcal{Y}_{s,i}$, so that $y^*_{s,c,i}$, $y^*_{s,n,i}$ are feasible for subsystem i. Equation (8b) represents the consensus of $y^*_{s,c,i}$, $i\in\mathcal{V}$.

Remark 3. *When the set-points from real-time optimization are infeasible, the conditions in* (6) *cannot be satisfied and the control method in Liu et al. (2023) is not effective. In this case,* $(y^*_{s,c,i}, y^*_{s,n,i})$ *described in* (8) *can be found to replace the set-points resulting from real-time optimization.* $(y^*_{s,c,i}, y^*_{s,n,i})$ *is not only feasible in control, but also meets the partial consensus requirement. If infeasible set-points were directly used by the advanced control,* $y^t_{con,i}$ *and* $y^t_{non,i}$ *will converge somewhere unknown. This assertion will be demonstrated in the experiments.*

According to (8), the POC feasible set-point optimal problem can be formulated as

$$\min_{\tilde{y}_{s,c,i},\tilde{y}_{s,n,i},i\in\mathcal{V}} \quad \|\tilde{y}_{s,c,1} - y_{d,con}\|^2 + \sum_{i\in\mathcal{V}} \|\tilde{y}_{s,n,i} - y_{d,i}\|^2 \tag{9a}$$

$$s.t. \quad \text{for all } i \in \mathcal{V},$$

$$(\tilde{y}_{s,c,i}, \tilde{y}_{s,n,i}) \in \mathcal{Y}_{s,i}, \tag{9b}$$

$$\tilde{y}_{s,c,i} - \tilde{y}_{s,c,j} = 0, j \in \mathcal{N}_{c,i}, \tag{9c}$$

where $\tilde{y}_{s,c,i}, \tilde{y}_{s,n,i}$ are the recalculated set-points and (9a) represents the Euclidean distance between $(\tilde{y}_{s,c,1}, \tilde{y}_{s,n,1}, \cdots, \tilde{y}_{s,n,N})$ and $(y_{d,con}, y_{d,1}, \cdots, y_{d,N})$. (9b) is the constraint on the feasible regions, corresponding to (8a). (9c) is the consensus constraint of $\tilde{y}_{s,c,i}$, corresponding to (8b). Then, it can be ensured that the optimal solution of (9) is $(y^*_{s,c,i}, y^*_{s,n,i})$, $i \in \mathcal{V}$.

The control objectives of this paper can now be summarized more formally as follows:

- To propose a distributed partial output consensus optimization method, which can solve the problem (9) and find $(y^*_{s,c,i}, y^*_{s,n,i})$, $i \in \mathcal{V}$.
- To design a distributed projection algorithm, which can guarantee the feasibility of $(y^*_{s,c,i}, y^*_{s,n,i})$ in $\mathcal{Y}_{s,i}$ and reduce the computational burden.

3. Main Results

In this section, according to the partial consensus requirement, an iterative strategy is designed for distributed partial output consensus optimization, and then its convergence is proven. To guarantee the feasibility of the recalculated set-points, a distributed projection algorithm is proposed in the iterative strategy.

For subsystem i, the corresponding recalculated set-points at the k-th iteration are denoted as $y_{s,c,i}^k$, $y_{s,n,i}^k$. Inspired by the projected consensus algorithm in Liu et al. (2017), the strategies of distributed partial output consensus optimization are formulated as follows.

For subsystem 1:

$$\tilde{y}_{s,c,1}^k = \varepsilon_{c,11} y_{s,c,1}^{k-1} + \varepsilon_{c,12} y_{s,c,2}^{k-1} + \alpha_c v_1^{k-1} + \delta_c y_{d,con}, \tag{10a}$$

$$\tilde{y}_{s,n,1}^k = \varepsilon_n y_{s,n,1}^{k-1} + \delta_n y_{d,1}, \tag{10b}$$

$$(y_{s,c,1}^k, y_{s,n,1}^k) = \mathrm{Proj}_{\mathcal{Y}_{s,1}} \left(\tilde{y}_{s,c,1}^k, \tilde{y}_{s,n,1}^k \right), \tag{10c}$$

$$v_1^k = v_1^{k-1} + L_{11} y_{s,c,1}^k + L_{12} y_{s,c,2}^k, \tag{10d}$$

where $\tilde{y}_{s,c,1}^k$, $\tilde{y}_{s,n,1}^k$ are the estimates of $y_{s,c,i}^*$, $y_{s,n,i}^*$ respectively, v_1^k is the estimated error between $y_{s,c,i}^k$ and $y_{s,c,j}^k$, $\varepsilon_{c,11} = 1 - \alpha_c (2 + L_{11}), \varepsilon_{c,12} = -\alpha_c L_{12}, \delta_c = 2\alpha_c, \varepsilon_n = 1 - 2\alpha_n, \delta_n = 2\alpha_n$, L_{ij} represents the element of L in row i and column j, and α_c, $\alpha_n > 0$ are the step sizes, where

$$\alpha_c \leq 1/\lambda_{\max} \left(4\bar{T} + (L \otimes T_c' T_c) \right), \alpha_n < 1,$$

and $\bar{T} = diag \left(I_p, \underbrace{T_c' T_c, \cdots, T_c' T_c}_{N-1} \right)$.

For subsystem $i = 2, \cdots, N$:

$$\tilde{y}_{s,c,i}^k = \varepsilon_{c,ii} y_{s,c,i}^{k-1} + \sum_{j \in \mathcal{N}_{c,i}} \varepsilon_{c,ij} y_{s,c,j}^{k-1} + \alpha_c v_i^{k-1}, \tag{11a}$$

$$\tilde{y}_{s,n,i}^k = \varepsilon_n y_{s,n,i}^{k-1} + \delta_n y_{d,i}, \tag{11b}$$

$$(y_{s,c,i}^k, y_{s,n,i}^k) = \mathrm{Proj}_{\mathcal{Y}_{s,i}} \left(\tilde{y}_{s,c,i}^k, \tilde{y}_{s,n,i}^k \right), \tag{11c}$$

$$v_i^k = v_i^{k-1} + L_{ii} y_{s,c,i}^k + \sum_{j \in \mathcal{N}_{c,i}} L_{ij} y_{s,c,j}^k. \tag{11d}$$

where $\varepsilon_{c,ii} = 1 - \alpha_c L_{ii}, \varepsilon_{c,ij} = -\alpha_c L_{ij}$.

Equations (10a) and (11a) contain the update strategies for $y_{s,c,i}^k$, which can reduce the distances between $\tilde{y}_{s,c,i}^k$ with $y_{d,con}$ and decrease the errors among $y_{s,c,i}^k, i \in \mathcal{V}$ using the information from the neighbours'. Combined with (10d) and (11d), (10a) and (11a) can make $\tilde{y}_{s,c,i}^k$ achieve consensus. (10b) and (11b) are the update strategies for $y_{s,n,i}^k$, which can reduce the distances between $\tilde{y}_{s,n,i}^k$ with $y_{d,i}$. (10c) and (11c) are the projection algorithms, which can find the feasible points of $\tilde{y}^k_{s,c,i}$, $\tilde{y}^k_{s,n,i}$ in $\mathcal{Y}_{s,i}$ by implementing the following distributed projection algorithm

$$
\begin{aligned}
&(y^k_{s,c,i}, y^k_{s,n,i}) = \\
&\underset{\hat{y}_{s,c,i},\hat{y}_{s,n,i}}{\arg\min} \quad \left\|\hat{y}_{s,c,i} - \tilde{y}^k_{s,c,i}\right\|_2^2 + \left\|\hat{y}_{s,n,i} - \tilde{y}^k_{s,n,i}\right\|_2^2 && \text{(12a)}\\
&\quad s.t. \quad \Pi_{i,i} z_{i,i} + \sum_{j\in\mathcal{N}_{c,i}} \Pi_{i,j} z_{i,j} = \mathbf{0}_n, && \text{(12b)}\\
&\qquad\quad z_{i,i} - z_{j,i} = \mathbf{0}_n, j \in \mathcal{N}_{c,i}, && \text{(12c)}\\
&\qquad\quad \hat{y}_{s,c,i} = T_c \bar{C}_i z_{i,i}, && \text{(12d)}\\
&\qquad\quad \hat{y}_{s,n,i} = T_n \bar{C}_i z_{i,i}, && \text{(12e)}\\
&\qquad\quad z_{i,i} \in \mathcal{X}_{s,i} \times \mathcal{U}_{s,i}. && \text{(12f)}
\end{aligned}
$$

where $\hat{y}_{s,c,i}, \hat{y}_{s,n,i}$ are the feasible points to be sought and (12a) represents the Euclidean distance between $(\hat{y}_{s,c,i}, \hat{y}_{s,n,i})$ and $(\tilde{y}^k_{s,c,i}, \tilde{y}^k_{s,n,i})$. The parameters are $\Pi_{i,i} = [A_{ii} - I_n, B_i]$, $\Pi_{i,j} = [A_{ij}, 0_{n\times m}]$ and $\bar{C}_i = diag(C_i, \mathbf{0}_{m\times m})$. Equations (12b), (12d) and (12e) refer to the steady-state model of subsystem i, and they form the feasible region in (7) with (12f). $z_{i,i} = col(x^i_{s,i}, u^i_{s,i})$ is the steady-state variable of subsystem i and $x^i_{s,i}, u^i_{s,i}$ represents the steady-state state and input of subsystem i corresponding to $\hat{y}_{s,c,i}, \hat{y}_{s,n,i}$. $z_{i,j} = col(x^i_{s,j}, u^i_{s,j})$ is the steady-state variable of subsystem j and $x^i_{s,j}, u^i_{s,j}$ represent the steady-state state and input of subsystem j as appears in subsystem i, and (12c) is the consistency constraint of $z_{i,i}$ and $z_{j,i}$, $j \in \mathcal{N}_{c,i}$. For subsystem i, the steady-state variable is denoted by ($z_{i,i}$ and $z_{i,j}$) in the steady-state model of subsystem i and j, $j \in \mathcal{N}_{c,i}$. In this way, the steady-state model can be represented by (12b), which has no interconnection. For the correctness of the results, there is a novel constraint (12c) designed to ensure that $z_{i,i}$ and $z_{i,j}$, $j \in \mathcal{N}_{c,i}$ are consistent. Then, (12b) and (12c) are equivalent to the steady-state model in (3) and (7).

Clearly equation (12) is a distributed optimization problem with a consistency constraint, which can be solved by using the Alternating Direction Method of Multipliers (ADMM) described in Boyd et al. (2011); Falsone et al. (2020); Yan et al. (2020). Compared to the centralized projection algorithm, there is only limited information ($z_{i,j}$ and $z_{j,i}$) that needs to be shared between subsystems i and j. Not only that, as the number of subsystems increases, (12) has more advantages than the corresponding centralized case in terms of computational efficiency. This will be verified in the simulation testing that follows.

Remark 4. *To demonstrate the advantages of the distributed projection algorithm, a centralized one has been provided for comparison. The centralised projection algorithm can be formulated as*

$$
\begin{aligned}
&(y^k_{s,c,1}, \cdots, y^k_{s,c,N}, y^k_{s,n,1}, \cdots, y^k_{s,c,N}) = \\
&\arg\min_{\hat{y}_{s,c,i}, \hat{y}_{s,n,i}, i\in\mathcal{V}} \sum_{i\in\mathcal{V}} \left\|\hat{y}_{s,c,i} - \tilde{y}^k_{s,c,i}\right\|_2^2 + \left\|\hat{y}_{s,n,i} - \tilde{y}^k_{s,n,i}\right\|_2^2 && \text{(13a)} \\
&\text{s.t.} \quad \text{for all } i \in \mathcal{V}, \\
&\quad \mathbf{0}_n = (A_{ii} - I)\,\hat{x}_{s,i} + B_i \hat{u}_{s,i} \\
&\qquad\qquad + \sum_{j\in\mathcal{N}_{c,i}} A_{ij}\hat{x}_{s,j}, j \in \mathcal{N}_{c,i}, && \text{(13b)} \\
&\quad \hat{y}_{s,c,i} = T_c C_i \hat{x}_{s,i}, && \text{(13c)} \\
&\quad \hat{y}_{s,n,i} = T_n C_i \hat{x}_{s,i}, && \text{(13d)} \\
&\quad x_{s,i} \in \mathcal{X}_{s,i}, u_{s,i} \in \mathcal{U}_{s,i}, && \text{(13e)}
\end{aligned}
$$

where $\hat{x}_{s,i}$, $\hat{u}_{s,i}$ are the steady-state state and input corresponding to the feasible set-points and (13b)-(13e) *form the feasible region in* (7). *Different from* (12)*, the interconnection term $\sum_{j\in\mathcal{N}_{c,i}} A_{ij}\hat{x}_{s,j}$ in* (13b) *means* (13) *requires global information and a central node for implementation. This brings greater challenges in terms of the construction of the communication network. Hence,* (12) *is more practical than* (13).

Based on the distributed projection algorithm, the proposed distributed partial output consensus optimization in (10) and (11) can be implemented in a fully distributed manner. For clarity, the distributed partial output consensus optimization method can be summarized in Algorithm 1 and its flow chart is shown in Figure 3. Then, the convergence analysis of (10) and (11) is given in the following theorem.

Theorem 1. *Suppose that Assumptions 1 and 2 hold, then for subsystem $i, i \in \mathcal{V}$ in* (1)*, $(y^k_{s,c,i}, y^k_{s,n,i})$ in* (10) *and* (11) *can finally converge to the optimal solution of* (9)*, that is, the feasible set-points $(y^*_{s,c,i}, y^*_{s,n,i})$ with partial consensus can be found by distributed partial output consensus optimization.*

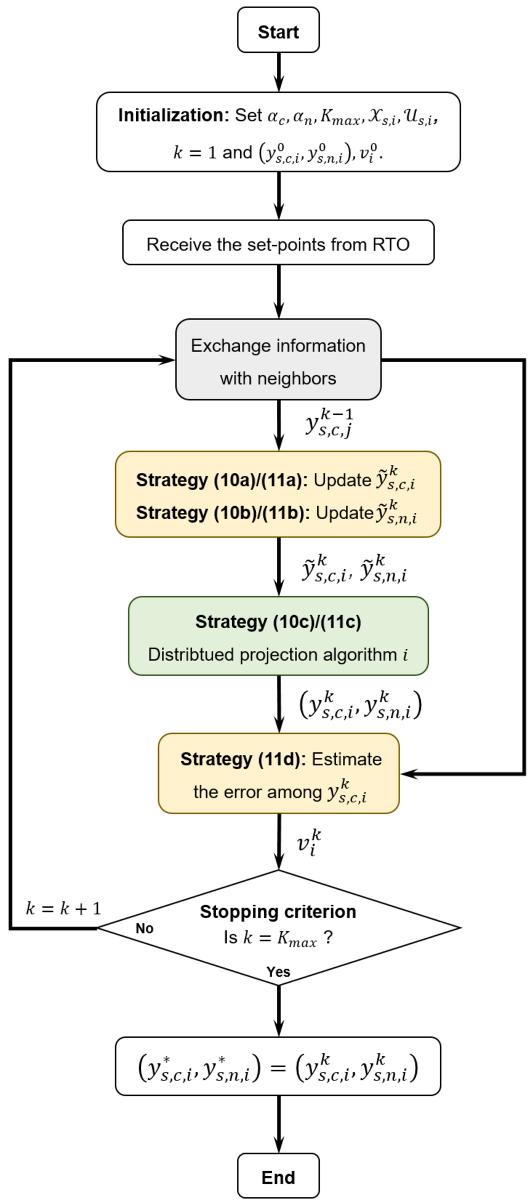


Figure 3: Flow chart of Algorithm 1.

Algorithm 1 Distributed partial output consensus optimization for chain interconnected systems

Initialization: Set α_c, α_n, K_{max}, $k=1$ and the initial value of $(y^0_{s,c,i}, y^0_{s,n,i})$, v^0_i. Then, choose $\mathcal{X}_{s,i}$, $\mathcal{U}_{s,i}$.

For subsystem i:

1: If $i=1$, receive $y_{d,con}$,$y_{d,1}$; else if, receive $y_{d,i}$.

2: Send $y^{k-1}_{s,c,i}$ and receive $y^{k-1}_{s,c,j}$ based on the communication network.

3: Calculate $(\tilde{y}^k_{s,c,i}, \tilde{y}^k_{s,n,i})$ by (10a), (10b)/(11a), (11b).

4: Solve (12) by ADMM, then obtain $(y^k_{s,c,i}, y^k_{s,n,i})$.

5: Calculate v^k_i by (10d)/(11d).

6: If $k=K_{max}$, let $(y^*_{s,c,i}, y^*_{s,n,i}) = (y^k_{s,c,i}, y^k_{s,n,i})$ and end the algorithm; else if, let $k=k+1$ and go to Step.2.

Result: The feasible set-point $(y^*_{s,c,i}, y^*_{s,n,i})$.

Proof. To make it clear, the 'Consensus' feasible region $\mathcal{Y}_{s,c,i}$ and the corresponding 'Nonconsensus' feasible region $\mathcal{Y}_{s,n,i}$ are defined as

$$\mathcal{Y}_{s,c,i} = T_c\mathcal{Y}_{s,i},$$
$$\mathcal{Y}_{s,n,i} = T_n\mathcal{Y}_{s,i},$$

and $\mathcal{Y}_{s,c}$, $\mathcal{Y}_{s,n}$ represent the compact forms of $\mathcal{Y}_{s,c,i}$ and $\mathcal{Y}_{s,n,i}$, that is

$$\mathcal{Y}_{s,c,1} \times \mathcal{Y}_{s,c,2} \times \cdots \times \mathcal{Y}_{s,c,N} = \mathcal{Y}_{s,c},$$
$$\mathcal{Y}_{s,n,1} \times \mathcal{Y}_{s,n,2} \times \cdots \times \mathcal{Y}_{s,n,N} = \mathcal{Y}_{s,n}.$$

According to the results in Liu et al. (2017), Liu and Wang (2013) and Yang et al. (2018), with the protocol in (10a), (10c), (10d) and (11a), (11c), (11d), $y^k_{s,c,i}$ can achieve consensus at an equilibrium point, which is denoted as $y^\circ_{s,c,i}$. According to (10) and (11), the equilibrium point can be formulated as

$$\begin{aligned} y^\circ_{s,c} =& \varepsilon_c y^\circ_{s,c} + \alpha_c v^\circ + 2\alpha_c \left(E_1 \otimes I_{p_c}\right) y_{d,con}, \\ y^\circ_{s,c} =& \mathrm{Proj}_{\mathcal{Y}_{s,c}} \left(y^\circ_{s,c}\right), \\ v^\circ =& v^\circ + \left(L \otimes I_{p_c}\right) y^\circ_{s,c}, \end{aligned} \tag{14}$$

where $y^\circ_{s,c} = col(y^\circ_{s,c,1}, \cdots, y^\circ_{s,c,N})$, $v^\circ = col(v^\circ_1, \cdots, v^\circ_N)$, $\varepsilon_c = [\varepsilon_{c,ij}]_{i,j\in\mathcal{V}}$. From (10) and (11), it can be inferred that $\varepsilon_c = I_{nN} - (2\alpha_c\left(E_1E_1' \otimes I_{p_c}\right) + (L \otimes I_n))$.

Let $v^{\circ} = (L \otimes I_{p_c})\, z^{\circ}$, where $z^{\circ} \in \mathcal{R}^{p_c N}$. Then (14) can be rewritten as

$$\begin{aligned} & y_{s,c}^{\circ} - \mathrm{Proj}_{\mathcal{Y}_{s,c}}\big(y_{s,c}^{\circ} - \alpha_c\,((L \otimes I_{p_c})\, z^{\circ} \\ & \quad + \underbrace{2\,(E_1 E_1' \otimes I_{p_c})\, y_{s,c}^{\circ} - (E_1 \otimes I_{p_c})\, y_{d,con}}_{\partial\left(\left\|y_{s,c,1}-y_{d,con}\right\|^2\right)/\partial y_{s,c}\big|_{y_{s,c}=y_{s,c}^{\circ}}})\big) = 0, \\ & (L \otimes I_{p_c})\, y_{s,c}^{\circ} = 0. \end{aligned} \tag{15}$$

Similarly, with (10b), (10c) and (11b), (11c), $y_{s,n,i}$ can reach its equilibrium point $y_{s,n,i}^{\circ}$, which satisfies

$$\begin{aligned} & y_{s,n}^{\circ} - \mathrm{Proj}_{\mathcal{Y}_{s,n}}\big(y_{s,n}^{\circ} \\ & \quad - \alpha_n \cdot \underbrace{2\left(y_{s,n}^{\circ} - y_d\right)}_{\partial\left(\|y_{s,n}-y_d\|^2\right)/\partial y_{s,n}\big|_{y_{s,n}=y_{s,n}^{\circ}}}\big) = 0. \end{aligned} \tag{16}$$

where $y_{s,n}^{\circ} = col(y_{s,n,1}^{\circ}, \cdots, y_{s,n,N}^{\circ})$ and $y_d = col(y_{d,1}, \cdots, y_{d,N})$.

According to the necessary and sufficient conditions for optimality in Liu and Wang (2013), for the optimal solution $(y_{s,c}^{*}, y_{s,n}^{*})$ of (9), there must exist $z^{*} \in \mathcal{R}^{p_c N}$ so that the following equalities hold

$$\begin{aligned} & \begin{bmatrix} y_{s,c}^{*} \\ y_{s,n}^{*} \end{bmatrix} - \mathrm{Proj}_{\mathcal{Y}_s}\Big(\begin{bmatrix} y_{s,c}^{*} \\ y_{s,n}^{*} \end{bmatrix} \\ & \quad - \alpha\left(\nabla J\left(y_{s,c}^{*}, y_{s,n}^{*}\right) + (L \otimes I_{p_c})\, z^{*}\right)\Big) = 0, \\ & (L \otimes I_{p_c})\, y_{s,c}^{*} = 0. \end{aligned} \tag{17}$$

where

$$\nabla J\left(y_{s,c}^{*}, y_{s,n}^{*}\right) = \begin{bmatrix} 2\,(E_1 E_1' \otimes I_{p_c})\, y_{s,c}^{*} - (E_1 \otimes I_{p_c})\, y_{d,con} \\ 2\left(y_{s,n}^{*} - y_d\right) \end{bmatrix}.$$

Clearly, from (15) and (16), $(y_{s,c}^{\circ}, y_{s,n}^{\circ})$ satisfies (17) , that is, $(y_{s,c}^{\circ}, y_{s,n}^{\circ}) = (y_{s,c}^{*}, y_{s,n}^{*})$. Based on the above analysis, for each subsystem, $(y_{s,c,i}^{k}, y_{s,n,i}^{k})$ in (10) and (11) can converge to the optimal solution $(y_{s,c,i}^{*}, y_{s,n,i}^{*})$ of (9). This means that distributed partial output consensus optimization can recalculate the feasible set-points with partial consensus requirement, and then Theorem 1 is proven. ■

Remark 5. *The proposed method is also suitable for chain interconnected systems with strong interconnections. In this paper, the interconnections are* *handled by redefining steady-state variables and designing a novel consistency constraint, rather than relying on the assumptions that the interconnections are bounded like those in René et al. (2019) and Yang et al. (2021). This method is universally applicable to the systems described by* (1)*.*

Remark 6. *In contrast to Li et al. (2008), Marchetti et al. (2014), Pang et al. (2015) and Wu et al. (2021), the outputs which have a partial consensus requirement are considered in this paper. Based on the iterative strategy in the distributed partial output consensus optimization, partial consensus and the feasibility of* $(y^*_{s,c,i}, y^*_{s,n,i})$ *can be guaranteed, respectively. Finally, the new set-points can be transmitted to the advanced control for achieving POC.*

4. SIMULATION AND EXPERIMENTAL VALIDATION

4.1. Numerical simulation

In this section, several numerical experiments are presented to validate the effectiveness of the proposed method. Consider the chain interconnected systems composed of five subsystems ($i = 1, 2, 3, 4, 5$), where the parameters are given by

$$A_{ii} = \begin{bmatrix} 1 & 0 & 0.8 \\ 0.1 * i & 0.9 & 0 \\ 0 & 0.3 & 0.8 \end{bmatrix}, B_i = \begin{bmatrix} 1 & 0 \\ 2 & 0 \\ 0 & 1 + 0.1 * i \end{bmatrix},$$
$$C_i = \begin{bmatrix} 1 & 0 & 0.2 \\ 0 & 1 & 0 \end{bmatrix},$$

and the interconnected terms are

$$A_{i,i+1} = \begin{bmatrix} 0 & 0 & 0.3 \\ 0 & 0.1 & 0 \\ 0.3 & 0 & 0 \end{bmatrix}, A_{i-1,i} = \begin{bmatrix} 0 & 0.1 & 0 \\ 0.2 & 0 & 0 \\ 0 & 0 & 0.4 \end{bmatrix}.$$

The constraint sets are set as

$$\begin{aligned} \mathcal{X}_i &= \{x_i \,|-5 \le \|x_i\|_\infty \le 5\}, \\ \mathcal{U}_i &= \{u_i \,|-2 \le \|u_i\|_\infty \le 2\}, i \in \mathcal{V}, \end{aligned} \tag{18}$$

and let $\mathcal{X}_{s,i} = \mathcal{X}_i$, $\mathcal{U}_{s,i} = \mathcal{U}_i$. The step size α_c and α_n are set as 0.3 and 0.1, respectively. The maximum number of iterations K_{max} is set as 150.

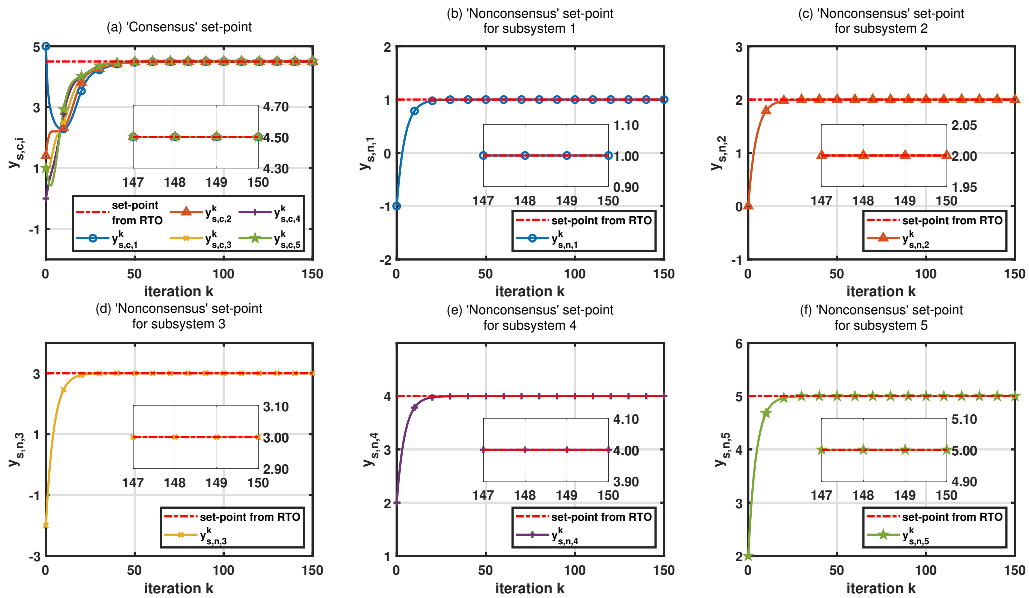


Figure 4: The behaviors of $y^k_{s,c,i}$, $y^k_{s,n,i}$ in case 1.

The initial values are set as $y^0_{s,1} = [5.00, -1.00]$, $y^0_{s,2} = [1.40, 00]$, $y^0_{s,3} = [1.00, -2.00]$, $y^0_{s,4} = [0, 2.00]$, $y^0_{s,5} = [1.00, 2.00]$.

There are four cases considered to fully demonstrate the effectiveness of the proposed method. Cases 1-3 present the results using distributed partial output consensus optimization, when the set-points from real-time optimization are all feasible, partially feasible and all infeasible, respectively. The corresponding behaviors of $y^k_{s,c,i}$, $y^k_{s,n,i}$ in the iteration are shown in Figure 4, Figure 5 and Figure 6, respectively. Case 4 presents the comparative results of centralized and distributed projection algorithms in terms of computational efficiency, and the computation time is shown in TABLE 1.

Case 1: All the set-points from real-time optimization are feasible.

For the subsystems with the constraint sets (18), the following set-points from real-time optimization are feasible: $y_{d,con} = 4.50$, $y_{d,1} = 1.00$, $y_{d,2} = 2.00$, $y_{d,3} = 3.00$, $y_{d,4} = 4.00$, $y_{d,5} = 5.00$. The behaviors of $y^k_{s,c,i}$ and $y^k_{s,c,i}$ are shown in Figure 4(a) and Figure 4(b-f), respectively, in which the horizontal coordinate is the number of iterations k, and the vertical coordinate is the value of $y^k_{s,c,i}$ or $y^k_{s,c,i}$. Chain lines represent the set-points from real-time optimization, solid lines with different markers represent the $y^k_{s,c,i}$, $y^k_{s,c,i}$ of

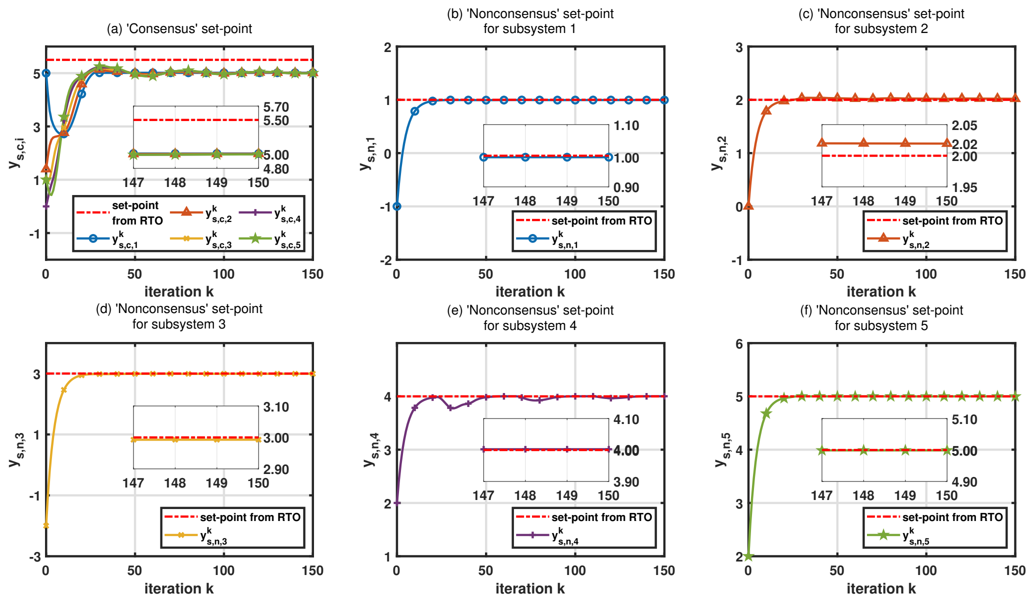


Figure 5: The behaviors of $y^k_{s,c,i}$, $y^k_{s,n,i}$ in case 2.

different subsystems and dashed lines represent the recalculated set-points.

From Figure 4(a), it can be seen that the distances between $y^k_{s,c,i}$ for all subsystems initially decrease, and then all $y^k_{s,c,i}$ try to approach $y_{d,con}$ and finally achieve consensus at $y_{d,con}$. At the same time, $y^k_{s,c,i}$ for all subsystems in Figure 4(b-f) can converge to $y_{d,i}$, $i = 1, 2, 3, 4, 5$ respectively. Hence, the recalculated set-points can be obtained as $y^*_{s,c,1} = \cdots = y^*_{s,c,5} = 4.50$, $y^*_{s,n,1} = 1.00$, $y^*_{s,n,2} = 2.00$, $y^*_{s,n,3} = 3.00$, $y^*_{s,n,4} = 4.00$ and $y^*_{s,n,5} = 5.00$, which are the same as the set-points obtained from real-time optimization.

Case 2: Partial set-points from real-time optimization are infeasible.

When $y_{d,con} = 5.50$, $y_{d,1} = 1.00$, $y_{d,2} = 2.00$, $y_{d,3} = 3.00$, $y_{d,4} = 4.00$, $y_{d,5} = 5.00$, only $y_{d,con}$ and $y_{d,2}$ are infeasible for the subsystems with the constraint set (18). In Figure 5(a), after about 10 iterations, all $y^k_{s,c,i}$ try to approach $y_{d,con}$ but fail due to the infeasibility of $y_{d,con}$. By the collaboration between the consensus and projection algorithms in the distributed partial output consensus optimization method, $y^k_{s,c,i}$ achieves consensus and finally converges to 5.00, and $y^k_{s,n,2}$ converges to 2.02 in Figure 5(c). $y^k_{s,n,i}$, $i = 1, 3, 4, 5$ can also converge to $y_{d,i}$. Then, partial consensus is guaranteed and the recalculated set-points can be obtained as $y^*_{s,c,i} = 5.00$ and $y^*_{s,n,i} = y_{d,i}$

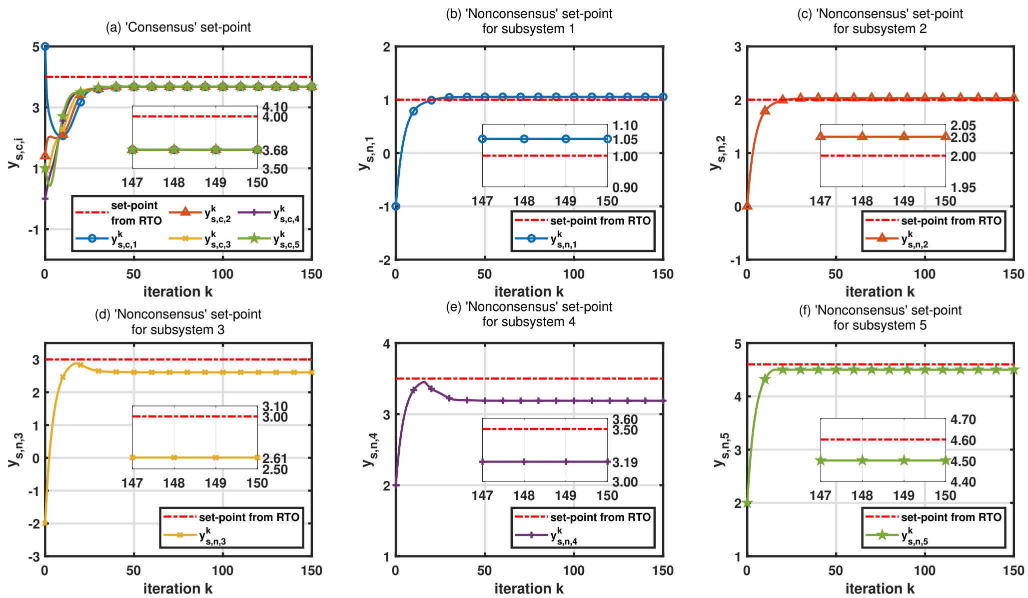


Figure 6: The behaviors of $y^k_{s,c,i}$, $y^k_{s,n,i}$ in case 3.

for $i = 1, 2, 3, 4, 5$. To verify the feasibility of $y^*_{s,c,i}$, the corresponding steady-state states and inputs $(x^*_{s,i}, u^*_{s,i})$ are shown as

$$\begin{aligned}
x^*_{s,1} &= [5.00; 0.99; 0.07]\,, u^*_{s,1} = [-0.30; -1.60]\,,\\
x^*_{s,2} &= [4.84; 2.02; 0.81]\,, u^*_{s,2} = [-1.03; -1.60]\,,\\
x^*_{s,3} &= [4.81; 2.99; 0.94]\,, u^*_{s,3} = [-1.26; -1.90]\,,\\
x^*_{s,4} &= [4.80; 4.00; 1.01]\,, u^*_{s,4} = [-1.48; -2.00]\,,\\
x^*_{s,5} &= [4.74; 5.00; 1.27]\,, u^*_{s,5} = [-1.41; -1.10]\,,
\end{aligned}$$

which do not violate the constraints in equation (18).

Case 3: All the set-points from real-time optimization are infeasible.

When the state and input constraints are

$$\begin{aligned}
\mathcal{X}_i &= \{x_i \,|-4.5 \le \|x_i\|_\infty \le 4.5\}\,,\\
\mathcal{U}_i &= \{u_i \,|-1.5 \le \|u_i\|_\infty \le 1.5\}\,, i \in \mathcal{V},
\end{aligned} \tag{19}$$

all of the following set-points from real-time optimization are infeasible for the subsystems: $y_{d,con} = 4.00$, $y_{d,1} = 1.00$, $y_{d,2} = 2.00$, $y_{d,3} = 3.00$, $y_{d,4} = 3.50$,

$y_{d,5} = 4.60$. In this case, all $y^k_{s,c,i}$ in Figure 6(a) achieve consensus and converge to a new point, and all $y^k_{s,n,i}$ in Figure 6(b-f) also converge to their recalculated set-points. All the set-points are recalculated, and the results are $y^*_{s,c,i} = 3.68$, $y^*_{s,n,1} = 1.05$, $y^*_{s,n,2} = 2.03$, $y^*_{s,n,3} = 2.61$ and $y^*_{s,n,4} = 3.19$ and $y^*_{s,n,5} = 4.50$. $(x^*_{s,i}, u^*_{s,i})$ are shown as

$$\begin{aligned}
x^*_{s,1} &= [3.66; 1.05; 0.07], u^*_{s,1} = [-0.23; -1.25],\\
x^*_{s,2} &= [3.56; 2.03; 0.58], u^*_{s,2} = [-0.75; -1.32],\\
x^*_{s,3} &= [3.56; 2.61; 0.61], u^*_{s,3} = [-0.92; -1.50],\\
x^*_{s,4} &= [3.52; 3.19; 0.76], u^*_{s,4} = [-1.13; -1.50],\\
x^*_{s,5} &= [3.51; 4.50; 0.86], u^*_{s,5} = [-1.00; -0.99],
\end{aligned}$$

which are in the state and input constraint sets. This means that the recalculated set-points are feasible and the distributed partial output consensus optimization method is still effective, when all the set-points from real-time optimization are infeasible.

Based on these results, it can be verified that the proposed distributed partial output consensus optimization method can find feasible set-points with partial consensus even when the set-points from real-time optimization are infeasible.

Case 4: Centralized and distributed projection algorithms.

In this paper, the proposed distributed projection algorithm performs well in terms of computational speed. To verify this point, a simulation is carried out using the following two methods:

- The centralized method: The projection algorithm is implemented by solving (13) in a centralized manner. This means that $\hat{y}_{s,c,i}, \hat{y}_{s,n,i}$ for all subsystems are calculated by solving a high-dimensional problem.

- The proposed distributed method: The projection algorithm is implemented by solving (12) in a distributed manner, which means that subsystems calculate their own $\hat{y}_{s,c,i}, \hat{y}_{s,n,i}$ by solving multiple low-dimension problems in parallel.

For $i = 1, 2, \cdots$, $\tilde{y}^k_{s,c,i}$, $\tilde{y}^k_{s,n,i}$ are set as $5 * sin(i\pi/10)$, $5 * cos(i\pi/10)$ respectively and other parameters are the same as that in Case 1. For the sake of fairness, ADMM in Boyd et al. (2011) is chosen as the optimization tool, and the parameters for the two methods are set as the same (the step

Table 1: The computation times of different projection algorithms

Number of	Computation time	
subsystems	The centralized method	The proposed method
5	3.762s	3.586s
10	7.810s	3.770s
30	12.149s	4.015s
50	16.426s	4.167s

size, penalty factor and convergence threshold are set as 0.001, 0.05 and 0.0001 respectively). According to the results in TABLE I, as the number of subsystems increases, the computation time of the centralized method becomes longer than that of the proposed method. In some scenarios, the centralized method cannot be implemented because the computation time is longer than the control period. Therefore, the proposed distributed method can work more efficiently than a centralized approach.

4.2. Experiment

In this subsection, the results of two solution proportioning experiments are shown to validate the effectiveness of the proposed method. The experiments are carried out on a platform in China University of Petroleum (East China), which is shown in Figure 7. In this platform, there are two tanks R-101 and R-102, which are connected by a pipeline from R-102 to R-101 (red line in Figure 8). Here, the two tanks can be considered as subsystems forming a typical chain interconnected system. According to Figure 8, the solution proportioning experiments can be described as follows. The objective is to maintain the NaOH solution so that it has different concentrations and the same temperature in R-101 and R-102. Water and high concentration NaOH solutions are the material and these are transported to R-101 and R-102. The flow rates of the material can be controlled to maintain the NaOH solution at a certain concentration. Hot water is pumped into the jackets of R-101 and R-102 to raise the temperature of the solution. Its flow rate can be controlled to maintain the solution at a certain temperature.

- Material tanks: V-111 has water at about $25°C$. V-112 10.0 $kmol/m^3$

Figure 7: The platform of two tanks Liu et al. (2023).

has NaOH solution at about $25°C$.

- Controlled variables: the temperature T_1, T_2 and concentration C_1, C_2 of R-101 and R-102. Because the temperatures of the solutions in the two tanks need to achieve consensus, T_1, T_2 are modeled as 'Consensus' outputs. The set-points of the concentrations in the two tanks are different, so C_1, C_2 are modeled as 'Nonconsensus' outputs.

- Manipulated variables: the flow rates $v_{s,1}$, $v_{s,2}$ of NaOH solution and the flow rates of hot water $v_{w,1}$, $v_{w,2}$ for R-101 and R-102.

- Constraints: the minimum and maximum values of T_1, T_2 are $20°C$ and $40°C$. The minimum and maximum values of C_1, C_2 are $0kmol/m^3$ and $10kmol/m^3$. Due to the limitations of the valves, the minimum and maximum values of the $v_{s,1}$, $v_{s,2}$, $v_{w,1}$ and $v_{w,2}$ are $0L/h$ and $25.0L/h$.

The set-points from real-time optimization are $C_{d,1} = 5.0kmol/m^3$, $C_{d,2} = 3.5kmol/m^3$ and $T_d = 41.0°C$, where T_d is not feasible. The process can be modelled as a linear 'chain interconnected system' as in (1), where the parameters are

$$A_{11} = \begin{bmatrix} 0.598 & 0.001 \\ -0.002 & 0.599 \end{bmatrix}, A_{22} = \begin{bmatrix} 0.597 & -0.003 \\ 0.001 & 0.598 \end{bmatrix},$$
$$A_{12} = \begin{bmatrix} 0.003 & -0.001 \\ 0 & -0.002 \end{bmatrix}, A_{21} = \begin{bmatrix} -0.002 & 0.001 \\ 0.002 & 0 \end{bmatrix},$$
$$B_1 = \begin{bmatrix} 0.613 & 0.001 \\ 0.005 & 0.458 \end{bmatrix}, B_2 = \begin{bmatrix} 0.904 & 0.002 \\ 0.001 & 0.325 \end{bmatrix},$$

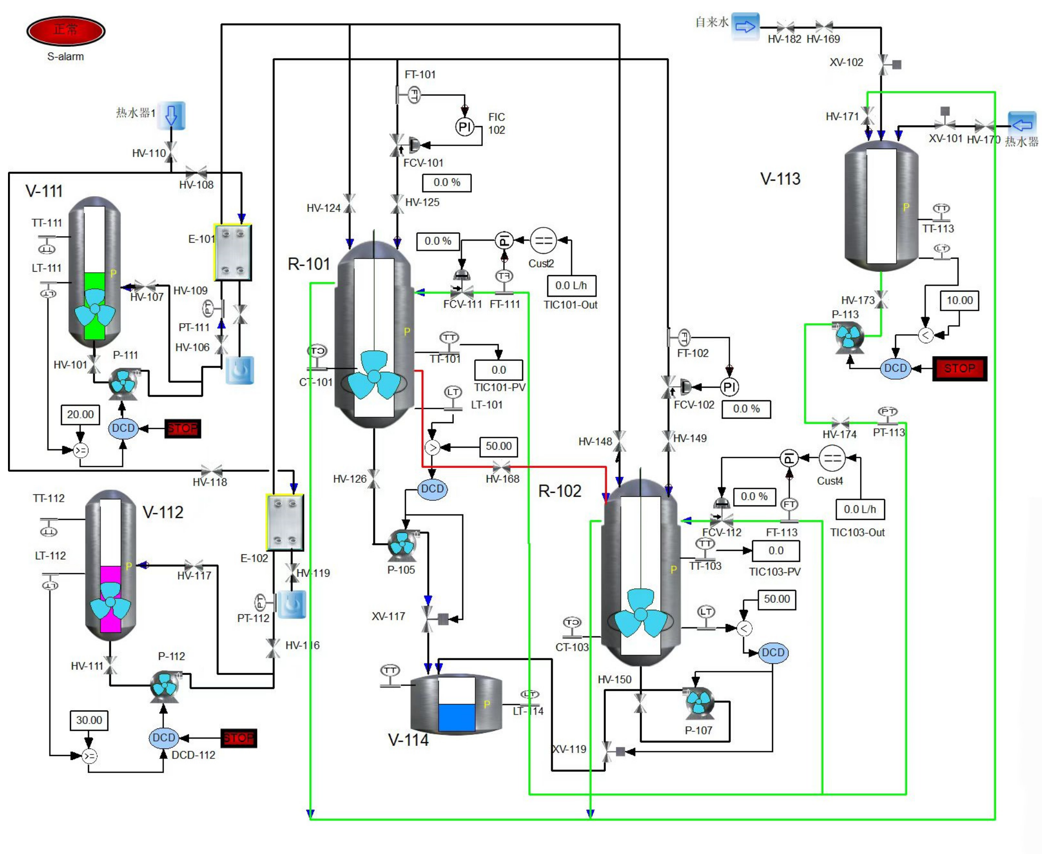


Figure 8: The solution proportioning experiments.

$$C_1 = C_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, T_c = \begin{bmatrix} 1 & 0 \end{bmatrix}, T_n = \begin{bmatrix} 0 & 1 \end{bmatrix}.$$

In this experiment, T_d is not feasible and the proposed distributed partial output consensus optimization method is used to recalculate the feasible set-points. The parameters for the distributed partial output consensus optimization are set as: $\alpha_c = 0.3$, $\alpha_n = 0.1$, $\delta = 0.0001$. The distributed projection algorithm is implemented based on ADMM in Boyd et al. (2011) (its step size, penalty factor and maximum number of iterations are 0.001, 0.05 and 40 respectively). The initial values are $T_{s,1}^0 = T_{s,2}^0 = 30°C$, $C_{s,1}^0 = C_{s,2}^0 = 0kmol/m^3$. In Figure 9, the behaviours of $T_{s,i}^k$, $C_{s,i}^k$ are shown and it is straightforward to see that the recalculated points with partial consensus can be obtained, which are $T_{s,1}^* = T_{s,2}^* = T_s^* = 38.5°C$,

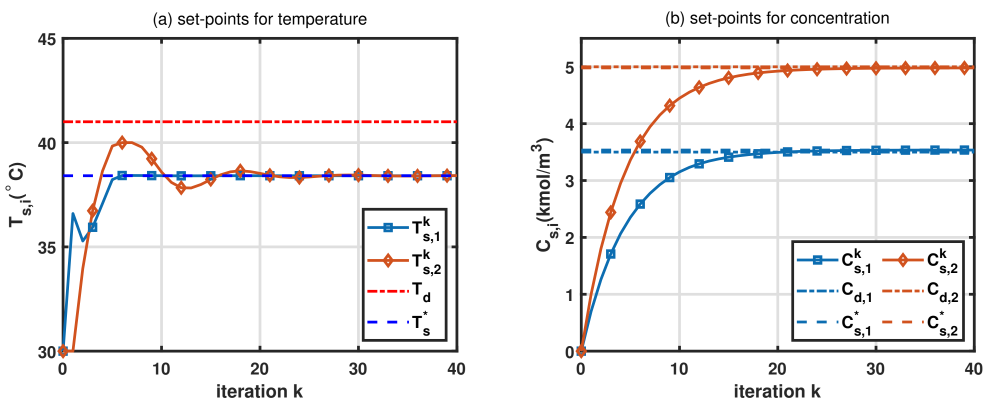


Figure 9: The results of the distributed partial output consensus optimization method.

$C^*_{s,1} = C_{d,1} = 5.0kmol/m^3$ and $C^*_{s,2} = C_{d,2} = 3.5kmol/m^3$. Then, according to the framework in Figure 2, the recalculated points can be utilized by the underlying control layer.

To further verify the effectiveness of the proposed distributed partial output consensus optimization method, the following experimental test conditions are considered:

- Distributed Model Predictive Control (DMPC) is used without deploying the proposed distributed partial output consensus optimization method: DMPC generates an optimal control based on the infeasible set-points generated by real-time optimization.

- DMPC is implemented with the proposed distributed partial output consensus optimization method: DMPC generates the optimal control based on the recalculated set-points obtained by the distributed partial output consensus optimization method.

To ensure objectivity and impartiality, the DMPC parameters used in both cases are identical: prediction horizon N_p is 10, weighting matrices Q_i and R_i are $2{*}I_3$ and I_2, terminal set ratio κ is 0.5. Figure 10 and Figure 11 show the trends of the temperatures and concentrations with the different control methods, where the horizontal coordinate is the time t, and the vertical coordinate is the value of T_i or C_i. Chain lines represent the set-points generated by real-time optimization, solid lines with square and triangle marks represent the trends of T_i, C_i for R-101 and R-102 respectively

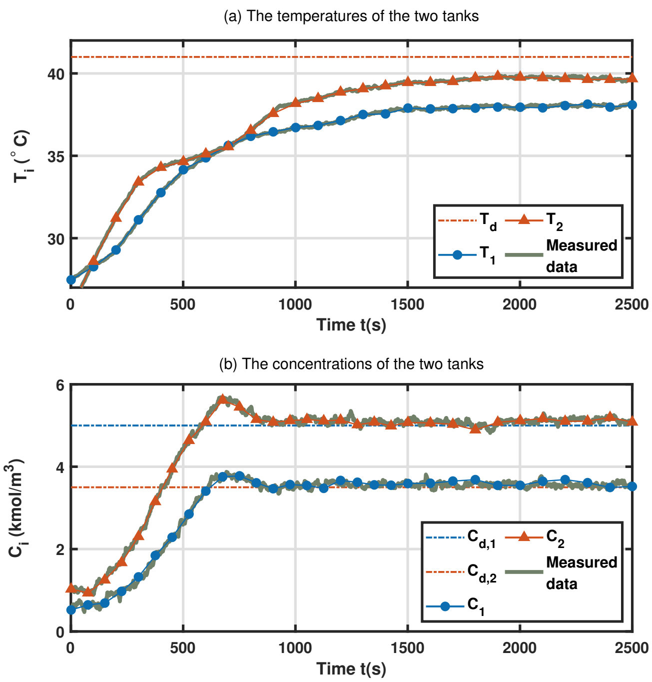


Figure 10: The results of DMPC without employing distributed partial output consensus optimization

and gray lines represent measured T_i, C_i. In addition, dashed lines represent the set-points recalculated using distributed partial output consensus optimization, which only appear in Figure 11.

Figure 10 shows the results of DMPC without distributed partial output consensus optimization. Although C_1 and C_2 in Figure 10(b) can reach $C_{d,1}$ and $C_{d,2}$ finally, T_1 and T_2 in Figure 10(a) cannot achieve consensus and converge to $39.5°C$ and $38.5°C$ respectively. Figure 11 shows the results of DMPC with distributed partial output consensus optimization. T_1, T_2 in Figure 11(a) can achieve consensus and reach the recalculated 'Consensus' set-point ($38.5°C$), and C_1, C_2 in Figure 11(b) can reach $3.5kmol/m^3$ and $5.0kmol/m^3$. DMPC without distributed partial output consensus optimization cannot achieve the objective of this experiment due to the failure to guarantee the consensus of T_1 and T_2. Based on the proposed distributed partial output consensus optimization, DMPC can ensure that subsystems achieve partial consensus, that is, T_1, T_2 reach a consensus value and C_1, C_2

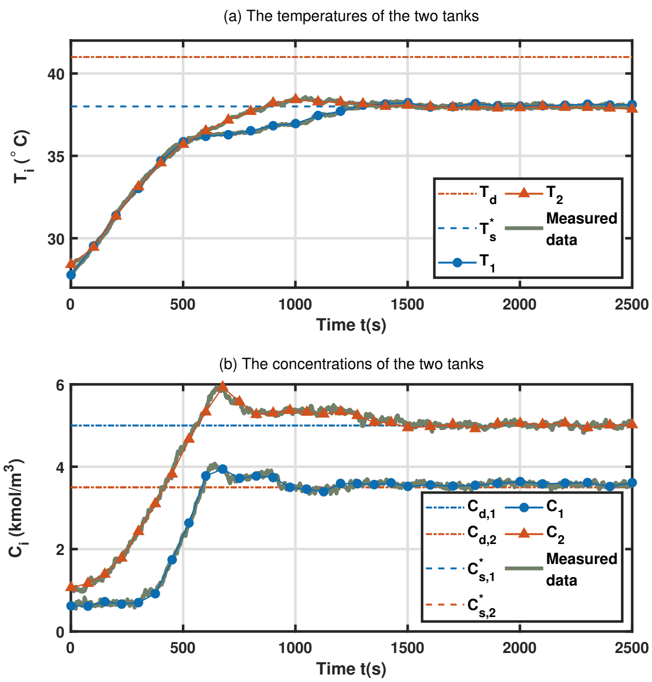


Figure 11: The results of DMPC with distributed partial output consensus optimization

reach their own set-points. It can be concluded that when the set-points from real-time optimization are infeasible, the proposed distributed partial output consensus optimization approach can not only handle the partial consensus requirement, but also recalculate the feasible set-points for the controller. Based on these experimental results, the effectiveness of the proposed distributed partial output consensus optimization approach is verified.

5. Conclusion

In this paper, when the set-points from real-time optimization are infeasible, partial output consensus optimization is investigated for chain interconnected systems and a distributed method is developed. The proposed algorithm can recalculate the feasible set-points with partial consensus to replace the infeasible ones obtained from real-time optimization. Firstly, the POC feasible set-point optimal problem is established, which contains the requirements of partial consensus and feasibility. Then, an iterative strategy is designed to calculate the ‘Consensus’ and ‘Nonconsensus’ set-points simultaneously. In this strategy, a distributed projection algorithm is proposed to guarantee the feasibility of the recalculated set-points. It is a further advantage that the approach has a low computational burden. The convergence of distributed partial output consensus optimization is analyzed. Finally, according to the results of both simulations and experiments, distributed partial output consensus optimization is effective in the case of single and multiple infeasible set-points.

6. Acknowledgments

This work is partially supported by the National Nature Science Foundation of China (Grant No 61973315) and the Science and Technology Program of Qingdao, China (Grant No 22-3-7-cspz-15-nsh).

References

Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J., et al., 2011. Distributed optimization and statistical learning via the alternating direction method of multipliers. Foundations and Trends in Machine Learning 3, 1–122.

Ding, B., 2017. Robust and adaptive model predictive control of nonlinear systems. IEEE Control Systems Magazine 37, 125–127.

Falsone, A., Notarnicola, I., Notarstefano, G., Prandini, M., 2020. Tracking-admm for distributed constraint-coupled optimization. Automatica 117, 108962.

Farina, M., Betti, G., Giulioni, L., Scattolini, R., 2014. An approach to distributed predictive control for tracking-theory and applications. IEEE Transactions on Control Systems Technology 22, 1558–1566.

Fontan, A., Shi, G., Hu, X., Altafini, C., 2020. Interval consensus for multi-agent networks. IEEE Transactions on Automatic Control 65, 1855–1869.

Li, S., Zheng, Y., Wang, B., 2008. Steady-state target calculation for constrained predictive control systems based on goal programming. Asia-Pacific Journal of Chemical Engineering 3, 648–655.

Liu, H., Li, S., Chai, T., 2003. Intelligent decoupling control of power plant main steam pressure and power output. International Journal of Electrical Power and Energy Systems 25, 809–819.

Liu, Q., Wang, J., 2013. A one-layer projection neural network for nonsmooth optimization subject to linear equalities and bound constraints. IEEE Transactions on Neural Networks and Learning Systems 24, 812–824.

Liu, Q., Yang, S., Hong, Y., 2017. Constrained consensus algorithms with fixed step size for distributed convex optimization over multiagent networks. IEEE Transactions on Automatic Control 62, 4259–4265.

Liu, Z., Zhao, D., Feng, J., Zhang, Y., Spurgeon, S.K., 2023. Distributed robust partial state consensus control for chain interconnected delay systems. IEEE Transactions on Control Systems Technology 31, 977–984.

Marchetti, A., Ferramosca, A., González, A., 2014. Steady-state target optimization designs for integrating real-time optimization and model predictive control. Journal of Process Control 24, 129–145.

Meng, Z., Yang, T., Shi, G., Dimarogonas, D.V., Hong, Y., Johansson, K.H., 2017. Targeted agreement of multiple lagrangian systems. Automatica 84, 109–116.

Pang, Q., Zou, T., Zhang, Y., Cong, Q., 2015. A steady-state target calculation method based on "point" model for integrating processes. ISA Transactions 56, 196–205.

René, S., Predrag, M., Dominique, B., 2019. Distributed modifier-adaptation schemes for the real-time optimisation of uncertain interconnected systems. International Journal of Control 92, 1123–1136.

Turan, M.S., Xu, L., Ferrari-Trecate, G., 2021. On consensusability of linear interconnected multiagent systems and simultaneous stabilization. IEEE Transactions on Control of Network Systems 9, 403–414.

Wu, Q., Du, W., Nagy, Z., 2021. Steady-state target calculation integrating economic optimization for constrained model predictive control. Computers and Chemical Engineering 145, 107145.

Xia, Z., Liu, Y., Qiu, J., Ruan, Q., Cao, J., 2023. An rnn-based algorithm for decentralized-partial-consensus constrained optimization. IEEE Transactions on Neural Networks and Learning Systems 34, 534–542.

Yan, J., Guo, F., Wen, C., Li, G., 2020. Parallel alternating direction method of multipliers. Information Sciences 507, 185–196.

Yang, T., Wan, Y., Wang, H., Lin, Z., 2018. Global optimal consensus for discrete-time multi-agent systems with bounded controls. Automatica 97, 182–185.

Yang, X., He, H., Zhong, X., 2021. Approximate dynamic programming for nonlinear-constrained optimizations.

Yang, Y., Ding, B., 2020. Two-layer model predictive control for chemical process model with integrating controlled variables. The Canadian Journal of Chemical Engineering 98, 237–253.

Yangzhou, C., Wei, L., Guiping, D., Aleksandrov, A.Y., 2015. Consensus analysis and design of linear interconnected multi-agent systems. Acta Mathematica Scientia 35, 1305–1317.

Yoshitani, N., 1993. Modelling and parameter estimation for strip temperature control in continuous annealing processes, in: Proceedings of IECON '93 - 19th Annual Conference of IEEE Industrial Electronics, pp. 469–474.

Zhang, S., Zhao, D., Spurgeon, S.K., 2022. Distributed static output feedback robust model predictive control for process networks. Control Engineering Practice 126, 105239.