Renewable Energy 216 (2023) 118978

Contents lists available at ScienceDirect

Renewable Energy

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

Clustering-based model predictive control of solar parabolic trough plants ☆

Paula Chanfreut [a,b,*], José M. Maestre [a], Antonio Gallego [a], Anuradha M. Annaswamy [c], Eduardo F. Camacho [a]

[a] Department of Systems and Automation Engineering, University of Seville, Spain
[b] Department of Mechanical Engineering, Eindhoven University of Technology, The Netherlands
[c] Department of Mechanical Engineering, Massachusetts Institute of Technology, USA

ARTICLE INFO

Keywords:
Model predictive control
Solar thermal power plants
Parabolic trough collectors
Control by clustering
Coalitional control
Hierarchical control

ABSTRACT

This paper presents a clustering-based model predictive controller for optimizing the heat transfer fluid (HTF) flow rates circulating through every loop in solar parabolic trough plants. In particular, we present a hierarchical approach consisting of two layers: a bottom layer, composed of a set of model predictive control (MPC) agents; and a top layer, which dynamically partitions the set of loops into clusters. Likewise, the top layer allocates a certain share of the total available HTF to each cluster, which is then distributed among the loops by the bottom layer in response to the varying conditions of the solar field, e.g., to deal with passing clouds. The dynamic clustering of the system reduces the number of variables to be coordinated in comparison with centralized MPC, thereby speeding up the computations. Moreover, the loops efficiencies and the heat losses coefficients, which influence the loops control model, are also estimated at the bottom layer. Numerical results on a 10-loop and an 80-loop plant are provided.

1. Introduction

Renewable energy generation plays a central role in the path towards a more sustainable and resilient energy system. The benefits of renewable energies go beyond lowering greenhouse gas emissions and also include the diversification of the energy supply and the reduction of the dependency on fossil fuels. In 2021, the share of renewables in the global electricity generation reached about a 28% and is expected to rise until 38% by 2027 according to the International Energy Agency (IEA) [1]. In particular, wind and solar resources reached a 10% of the global electricity mix as reported by the Renewable Energy Policy Network for the 21st Century (REN21) [2].

The radiant energy per unit area that the Sun provides at the top of the Earth's atmosphere is roughly 1361 W/m^2 [3]. In this regard, according to the World Energy Council (WEC) [4], the annual solar radiation reaching the Earth's surface is about 7500 times the world's annual energy consumption, which illustrates its potential. In recent years, the advances and investment on photovoltaic technology have brought a huge growth of solar energy generation [1], yet there are more 6GW of installed capacity of concentrated solar thermal (CST) power technology [2]. Unlike photovoltaics, which directly convert the sunlight into electricity, CST technologies use mirrors to focus the solar rays on a receiver to heat a working fluid, commonly referred to as heat transfer fluid (HTF) [5]. There are four main types of CST systems: parabolic troughs, linear Fresnel reflectors, power towers, and parabolic dishes [6,7]. While the first two use a linear receiver tube [8], the latter two concentrate the solar rays onto a point. The thermal energy gained by the HTF is then converted into mechanical energy using a turbine to produce electricity. Moreover, CST plants can incorporate thermal storage systems, which provides significant advantages to deal with intermittent and variable nature of the solar radiation [9–12].

This article is focused on solar parabolic trough plants [13–15], which represents the most widespread CST technology deployed up to date, with Spain and the United States holding the largest share of the installed capacity [2,16]. The solar field in this type of plants consists of a set of arrays of parabolic collectors, i.e., *loops*, through which circulates the HTF, which is typically a synthetic oil or a molten salt [17]. To control the HTF temperature around a given setpoint, its flow rate must be dynamically manipulated taking into account the plant dynamics and possible fluctuations of the received solar irradiance. Model predictive control (MPC) has provided positive results to address this problem as reviewed by Camacho et al. [18] (Chapter 5)

☆ This work is supported by the European Research Council Advanced Grant OCONTSOLAR under Grant SI-1838/24/2018, and by the Spanish MCIN/AEI/10.13039/501100011033 Project C3PO-R2D2 under Grant PID2020-119476RB-I00.

* Correspondence to: Department Mechanical Engineering, Gemini Building number 15, Groene Loper, 5612 AE Eindhoven, The Netherlands.
E-mail addresses: pchanfreut@us.es, p.chanfreut.palacio@tue.nl (P. Chanfreut), pepemaestre@us.es (J.M. Maestre), agallego2@us.es (A. Gallego), aanna@mit.edu (A.M. Annaswamy), efcamacho@us.es (E.F. Camacho).

https://doi.org/10.1016/j.renene.2023.118978
Received 23 February 2023; Received in revised form 12 June 2023; Accepted 29 June 2023
Available online 10 July 2023

and illustrated, for example, by Andrade et al. [19], Navas et al. [20], López-Bautista et al. [21] and Gholaminejad and Khaki-Sedigh [22].

MPC is an advanced control method based on iterative computations of the sequence of control inputs that optimizes the system performance along a certain finite time horizon [23,24]. Its inherent optimal nature, and ability to handle complex systems with constraints, are some of the advantages that have made it increasingly popular over the last decades [25,26]. More recently, distributed implementations have allowed scaling the benefits of MPC to large-scale systems, where a centralized control is not viable or practical for real time control [27].

The essence of distributed MPC (DMPC) is to tackle the overall control problem by using multiple local MPC agents that exchange information to coordinate their decisions. Unlike decentralized MPC, DMPC allows for cooperation among agents to improve performance [28], but this comes at the cost of increased communication and computational resources, which may not be desirable or efficient. Motivated by this, a number of recent works have explored distributed approaches with partial and dynamic inter-agent cooperation as surveyed by Chanfreut et al. [29]. The underlying idea to boost the scalability of distributed architectures is often to use time-varying communication topologies, e.g., [30,31]. One of the strategies proposed in this framework is the so-called *coalitional* control [32], an approach that is characterized by the dynamic formation of *clusters* of cooperative agents, referred to as *coalitions*.

In this context, the project *Optimal Control of Solar Energy Systems* (OCONTSOLAR), which has been funded by the European Research Council [33], develops coalitional MPC algorithms to optimize the HTF flow rates in solar parabolic trough plants. Conversely to the classic operation of these plants, the flow in different loops are considered to be manipulable in real-time by using valves. The latter is of special interest when the conditions in the solar field are such that the capacity to warm up the HTF is uneven among the loops, e.g., due to cloud shading [34,35]. Indeed, by using the valves, it is possible to obtain a better thermal balance of the HTF temperature in each loop, which helps to minimize defocusing actions and hence energy losses [36]. However, optimizing the flows in every loop involves dealing with a large set of optimization variables that would also scale with the length of the prediction horizon. The use of a centralized MPC lacks of redundancy and imposes a substantial burden on computation and communication, which motivates the study of noncentralized MPC strategies and, in particular, the mentioned coalitional approaches. In addition, for the flows control to be effective, MPC-based schemes require an accurate model of the solar field. The loops dynamics are though characterized by parameters that are uncertain, difficult to measure, and which may also vary in time, such as their optical efficiencies and heat losses coefficients.

The main contribution of this article is a hierarchical cluster-based MPC for optimizing the loops flow rates to track reference outlet temperatures. The proposed approach integrates a bottom layer, comprising a set of MPC agents that control the valves of each loop; and a top layer, which acts as a supervisor. The bottom layer uses approaches based on coalitional MPC algorithms and parameter identification using recursive least squares. On the other hand, the supervisor at the top layer partitions the set of loops in a dynamic manner into clusters, and assigns to these clusters a maximum HTF flow considering the plant constraints. To optimize these maximum flows, a clusters-based lumped description of the overall solar field is considered, which provides a simplified representation of the overall system. Our results on a 10-loop and a 80-loop plant show that the proposed approach can closely approximate the centralized performance while reducing both the computation times and the need for centralized computations.

The rest of the article is organized as follows. Section 2 presents the system dynamics and constraints, and introduces the control objectives. Section 3 describes the architecture of the proposed controller, details the elements comprising the bottom and top layers, and provides a pseudo-code with the control algorithm. Finally, our simulation results and concluding remarks are respectively presented in Sections 4 and 5.

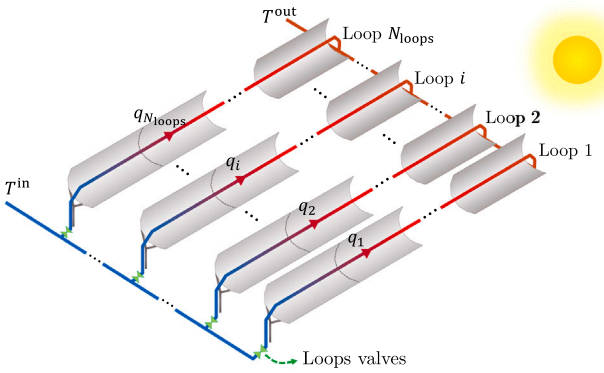


Fig. 1. Scheme of the solar field. The loops of parabolic collectors are equipped with valves that allow controlling the HTF flow rates that pass through them.

2. Problem formulation

Consider a solar parabolic trough plant comprising a set of parallel loops $\mathcal{N} = \{1, 2, \dots, N_{\text{loops}}\}$ as shown in Fig. 1. In this respect, throughout this paper we will consider that the HTF flow that is pumped to different loops can be controlled in real-time by using valves.

2.1. System dynamics and constraints

Considering the variation of the internal energy of the fluid, the dynamics of the output temperature of a given loop $i \in \mathcal{N}$, i.e., T_i^{out} [°C], can be described by the following concentrated parameter model[1]:

$$C_i \frac{dT_i^{\text{out}}}{dt} = \eta_i \mathcal{I}_i - q_i P_i (T_i^{\text{out}} - T^{\text{in}}) - \hbar_i, \tag{1}$$

where T^{in} [°C] is the inlet temperature, q_i [m^3/s] represents the HTF that is pumped to loop i, C_i [J/°C] is the thermal capacity of the loop, P_i [J/(m^3 °C)] is related to its geometrical and thermal properties, and $\hbar_i$ [W] is a function weighting the heat losses of loop i. Also, $\mathcal{I}_i = S_{\text{loop}} I_i$, where S_{loop} [m^2] denotes the loops' reflective surface and I_i [W/m^2] is the direct normal irradiance (DNI) received by loop i. In addition, η_i is the loop efficiency, which considers the optical and geometric efficiency of the collectors in i. Finally, some of the variables abovementioned are defined as a function of the temperature as follows:

$$\begin{aligned} &\rho_i = 903 - 0.672 T_i^{\text{m}}, \quad c_i = 1820 + 3.478 T_i^{\text{m}}, \quad P_i = \rho_i c_i, \quad C_i = \rho_i c_i A L, \\ &\hbar_i = S_{\text{loop}} \left(h_{2,i} (T_i^{\text{m}} - T^{\text{a}})^2 + h_{1,i} (T_i^{\text{m}} - T^{\text{a}}) \right), \qquad T_i^{\text{m}} = (T_i^{\text{out}} + T^{\text{in}})/2, \end{aligned} \tag{2}$$

where T^{a} [°C] is the ambient temperature, A [m^2] is the cross sectional area of the tube, L [m] is the loops length, and coefficients $h_{1,i}$ and $h_{2,i}$ will be estimated as discussed in Section 3.1.2. Note that (2) considers the properties of HTF Therminol 55, which was used in the ACUREX solar plant of the *Plataforma Solar de Almería* [18, Chapter 4].

Regarding the system constraints, hereafter let Q_{T} denote the maximum available flow rate that can be used in the overall plant, $q_{\min}$ the minimum flow rate allowed in the loops, and $q_{\max}$ the corresponding maximum value. Then, it must be guaranteed that

$$\sum_{i \in \mathcal{N}} q_i \leq Q_{\text{T}}, \quad \text{and} \quad q^{\min} \leq q_i \leq q^{\max}, \ \forall i \in \mathcal{N}. \tag{3}$$

Moreover, the outlet temperature is desired to evolve in range [$T^{\min}$, $T^{\max}$] for all $i \in \mathcal{N}$, where $T^{\min}$ and $T^{\max}$ denote respectively the minimum and maximum temperatures.

[1] For the sake of clarity, the time index is omitted in this section.

2.1.1. Cluster-based model

As will be discussed in Section 3.2, the set of loops $\mathcal{N}$ will be dynamically partitioned into a set of N_{cl} non-overlapping clusters, i.e., $\{C_1, C_2, \dots, C_{N_{\text{cl}}}\}$, such that

$$\bigcup_{j=1}^{N_{\text{cl}}} C_j = \mathcal{N}, \quad \text{and} \quad C_j \cap C_l = \emptyset \quad \forall j, l \in [1, N_{\text{cl}}],\ j \neq l. \tag{4}$$

Similarly to (1), the dynamics of any cluster C_j can be modeled as:

$$C_{C_j} \frac{dT^{\text{out}}_{C_j}}{dt} = \eta_{C_j} \mathcal{I}_{C_j} - q_{C_j} P_{C_j} (T^{\text{out}}_{C_j} - T^{\text{in}}) - \hbar_{C_j}, \tag{5}$$

where $T^{\text{out}}_{C_j}$ is the outlet temperature of cluster j, q_{C_j} denotes the total HTF pumped to the loops in the cluster, and C_{C_j}, P_{C_j} and $\hbar_{C_j}$ are defined analogously to (2). In addition, $\eta_{C_j} \mathcal{I}_{C_j} = \sum_{i \in C_j} \eta_i \mathcal{I}_i$ represents the effective irradiance in cluster j.

2.2. Control objectives

In this paper, each loop $i \in \mathcal{N}$ will be treated as a subsystem controlled by an MPC agent that can manipulate its inlet valve, thus controlling flow q_i. In turn, the set of MPC agents can communicate to coordinate their decisions to some extent. In this framework, the main control goals and premises considered can be summarized as follows:

- The overall system should track piecewise constant references on the loops outlet temperature. In this respect, the total available HTF can be unevenly distributed among the set of loops to exploit the time- and space-varying conditions of the solar field.
- Centralized and fully coordinated control of all valves should be avoided due to the high computation and communication burden. In this regard, the system performance should approximate that of the centralized controller but reducing the need for centralized computations.
- Parameter η_i and the heat losses coefficients, i.e., $h_{1,i}$ and $h_{2,i}$, must be estimated for all loops $i \in \mathcal{N}$. The efficiency η_i is assumed to be slowly time-varying, and $h_{1,i}$ and $h_{2,i}$ can be considered constants.

Regarding the outlet temperature reference, hereafter T^{ref}, our controller will be designed to be integrated with a setpoint optimization layer as described in Camacho and Gallego [37]. That is, we assume that a different control layer will dynamically provide us with the optimal T^{ref} to maximize the net electricity production or satisfy a given scheduled electrical production.

Remark 1. Parameter η_i considers the geometric and optical efficiency of the collectors in loop i. The geometric efficiency depends on the position of the collectors with respect to the radiation beam vector, whereas the optical efficiency takes into account aspects as the mirrors reflectivity, the interception factor, and the absorptance of the tube [38, 39]. This efficiency can vary over time due to several reasons, e.g., dust accumulation on the mirrors or deterioration of the selective absorbing coating. Conversely to the heat losses coefficients, which change due to the tubes degradation and vacuum losses on a yearly time scale, changes in the loops efficiencies can be observed over the days.

3. Proposed controller architecture

Considering the objectives in Section 2.2, this article proposes the hierarchical approach that is illustrated in Fig. 2. As can be seen, it consists of a top and a bottom layer comprising the following elements:

(i) A set of local agents associated to the loops, which optimize the loops flow q_i, and identify the unknown parameters η_i, $h_{1,i}$, and $h_{2,i}$, for all $i \in \mathcal{N}$.

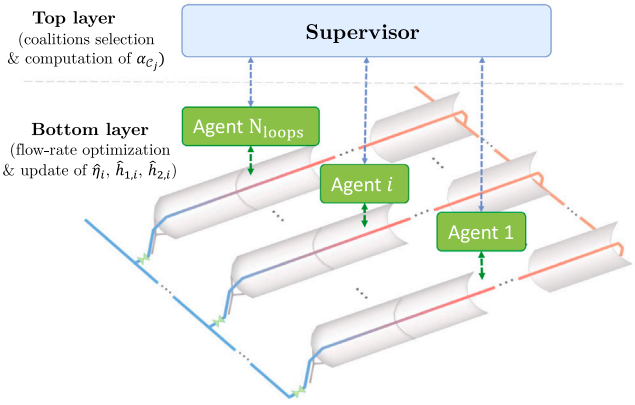


Fig. 2. Hierarchical architecture of the proposed approach. At the bottom layer, a set of agents control the HTF flow rates in each loop, whereas the top layer adjusts their level of coordination and constraints considering (3).

(ii) A supervisor, which partitions dynamically the set of loops into clusters, and updates the flow constraints using the corresponding cluster-based description of the field.

In particular, given a partition of the set of loops, say $\{C_1, C_2, \dots, C_{N_{\text{cl}}}\}$, the agents associated with each cluster C_j will share data to coordinate their decisions on flow rates q_i for all $i \in C_j$. As introduced before, these groups of cooperative agents will be also denote as *coalitions*. That is, coordination is limited to subsets of agents, thus providing a half-way solution between a centralized and complete decentralized controllers. The optimization within each coalition will consider the constraints imposed by the supervisor, which will set a maximum on the HTF that they can use. Additionally, the proposed approach considers different time scales. For simulation purposes, we use a discrete-time version of model (1) for all $i \in \mathcal{N}$ with an integration step size of Δt^{s} seconds. In this regard, in what follows we use k as the discrete time index, i.e., time step k refers to instant $k\Delta t^{\text{s}}$. On the other hand, the control step size to update the flow rates is $\Delta t^{\text{c}} = \delta^{\text{c}} \Delta t^{\text{s}}$ seconds, where $\delta^{\text{c}} \in \mathbb{N}^+$. For the sake of simplicity, we will consider that the sampling time of the model used by the agents and supervisor is also Δt^{c}.

The following subsections provide further details on the function and design of the different elements involved in the presented approach.

3.1. Bottom layer

The bottom layer determines dynamically the HTF flow rates that circulate through the loops and identifies the unknown parameters.

3.1.1. HTF flow optimization

Considering (1), the dynamics of each loop $i \in \mathcal{N}$ will be predicted using the following discrete-time model:

$$\begin{aligned} T^{\text{out}}_i(n + \delta^{\text{c}}|k) = T^{\text{out}}_i(n|k) + \gamma_i(n|k) \big(& \hat{\eta}_i(k) \mathcal{I}_i(k) \\ & - q_i(n|k) P_i(n|k) \left(T^{\text{out}}_i(n|k) - T^{\text{in}}(k) \right) - \hat{\hbar}_i(n|k) \big), \end{aligned} \tag{6}$$

where the notation $(n|k)$ indicates the prediction for time step $n \geq k$ made at time k, and $\gamma_i(n|k) = \Delta t^{\text{c}} / C_i(n|k)$. In addition, $P_i(n|k)$ and $C_i(n|k)$ are computed according to (2) considering the predicted mean temperature, i.e., $T^{\text{m}}_i(n|k) = (T^{\text{out}}_i(n|k) + T^{\text{in}}(k))/2$, and the current ambient temperature $T^{\text{a}}(k)$. Finally, $\hat{\eta}_i(k)$ is the estimated efficiency of loop i at k, and $\hat{\hbar}_i(n|k) = S_{\text{loop}} (\hat{h}_{2,i}(k)(T^{\text{m}}_i(n|k) - T^{\text{a}}(k))^2 + \hat{h}_{1,i}(k)(T^{\text{m}}_i(n|k) - T^{\text{a}}(k)))$, with $\hat{h}_{1,i}(k)$ and $\hat{h}_{2,i}(k)$ being the coefficients estimation available at k.

Consider model (6) and take some time step k such that $k = \kappa \Delta t^{\text{c}}$, where $\kappa \in \mathbb{N}^+$, i.e., the flow rates are updated at step k. Also, let $\mathcal{P}(k)$ be the agents partition selected at k, and C_j be the jth cluster in $\mathcal{P}(k)$.

Finally, let us define the prediction horizon used by the MPC as an interval of $\delta^c N_p$ integration steps ahead, and consider set $\mathcal{H} = \{k, k+\delta^c, k+2\delta^c, \dots, k+\delta^c N_p\}$. Then, at time k, coalition C_j solves the following MPC problem:

$$\min_{[q_i]_{i\in C_j}} \sum_{i\in C_j}\sum_{n\in\mathcal{H}} \left(w_e e_i^2(n+\delta^c|k) + w_q \Delta q_i^2(n|k)\right)$$

s.t. Loops prediction model (6),

$$T_i^{out}(k|k) = T_i^{out}(k), \tag{7a}$$

$$T^{min} \le T_i^{out}(n|k) \le T^{max}, \tag{7b}$$

$$q^{min} \le q_i(n|k) \le q^{max}, \tag{7c}$$

$$\sum_{i\in C_j} q_i(n|k) \le \alpha_{C_j}(n) Q_T, \tag{7d}$$

$$\forall i \in C_j,\ \forall n \in \mathcal{H}, \tag{7e}$$

where w_e and w_q are positive definite weighting scalars, $\alpha_j(\cdot)$ is a scale factor that will be set by the top layer, and

$$\begin{aligned} e_i(n+\delta^c|k) &= T_i^{out}(n+\delta^c|k) - T^{ref}(n+\delta^c), \\ \Delta q_i(n+\delta^c|k) &= q_i(n+\delta^c|k) - q_i(n|k), \quad \forall n \in \mathcal{H}, \end{aligned} \tag{8}$$

Notice that at the initial time instant

$$\Delta q_i(k|k) = q_i(k|k) - q_i(k-\delta^c),$$

where $q_i(k-\delta^c)$ is known and equal to $q_i(k-1)$ since the control inputs are updated each δ^c steps.

Remark 2. In problem (7), prediction model (6) considers as simplification that temperatures T^{in} and T^a maintain their value at k during the prediction horizon. Similarly, the effective irradiance is defined as the product of the last updated estimated efficiency and the DNI at k, which is assumed to be known.

3.1.2. Parameters identification

As introduced in Remark 1, the loops efficiencies and heat losses coefficients are unknown parameters that can vary in time for multiple reasons, e.g. dust accumulation and vacuum losses. To estimate them, this article considers a local system identifier based on the recursive least-square method with exponential forgetting [40] that is designed as follows.

Let us consider continuous-time model (1), which can be rewritten as

$$\frac{dT_i^{out}}{dt} = \dot{T}_i^{out} = \varphi_i + \phi_i^{\mathcal{I}} \eta_i + \phi_i^{\hbar} [h_{2,i},\ h_{1,i}]^T, \tag{9}$$

where $\varphi_i = -q_i P_i (T_i^{out} - T^{in})/C_i$, $\phi_i^{\mathcal{I}} = \mathcal{I}_i/C_i$, and $\phi_i^{\hbar} = -S_{loop}/C_i [(T_i^m - T^a)^2,\ (T_i^m - T^a)]$. Also, note that parameters η_i, $h_{1,i}$ and $h_{2,i}$ may not be uniquely identifiable and, moreover, they change in different time scales (see Remark 1). To deal with this issue, we consider that $h_{1,i}$ and $h_{2,i}$ will only be estimated by using data collected in periods of low irradiance, e.g., during defocusing actions or at the end of the day. In that case, $\phi_i^{\mathcal{I}} \approx 0$, and hence η_i does not affect the temperature evolution. Once we have an estimation of $h_{1,i}$ and $h_{2,i}$, they will be treated as *known* constant parameters, and we will only adapt η_i during the periods of higher irradiance.

For the estimation of $h_{1,i}$ and $h_{2,i}$, consider a filtered version of the variables in (9), i.e., $\dot{T}_{i,f}^{out} = \lambda(T_i^{out} - T_{i,f}^{out})$, $\phi_{i,f}^{\hbar} = \lambda(\phi_i^{\hbar} - \phi_{i,f}^{\hbar})$ and $\dot{\varphi}_{i,f} = \lambda(\varphi_i - \varphi_{i,f})$, where $\lambda \ge 0$ is the filter inverse constant, and sub-index 'f' holds for filtered [41]. Then, we can transform (9) into:

$$z_i = \lambda(T_i^{out} - T_{i,f}^{out}) - \varphi_{i,f} = \phi_{i,f}^{\hbar} [h_{2,i},\ h_{1,i}]^T. \tag{10}$$

Considering this, we will use the following least squares adaptive law with exponential forgetting for adjusting these coefficients:

$$\begin{bmatrix} \dot{\hat{h}}_{2,i} \\ \dot{\hat{h}}_{1,i} \end{bmatrix} = \Gamma_i \phi_{i,f}^{\hbar} \left(z_i - \phi_{i,f}^{\hbar} \begin{bmatrix} \hat{h}_{2,i} \\ \hat{h}_{1,i} \end{bmatrix} \right), \qquad \begin{bmatrix} \hat{h}_{2,i}(0) \\ \hat{h}_{1,i}(0) \end{bmatrix} = \hat{h}_0, \tag{11}$$

where $\hat{h}_0$ is the initial estimation, and

$$\dot{\Gamma}_i = \beta \Gamma_i - \Gamma_i \phi_{i,f}^{\hbar} (\phi_{i,f}^{\hbar})^T \Gamma_i, \qquad \Gamma_i(0) = \Gamma_0, \tag{12}$$

with $\beta \ge 0$ being the continuous time forgetting factor, and $\Gamma_0 > 0$. Notice that Eqs. (11) and (12) can be discretized providing a discrete-time adaptation law that does not require a discretized measure of derivative $\dot{T}_i^{out}$ but only samples of T_i^{out}, i.e., it depends on variables that can be directly measured in the system. In this regard, the least-squares method with exponential forgetting has been chosen for the simplicity of its implementation and its convergence guarantees of the parameters error under persistently exciting regression vectors.

Finally, for the adaptation of η_i along the days, we will proceed analogously and fix the heat losses coefficients to their last estimated value. Note that then (9) can be rewritten in the form of $\dot{T}_i^{out} = \tilde{\varphi}_i + \phi_i^{\mathcal{I}} \eta_i$, where $\tilde{\varphi}_i$ is treated as a known variable.

3.2. Top layer

The top layer sets periodically a partition into coalitions and distributes the total available flow among them for its optimization at the bottom layer.

3.2.1. Clusters formation

The set of agents will be dynamically clustered into coalitions such that the dynamics of the loops controlled by the same coalition are similar. Therefore, the elements in the same cluster are approximately characterized by the same lumped parameters. To this end, we build up the following matrix at all control time steps k:

$$\mathbf{P}(k) = \left[\hat{\eta}_i(k)\mathcal{I}_i(k),\quad T_i^m(k),\quad \hat{\hbar}_i(k)\right]_{i\in\mathcal{N}}. \tag{13}$$

Note that, given (2), variable $T_i^m(k)$ characterizes parameters $C_i(k)$ and $P_i(k)$ that also appear in the loops model.

Using matrix $\mathbf{P}(k)$, we can find a partition of the set of loops in the solar field using clustering methods [42]. Without loss of generality, we consider the well-known K-means algorithm [43], which is an iterative centroid-based algorithm that provides a partition of a data set into K clusters. In particular, we will use Matlab® function `kmeans` to find a partition of the N_{loops} entries in $\mathbf{P}(k)$ into N_{cl} clusters.[2]

The number of clusters N_{cl} can be selected following the *elbow* method, which runs the K-means clustering algorithm for a range of values for K, and for each of them computes an average score for the resulting clusters. The range of K to be tested is defined as $[2, N_{cl}^{max}]$, where N_{cl}^{max} denotes the maximum number of clusters. As for the score, different criteria can be used [44], being the Calinski-Harabasz index [45] the one considered in our simulations, which measures the between-cluster and within-cluster variance so as to find dense and well separated clusters.

Remark 3. Setting N_{cl}^{max} close to N_{loops} generates smaller coalitions, and hence sparser communication topologies, but will increase the number of decision variables of the supervisor (presented in Section 3.2.2). For this reason, N_{cl}^{max} should be set so that the number of optimization variables handled by the coalitions and those of the supervisor are significantly fewer than those corresponding to the centralized control problem. Also, smaller N_{cl}^{max} allows the real-time implementation of the *elbow* method to find the optimal partition.

[2] Note that N_{cl} corresponds to the value of K used when running the K-means clustering algorithm.

3.2.2. Clusters-based flow allocation

The supervisor will consider a clusters-based model of the overall solar field to find a *candidate* distribution of the HTF among the coalitions, which will then be used to set the constraints for the bottom layer. The model used at the top layer describes the solar field as an aggregation of clusters of loops modeled by a discrete-time version of (5).

Let k be the current time instant, $\mathcal{P}(k)$ the chosen partition, and consider some cluster $\mathcal{C}_j \in \mathcal{P}(k)$. Then, the prediction model of $\mathcal{C}_j$ is given by:

$$\begin{aligned} T^{\text{out}}_{\mathcal{C}_j}(n+\delta^{\text{c}}|k) = T^{\text{out}}_{\mathcal{C}_j}(n|k) + \gamma_{\mathcal{C}_j}(n|k)\Big(\hat{\eta}_{\mathcal{C}_j}(k)\mathcal{J}_{\mathcal{C}_j}(k) \\ -q_{\mathcal{C}_j}(n|k)P_{\mathcal{C}_j}(n|k)\Big(T^{\text{out}}_{\mathcal{C}_j}(n|k) - T^{\text{in}}(k)\Big) - \hat{\hbar}_{\mathcal{C}_j}(n|k)\Big), \end{aligned} \tag{14}$$

where $\gamma_{\mathcal{C}_j}(n|k) = \Delta t^{\text{c}}/C_{\mathcal{C}_j}(n|k)$. An overall model of the field can therefore be defined as the aggregation of (14) for all $\mathcal{C}_j \in \mathcal{P}(k)$. Considering the latter, the MPC problem solved by the supervisor is the following:

$$\min_{\mathbf{q}(\cdot|k)} \sum_{\mathcal{C}_j\in\mathcal{P}(k)} \sum_{n\in\mathcal{H}^{\text{top}}} \Big(w^{\text{top}}_{\text{e},\mathcal{C}_j} \|e_{\mathcal{C}_j}(n+1|k)\|^2 + w^{\text{top}}_{\text{q},\mathcal{C}_j} \|\Delta q_{\mathcal{C}_j}(n|k)\|^2 \Big)$$

s.t. Clusters prediction model (14),

$$T^{\text{out}}_{\mathcal{C}_j}(k|k) = \frac{\sum_{i\in\mathcal{C}_j} q_i(k-1) T^{\text{out}}_i(k)}{\sum_{i\in\mathcal{C}_j} q_i(k-1)}, \tag{15a}$$

$$T^{\min} \le T^{\text{out}}_{\mathcal{C}_j}(n|k) \le T^{\max}, \tag{15b}$$

$$|\mathcal{C}_j| q^{\min} \le q_{\mathcal{C}_j}(n|k) \le |\mathcal{C}_j| q^{\max}, \tag{15c}$$

$$\sum_{\mathcal{C}_j\in\mathcal{P}(k)} q_{\mathcal{C}_j}(n|k) \le Q_{\text{T}}, \tag{15d}$$

$$\forall \mathcal{C}_j \in \mathcal{P}(k),\ \forall n \in \mathcal{H}^{\text{top}}, \tag{15e}$$

where $w^{\text{top}}_{\text{e},\mathcal{C}_j}$ and $w^{\text{top}}_{\text{q},\mathcal{C}_j}$ are positive definite weighting scalars, $\mathbf{q}(n|k) = [q_{\mathcal{C}_j}(n|k)]_{\mathcal{C}_j\in\mathcal{P}(k)}$ is a vector aggregating the predicted flow rates for all clusters at instant n, and $\mathcal{H}^{\text{top}} = \{k, k+\delta^{\text{c}}, k+2\delta^{\text{c}}, \dots, k+\delta^{\text{c}} N^{\text{top}}_{\text{p}}\}$, i.e., the prediction horizon is set as an interval of $\delta^{\text{c}} N^{\text{top}}_{\text{p}}$ integration steps ahead. Also, $|\mathcal{C}_j|$ is the cardinality of set $\mathcal{C}_j$, i.e., the number of loops in the cluster, and

$$\begin{aligned} e_{\mathcal{C}_j}(n+\delta^{\text{c}}|k) &= T^{\text{out}}_{\mathcal{C}_j}(n+\delta^{\text{c}}|k) - T^{\text{ref}}(n+\delta^{\text{c}}), \\ \Delta q_{\mathcal{C}_j}(n+\delta^{\text{c}}|k) &= q_{\mathcal{C}_j}(n+\delta^{\text{c}}|k) - q_{\mathcal{C}_j}(n|k), \quad \forall n \in \mathcal{H}^{\text{top}}. \end{aligned} \tag{16}$$

Note that constraint (15d) implies that the sum of all clusters' allocated flow rates do not exceed the total flow rate that can be used in the plant (see (3)). The minimizer obtained by solving (15) is therefore a sequence of HTF flow rates to be pumped to different clusters of loops during the following time steps, say

$$\Big[q^*_{\mathcal{C}_j}(k|k), \quad q^*_{\mathcal{C}_j}(k+\delta^{\text{c}}|k), \quad \dots, \quad q^*_{\mathcal{C}_j}(k+\delta^{\text{c}} N^{\text{top}}_{\text{p}}|k)\Big],$$

for all $\mathcal{C}_j \in \mathcal{P}(k)$. Considering this, let us define

$$\bar{\alpha}_{\mathcal{C}_j}(n) = Q_{\text{T}} \frac{q^*_{\mathcal{C}_j}(n|k)}{\sum_{\mathcal{C}_j\in\mathcal{P}(k)} q^*_{\mathcal{C}_j}(n|k)}, \quad n \in \mathcal{H}^{\text{top}}, \tag{17}$$

as a factor that will be used to scale the flow constraints of the MPC agents at the bottom layer. Notice that, given (17), $\bar{\alpha}_{\mathcal{C}_j}(n) > 0$ and $\sum_{\mathcal{C}_j} \bar{\alpha}_{\mathcal{C}_j}(n) = 1$, for all $n \in \mathcal{H}^{\text{top}}$.

Remark 4. If the maximum flow rate Q_{T} is evenly distributed among all clusters, then each $\mathcal{C}_j$ would have to satisfy $q_{\mathcal{C}_j}(\cdot) \le Q_{\text{T}}|\mathcal{C}_j|/N_{\text{loops}}$ so that the overall system meets (3). Instead of using ratio $|\mathcal{C}_j|/N_{\text{loops}}$, these constraints can be made more or less restrictive by introducing time-varying factors $\bar{\alpha}_{\mathcal{C}_j}(\cdot)$, i.e., $q_{\mathcal{C}_j}(\cdot) \le \bar{\alpha}_{\mathcal{C}_j}(\cdot) Q_{\text{T}}$. The latter allows us to readjust the amount of HTF available in different clusters, e.g., to provide greater amount of flow to those clusters receiving higher DNI.

Finally, to limit the extent to which the flow constraints are reduced, the scale factors resulting from (17) can be *corrected* so that they become greater than a certain threshold. For example, consider that at least a flow rate of $\zeta Q_{\text{T}}/N_{\text{loops}}$ must be available for every loop at all time steps, where $\zeta > 0$. Then, instead of using $\bar{\alpha}_{\mathcal{C}_j}(\cdot)$, we consider in (7) scale factors $\alpha_{\mathcal{C}_j}(\cdot) = \bar{\alpha}_{\mathcal{C}_j}(\cdot) + \varepsilon_{\mathcal{C}_j}(\cdot)$, where the value of $\varepsilon_{\mathcal{C}_j}(\cdot)$ is computed by solving the following problem:

$$\begin{aligned} \min_{[\varepsilon_{\mathcal{C}_j}(\cdot)]_{\mathcal{C}_j\in\mathcal{P}(k)}} & \sum_{\mathcal{C}_j\in\mathcal{P}(k)} \sum_{n\in\mathcal{H}^{\text{top}}} \varepsilon_{\mathcal{C}_j}(n)^2 \\ \text{s.t.} \quad & \alpha_{\mathcal{C}_j}(n) = \bar{\alpha}_{\mathcal{C}_j}(n) + \varepsilon_{\mathcal{C}_j}(n) \ge \zeta|\mathcal{C}_j|/N_{\text{loops}}, \\ & \sum_{\mathcal{C}_j\in\mathcal{P}(k)} \alpha_{\mathcal{C}_j}(n) = 1. \end{aligned} \tag{18}$$

3.3. Control algorithm

The proposed hierarchical MPC algorithm is summarized in Algorithm 1. The overall solar field is simulated in step 11 of this algorithm as an aggregation of parallel loops modeled as

$$\begin{aligned} T^{\text{out}}_i(k+1) = T^{\text{out}}_i(k) + \gamma(k)\big(\eta_i(k)\mathcal{J}_i(k) \\ -q_i(k)P_i(k)\big(T^{\text{out}}_i(k) - T^{\text{in}}(k)\big) - \hbar_i(k)\big), \end{aligned} \tag{19}$$

where $\gamma(k) = \Delta t^{\text{s}}/C_i(k)$, and Δt^{s} is the simulation step size (see Section 3). The outlet temperature of the overall field at instant k is accordingly defined as

$$T^{\text{out}}(k+1) = \frac{\sum_{i\in\mathcal{N}} q_i(k) T^{\text{out}}_i(k+1)}{\sum_{i\in\mathcal{N}} q_i(k)}. \tag{20}$$

In addition, the inlet temperature is governed by the following first order system:

$$\frac{T^{\text{in}}(s)}{T^{\text{out}}(s) - 80} = \frac{1}{600s+1} \tag{21}$$

where T^{in} is expressed as a function of the outlet temperature of the steam generator, approximated here as $T^{\text{out}} - 80$ °C. Finally, regarding the flows optimization in step 9 of Algorithm 1, it is sufficient for one of the agents in each coalition to solve (7) and communicate the solution to the rest of members. In this case, the agents in each coalition would internally adopt a star communication topology to share the necessary data with the one performing the optimization. Alternatively, we may consider distributed optimization methods, such as ADMM

Algorithm 1 Control algorithm

Let $\mathcal{K}^{\text{c}} = \{0, \delta^{\text{c}}, 2\delta^{\text{c}}, \dots\}$ be the set of instants in which the flow rates are optimized and updated. Also, assume that the partition into clusters is revised each $\Delta t^{\text{cl}} = \delta^{\text{cl}}\Delta t^{s}$ seconds, and, similarly, define $\mathcal{K}^{\text{cl}} = \{0, \delta^{\text{cl}}, 2\delta^{\text{cl}}, \dots\}$. Then, at all time instants k, proceed as follows:

1: **if** $k \in \mathcal{K}^{\text{c}}$ **then**
2: Compute matrix $\mathbf{P}(k)$ given in (13).
3: **if** $k \in \mathcal{K}^{\text{cl}}$ **then**
4: Update the clusters as described in Section 3.2.1 and define partition $\mathcal{P}(k)$.
5: **end if**
6: **if** $\|\mathbf{P}(k) - \mathbf{P}(k-1)\|_\infty > \epsilon$ **or** $\mathcal{P}(k) \neq \mathcal{P}(k-1)$ **then**
7: The supervisor solves (15), and updates scale factors $\alpha_{\mathcal{C}_j}$ for all $\mathcal{C}_j \in \mathcal{P}(k)$ by using (17) and (18).
8: **end if**
9: All coalitions $\mathcal{C}_j \in \mathcal{P}(k)$ solve problem (7) and find the flow rates to be implemented during interval $[k, k+\delta^{\text{c}})$.
10: **end if**
11: Update the system state and parameters considering (19), (2), and the current flow rates for each loop $i \in \mathcal{N}$.

or ALADIN [46], where all the agents in each coalition will perform iterative computations until converging to a solution. While we will consider the former approach in our simulations, both are equally valid and lead to similar results.

4. Simulation results

In this section, we apply the presented coalitional control algorithm to a model of a solar parabolic trough plant, and provide a comparison with decentralized MPC, centralized MPC, and with the case in which no loops valves are used. Moreover, we consider the following scenarios:

(i) Parameters η_i, $h_{1,i}$, and $h_{2,i}$ are identified as described in Section 3.1.2 for all loops $i \in \mathcal{N}$.
(ii) Efficiency η_i is identified following Section 3.1.2, but the heat losses coefficients are set to nominal values $\overline{h}_1 = -0.06133$ and $\overline{h}_2 = 0.00249$ for all loops $i \in \mathcal{N}$.[3]
(iii) Both the efficiency and heat losses coefficients are set to constant nominal values $\overline{\eta} = 0.7$, $\overline{h}_1$ and $\overline{h}_2$ for all loops $i \in \mathcal{N}$.

For a better illustration of the algorithm performance, we first present results using a 10-loop plant, and subsequently on a extended plant with 80 loops. All simulations have been carried out in a 1.8 GHz quad-core Intel® Core™ i7/8 GB RAM computer using Matlab® and software CasADi [47] for solving non-linear problems (7) and (15).[4] Besides the properties introduced in (2) (i.e., ρ_i and c_i for all $i \in \{1, 2, \dots, N_{\text{loops}}\}$), other parameters used in the simulations are given in Table 1. Note that the loops have been modeled as those of the ACUREX solar plant of the *Plataforma Solar de Almería* (see Chapter 4 of Camacho et al. [18]). Likewise, we considered $Q_{\text{T}} = 9 \cdot 10^{-4} N_{\text{loops}}$ [m^3/s], where N_{loops} is equal to 10 or 80 as appropriate. Finally, $w^{\text{top}}_{\text{e},\mathcal{C}_j}$ and $w^{\text{top}}_{\text{q},\mathcal{C}_j}$ are respectively defined as $|\mathcal{C}_j| w_{\text{e}}$ and $|\mathcal{C}_j| w_{\text{q}}$, where $|\mathcal{C}_j|$ is the size of the corresponding cluster.

Notice that in the centralized MPC case, a single controller optimizes all loops flow rates considering prediction model (6) for all $i = 1, \dots, N_{\text{loops}}$. Likewise, the flow rate constraints in the centralized problem (recall (3)) are given by $q^{\min} \leq q_i(\cdot|k) \leq q^{\max}$ and $\sum_{i=1}^{N_{\text{loops}}} q_i(\cdot|k) \leq Q_{\text{T}}$. On the other hand, in the decentralized case, each agent i considers only its loops dynamics and imposes as flow constraints $q^{\min} \leq q_i(\cdot|k) \leq Q_{\text{T}}/N_{\text{loops}}$ to satisfy (3). Therefore, the centralized performance would match that of a grand-coalition $\mathcal{P}(k) = \mathcal{N} = \{1, 2, \dots, N_{\text{loops}}\}$ during the entire simulation, with $\alpha_{\mathcal{N}}(k) = 1$, for all $k \geq 0$. Conversely, the decentralized case corresponds to the case of imposing partition $\mathcal{P}(k) = \{\{1\}, \{2\}, \dots, \{N_{\text{loops}}\}\}$ and $\alpha_{\{i\}} = 1/N_{\text{loops}}$ for all $k \geq 0$ and $i = 1, \dots, N_{\text{loops}}$. Finally, in the no-valves case, the HTF flow rate is evenly distributed among all loops at all time steps. In particular, the control model considers the loops dynamics, but we impose that the HTF flow rate is equal for all loops.

4.1. 10-loop plant

The simulations of the 10-loop plant consider a time period of 7 h (from 8:30 am to 3:30 pm) of a day in which the DNI received by each loop and the ambient temperature evolve as shown in Fig. 3. As can be seen, there are temporary drops in the irradiance evolution due to the effect of passing clouds.

Fig. 4 shows the evolution of the loops outlet temperature and the HTF flows in the coalitional, decentralized and no-valves cases considering scenario (i) described at the beginning of this section.

[3] Coefficients $\overline{h}_1$ and $\overline{h}_2$ are those modeling the heat losses of the ACUREX plant. In our simulations, the actual $h_{1,i}$ and $h_{2,i}$ belong in range $[0.85\overline{h}_1, 1.15\overline{h}_1]$ and $[0.85\overline{h}_2, 1.15\overline{h}_2]$, respectively, for all $i \in \mathcal{N}$.

[4] The constraints on the maximum outlet temperature have been imposed as soft constraints in the presented simulations.

Table 1
Parameters used in the simulation.

	Value	Unit		Value	Unit		Value	Unit
$q^{\min}$	$0.2 \cdot 10^{-3}$	m^3/s	S_{loop}	267.4	m^2	Δt^{s}	0.5	s
$q^{\max}$	$2 \cdot 10^{-3}$	m^3/s	w_{e}	1	–	Δt^{c}	30	s
$T^{\min}$	220	°C	w_{q}	$5 \cdot 10^{8}$	–	Δt^{cl}	150	s
$T^{\max}$	305	°C	N_{p}	5	–	ζ	0.1	–
A	$5.067 \cdot 10^{-4}$	m^2	$N_{\text{p}}^{\text{top}}$	10	–	$\hat{I}_0$	$1 \cdot 10^{-4}$	–
L	142	m	ϵ	60	–	λ	0.02	–

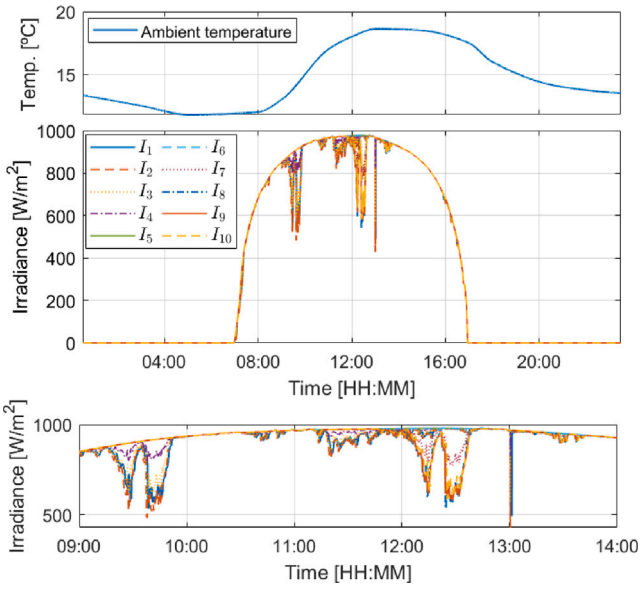
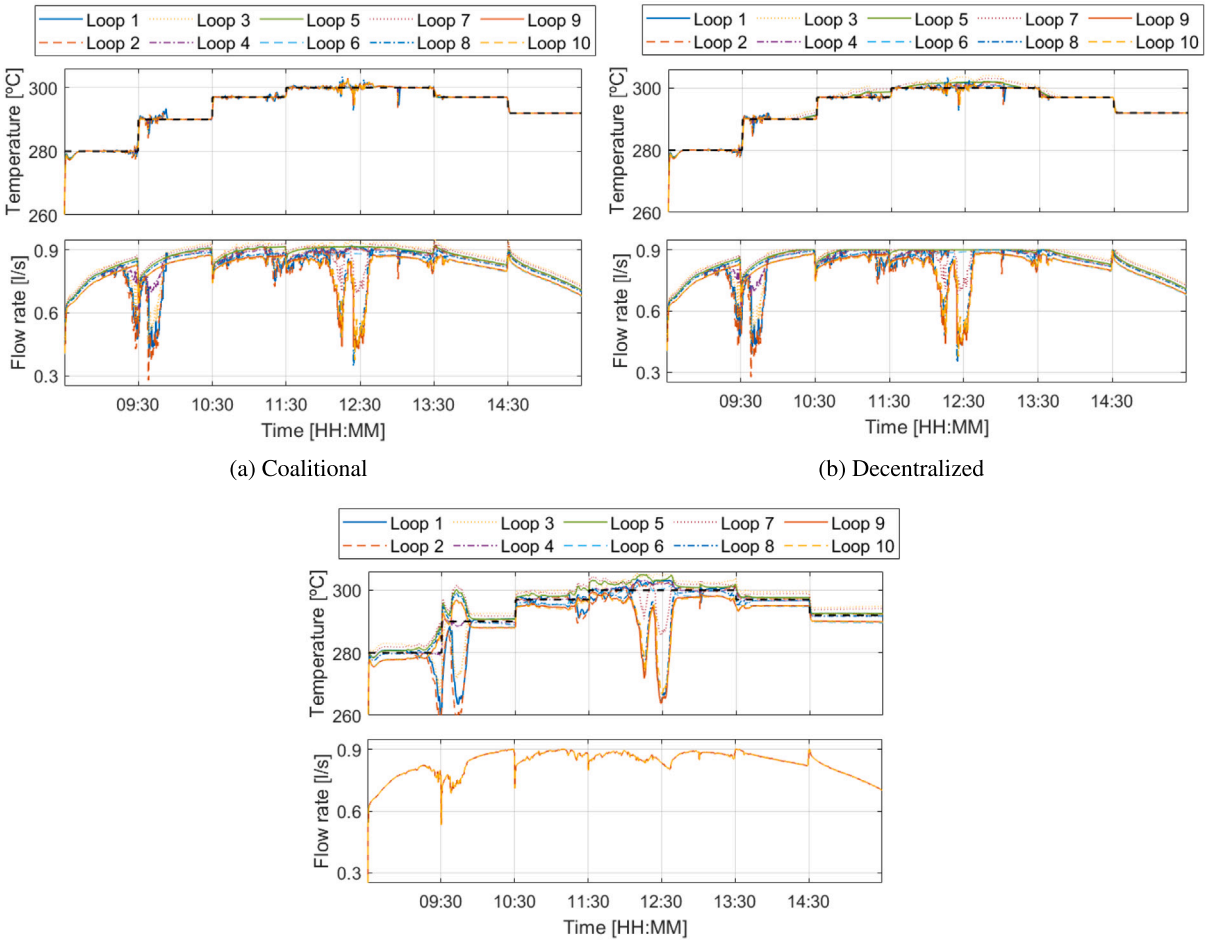


Fig. 3. Evolution of the ambient temperature (top) and of the DNI for each loop (middle). The bottom plot zooms the irradiance graph in a time period affected by clouds.

As can be seen, in the coalitional and decentralized cases, the loops outlet temperature follows closely the reference, and the flow rates decrease as the irradiance falls. However, if no valves are used and the temperature of some loop is close to $T^{\max}$, then the controller tends to set all loops flows close to the maximum possible, thus decreasing their temperature even for those that were already below the reference. The centralized graphs have not been added to Fig. 4 because they were very similar to the coalitional ones. Indeed, the maximum difference between the loops outlet temperature in both cases was 3 °C. This is also reflected in Table 2, which provides the cumulative performance costs in each of the simulations (see rows related to scenario (i)). In particular, it provides the value of the following indexes:

$$J_{\text{l}} = \sum_k \sum_{i=1}^{N_{\text{loops}}} \left(w_{\text{e}} e_i^2(k) + w_{\text{q}} (q_i(k) - q_i(k-1))^2 \right), \quad \text{and}$$
$$J_{\text{g}} = \sum_k \left(w_{\text{e}} \|e(k)\|^2 + \sum_{i=1}^{N_{\text{loops}}} w_{\text{q}} (q_i(k) - q_i(k-1))^2 \right), \tag{22}$$

where $e_i(k) = T_i^{\text{out}}(k) - T^{\text{ref}}(k)$ and $e(k) = T^{\text{out}}(k) - T^{\text{ref}}(k)$. Note that J_{l} weights the local error in the outlet temperature of each loop, whereas J_{g} considers the error in the global outlet temperature of the solar field (see (20)). In view of Fig. 4 and Table 2, it should also be noted that the centralized and coalitional controllers introduce further flexibility in the distribution of the HTF among the loops, which allows for a better performance considering the same Q_{T}. While in the decentralized case all loops flows are kept lower or equal $Q_{\text{T}}/10$, this threshold is occasionally exceeded in the centralized and coalitional cases without compromising the satisfaction of constraints (3). In this respect, Fig. 5 provides the evolution of the total HTF pumped to the solar field, i.e., $\sum_{i=1}^{10} q_i(k)$, during part of the simulations of scenario (i).



(a) Coalitional

(b) Decentralized

(c) No valves

Fig. 4. Evolution of the loops outlet temperature and of the HTF flow rates in the 10-loop plant with different controllers in scenario (i). The dashed black line indicates the reference temperature.

In addition, Figs. 6 and 7 focus on the tasks performed by the supervisor in the proposed coalitional approach, i.e., the loops clustering and the constraints update. In particular, Fig. 6 illustrates the moving averages over a sliding window of 15 min of the maximum, mean, and minimum number of loops per clusters over time. On average, there were 3.51 clusters in the system with 2.97 loops. These clusters were selected using the K-means algorithm and considering $N_{\text{cl}}^{\max} = 4$. Note that that the number and members of these coalitions are dynamically updated leading to time-varying cooperation structures at the bottom layer. In addition, Fig. 7 provides a swarm scatter chart with the values of factors α_C set by the supervisor for different clusters sizes. Recall that this factor determines flow constraints (7d), and hence the share of Q_{T} that the coalitions can use. Also, note that this type of plot illustrates the distribution of data while also showing the individual points. In particular, for a given coalition size $|C|$, the values obtained for α_C throughout the entire simulation are jittered in the x-dimension, i.e., moved aside a bit so that any of them overlap. As can be seen, many of the points concentrate around the value of $|C|/10$ for each $|C|$, which correspond to an even distribution of Q_{T} among the 10 loops. Nonetheless, we also obtained factors α_C differing from the latter, e.g., for $|C| = 3$, there are some points around 0.2. This indicates that at some time instants the flow constraints of some loops were tightened in favor of others.

Finally, Fig. 8 illustrates the estimation of the loops efficiency and thermal losses coefficients also in scenario (i). These results have been obtained by discretizing adaptation law described in Section 3.1.2 with a sampling time equal to Δt^{s}, and using as initial estimations $\hat{\eta}_0 = \overline{\eta}$ and $\hat{h}_0 = [\overline{h}_2, \overline{h}_1]^{\text{T}}$. During part of the simulation, the efficiency of three of the loops was decreased to simulate a deficient sun tracking. Note that $\hat{\eta}_i$, for all $i = 1, \dots, 10$, is adapted in real time in the simulations of the centralized, coalitional, decentralized and no-valves controllers. On the other hand, coefficients $\hat{h}_{1,i}$ and $\hat{h}_{2,i}$ are assumed to be computed during a low irradiance period on some of the previous days. In particular, Fig. 8 (top) shows the evolution of $\hat{\eta}_i$ in the coalitional case, and the computation of $\hat{h}_{1,i}$ and $\hat{h}_{2,i}$ is illustrated in Fig. 8 (bottom). That is, the values of $\hat{h}_{1,i}$ and $\hat{h}_{2,i}$ at convergence were then used in the loops control model of all presented simulations of scenario (i). Regarding

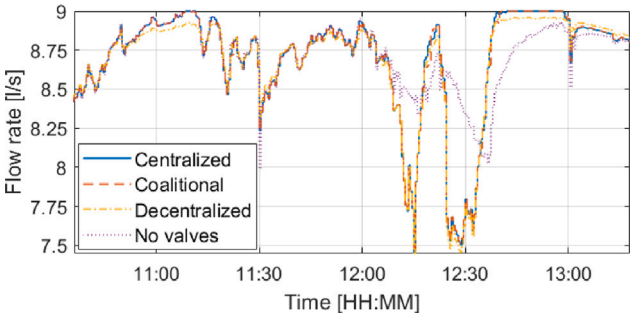


Fig. 5. Evolution of the total HTF flow rate with different controllers during a cloudy period.

Table 2
Cumulative performance costs in the 10-loop plant simulations.

		Centr.	Coalitional	Decent.	No-valves			Centr.	Coalitional	Decent.	No-valves
J_l	(i)	$2.760 \cdot 10^5$	$2.857 \cdot 10^5$	$5.288 \cdot 10^5$	$1.030 \cdot 10^7$	J_g	(i)	$2.649 \cdot 10^4$	$2.681 \cdot 10^4$	$3.701 \cdot 10^4$	$1.485 \cdot 10^5$
	(ii)	$2.812 \cdot 10^5$	$3.600 \cdot 10^5$	$5.344 \cdot 10^5$	$1.030 \cdot 10^7$		(ii)	$2.668 \cdot 10^4$	$2.925 \cdot 10^4$	$3.723 \cdot 10^4$	$1.487 \cdot 10^5$
	(iii)	$1.095 \cdot 10^6$	$1.063 \cdot 10^6$	$1.190 \cdot 10^6$	$1.079 \cdot 10^7$		(iii)	$7.661 \cdot 10^4$	$7.355 \cdot 10^4$	$6.804 \cdot 10^4$	$1.967 \cdot 10^5$

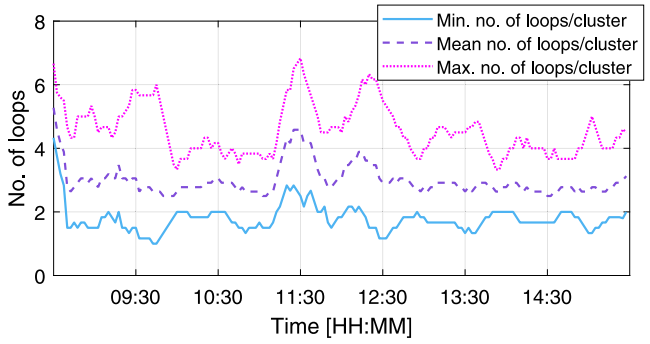


Fig. 6. Moving averages over a sliding window of 15 min of the maximum, mean and minimum number of loops per cluster in scenario (i).

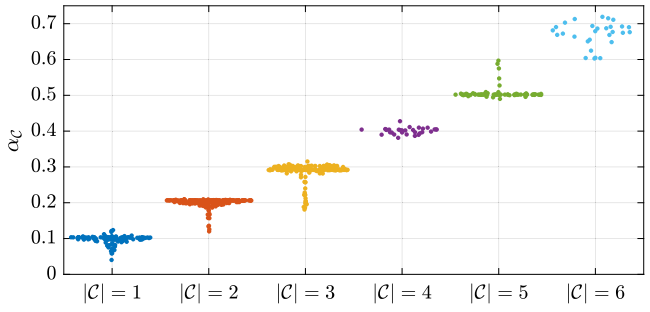


Fig. 7. Swarm chart of the scale factors $\alpha_{\mathcal{C}}$ resulting for different coalitions sizes $|\mathcal{C}|$. Note that if Q_{T} is evenly distributed among all loops, $\alpha_{\mathcal{C}}$ would be equal to $|\mathcal{C}|/10$.

scenarios (ii) and (iii), Table 2 provides the resulting values of performance indicators (22). As can be seen, the model error introduced by overestimating η_i significantly degrades the system performance and even undermines the advantage of the centralized approach against the decentralized one. Scenario (ii) provided a closer but still suboptimal approximation to the results in case (i), thus also reflecting the benefits of the parameters identification.

4.2. 80-loop plant

In the simulations of the 80-loop plant, we consider again a cloudy period of 7 h and the same evolution of the reference and ambient temperatures as before. The resulting overall outlet temperature (see (20)) and HTF flow rates with the proposed coalitional controller in scenario (i) are illustrated in Fig. 9. In contrast to Figs. 4(a), 9 indicates the maximum and minimum of these variables among the 80 loops for the sake of clarity. In addition, Table 3 compares the performance indicators in (22) with those obtained in the centralized, decentralized and no-valves cases. Both the results in Fig. 9 and Table 3 are consistent with the performance observed in the simulations of the 10-loop plant in scenario (i). In particular, the coalitional controller approximates again the centralized performance and improves the costs obtained with the decentralized controller. Likewise, it was possible to obtain a much better thermal balance among all loops than in the no-valves case as reflected by costs J_l. As for scenarios (ii) and (iii), a significant decrease of performance was again observed when unknown parameter

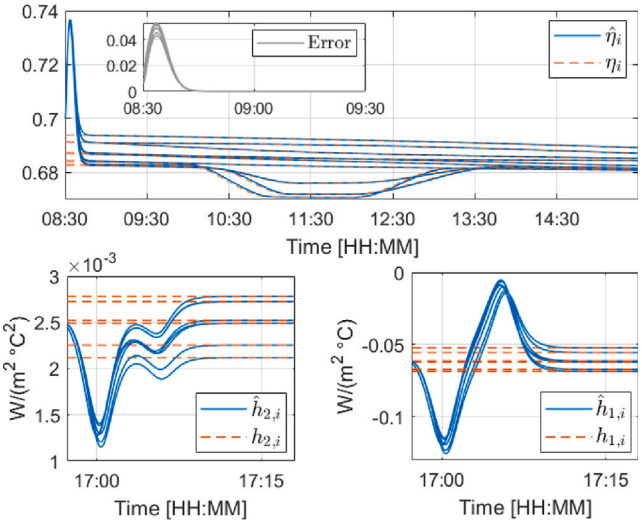


Fig. 8. Real η_i, $h_{1,i}$ and $h_{2,i}$ (dashed lines) and their estimated value (solid lines) for all loops i.

η_i is set to nominal value $\overline{\eta}$ for all loops i. In this respect, the actual efficiencies were such that $1/(80|\mathcal{K}|)\sum_{k=0}^{|\mathcal{K}|}\sum_{i=1}^{80}(\eta_i(k)-\overline{\eta})^2 = 1.922 \cdot 10^{-4}$, where $|\mathcal{K}|$ denotes the number of simulation time steps.

Fig. 10 illustrates the formation of coalitions at two different time steps in scenario (i). In particular, the plots axis are associated with the estimated effective irradiance of the loops ($\hat{\eta}_i \mathcal{I}_i$) and their thermal losses ($\hat{h}_i$), which are the most relevant features that determine the clustering (see (13)). The markers color indicate the partition obtained implementing the K-means algorithm and considering $N_{\mathrm{cl}}^{\max} = 12$. With this maximum number of clusters, the mean number of agents in the selected coalitions was 9.1. Regarding the estimation of parameters $\hat{\eta}_i$, $\hat{h}_{1,i}$ and $\hat{h}_{2,i}$, the results were also similar to those of the 10-loop plant. After a transient period of about 20 min, the maximum error between the estimated and actual efficiencies was $7.07 \cdot 10^{-4}$; and the error on the heat losses coefficients was below $5.28 \cdot 10^{-8}$.

This larger system with 80 loops allows for a better comparison of the different control approaches in terms of computation time. To this end, consider the following index:

$$\bar{T} = \frac{1}{|\mathcal{K}^{\mathrm{c}}|}\sum_{k\in\mathcal{K}^{\mathrm{c}}}\max_{\mathcal{C}_j\in\mathcal{P}(k)} t_{\mathcal{C}_j}(k), \tag{23}$$

where $t_{\mathcal{C}_j}(\cdot)$ denotes the time to setup and solve (7) by the jth coalition in the system. Recall that $\mathcal{K}^{\mathrm{c}}$ is the set of instants in which the flows are updated (see Algorithm 1), and that in the centralized and decentralized cases we always have a single grand-coalition and a set of N_{loops} singletons, respectively. Also, notice that (23) considers the maximum time among all coalitions since they can optimize their actions in parallel. In scenario (i), the resulting value of (23) was 2.3234 s in the centralized case, 0.6597 s in the coalitional case, and 0.0614 s in the decentralized case. Moreover, Fig. 11 illustrates the time to solve problem (7) for different coalition sizes. As expected, choosing larger clusters increases the number of decision variables involved in (7) and hence the computation times. Finally, notice that the centralized approach requires solving (7) for $\mathcal{P}(k) = \mathcal{N}$ at all control time steps k. That is, $N_{\mathrm{p}} \cdot N_{\mathrm{loops}}$ (i.e., 400) flow variables were centrally optimized at 840 instants of the simulation. On the other hand, the coalitional

Table 3
Cumulative performance costs in the 80-loop plant simulations.

		Centr.	Coalitional	Decent.	No-valves			Centr.	Coalitional	Decent.	No-valves
J_l	(i)	$2.012 \cdot 10^6$	$2.100 \cdot 10^6$	$2.603 \cdot 10^6$	$2.883 \cdot 10^7$	J_g	(i)	$3.813 \cdot 10^4$	$3.818 \cdot 10^4$	$3.969 \cdot 10^4$	$5.677 \cdot 10^4$
	(ii)	$2.025 \cdot 10^6$	$2.353 \cdot 10^6$	$2.616 \cdot 10^6$	$2.883 \cdot 10^7$		(ii)	$3.816 \cdot 10^4$	$3.844 \cdot 10^4$	$3.972 \cdot 10^4$	$5.677 \cdot 10^4$
	(iii)	$5.890 \cdot 10^6$	$5.724 \cdot 10^6$	$6.098 \cdot 10^6$	$3.110 \cdot 10^7$		(iii)	$7.236 \cdot 10^4$	$6.992 \cdot 10^4$	$6.910 \cdot 10^4$	$8.452 \cdot 10^4$

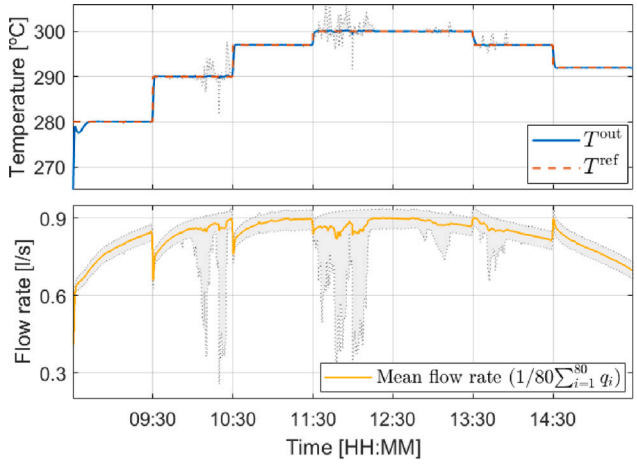


Fig. 9. Outlet temperature and HTF flow in the 80-loop plant with the coalitional controller in scenario (i). The gray dotted lines indicate accordingly the maximum and minimum outlet temperature and flow rates among the set of loops.

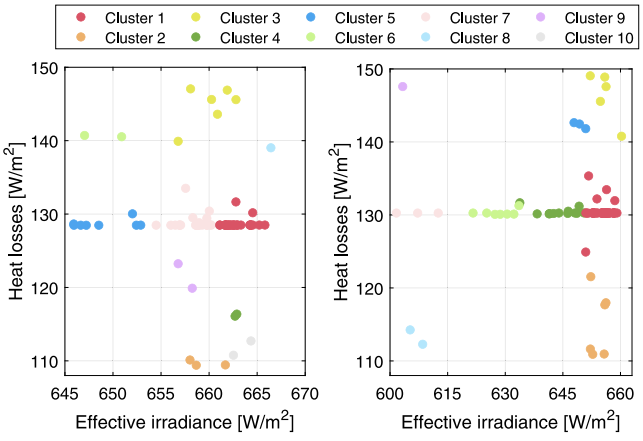


Fig. 10. Clusters formation according to the loops dynamics.

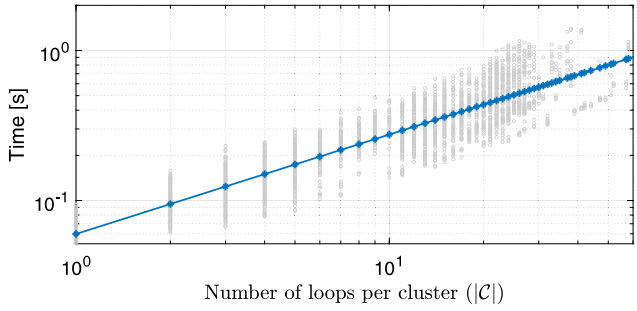


Fig. 11. Computation time to optimize the flow rates for different coalition sizes $|\mathcal{C}|$.

approach only considered centralized computations at 168 instants to solve problem (15), which in turn considers a simplified description of the field with $N_p^{top} \cdot N_{cl}$ flow variables (i.e., about 90 flow variables on average).

5. Conclusions

A hierarchical clustering-based MPC for optimizing the HTF flow rates in large-scale solar parabolic trough plants has been presented. The proposed approach is built on the idea that the HTF circulating through the loops of solar collectors are controlled in real time by a set of MPC agents. In turn, a supervisor partitions dynamically these agents into cooperative clusters (also called coalitions) and assigns to each of them a maximum flow rate considering the overall conditions and constraints of the solar field. Moreover, the efficiency and heat losses coefficients that determine the loops control model are adapted by using measurements of their outlet temperature.

Our results show that the proposed clustering-based MPC can closely approximate the centralized performance, while reducing the need for centralized computations and the coordination efforts. In particular, the loss of performance regarding the overall outlet temperature was under 1.21% in both simulations of a 10-loop and an 80-loop plant. The supervisor allows the MPC agents to take into account the varying conditions among the loops through their constraints adjustment. Therefore, it can restrict the flow that can be pumped to loops covered by clouds to provide more HTF to those receiving higher irradiance. On the other hand, the formation of coalitions among the MPC agents allows for reducing both the communication and the number of variables that they must coordinate, thus also reducing the computation demands.

Future research will optimize the outlet temperature reference so as to maximize the net electricity production, as well as studying a fully distributed implementation of the proposed controller. In this regard, we will also explore the potential performance benefits that can be achieved by dynamically adjusting the temperature reference of each of the clusters of loops.

CRediT authorship contribution statement

Paula Chanfreut: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Writing – original draft, Writing – review & editing, Visualization. **José M. Maestre:** Conceptualization, Formal analysis, Writing – review & editing, Supervision, Project administration, Funding acquisition. **Antonio Gallego:** Conceptualization, Methodology, Formal analysis, Validation, Writing – review & editing. **Anuradha M. Annaswamy:** Conceptualization, Methodology, Formal analysis, Validation, Writing – review & editing. **Eduardo F. Camacho:** Conceptualization, Methodology, Formal analysis, Validation, Supervision, Project administration, Funding acquisition.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Eduardo Fernandez Camacho reports financial support was provided by European Research Council. Jose Maria Maestre reports financial support was provided by Spain Ministry of Science and Innovation.

References

[1] International Energy Agency (IEA), Renewable electricity, 2022, URL: https://www.iea.org/reports/renewable-electricity.

[2] Renewable Energy Policy Network for the 21st Century (REN21), Renewables 2022: Global status report, 2022, URL: https://www.ren21.net/wp-content/uploads/2019/05/GSR2022_Full_Report.pdf.

[3] O. Coddington, J. Lean, P. Pilewskie, M. Snow, D. Lindholm, A solar irradiance climate data record, Bull. Am. Meteorol. Soc. 97 (7) (2016) 1265–1282.

[4] World Energy Council (WEC), World energy resources: Solar, 2013, URL: https://www.worldenergy.org/publications/entry/world-energy-resources-2013-survey.

[5] D.A. Baharoon, H.A. Rahman, W.Z.W. Omar, S.O. Fadhl, Historical development of concentrating solar power technologies to generate clean electricity efficiently–A review, Renew. Sustain. Energy Rev. 41 (2015) 996–1027.

[6] H. Zhang, J. Baeyens, J. Degrève, G. Cacères, Concentrated solar power plants: Review and design methodology, Renew. Sustain. Energy Rev. 22 (2013) 466–481.

[7] M.H. Ahmadi, M. Ghazvini, M. Sadeghzadeh, M. Alhuyi Nazari, R. Kumar, A. Naeimi, T. Ming, Solar power technology for electricity generation: A critical review, Energy Sci. Eng. 6 (5) (2018) 340–361.

[8] A. Giostri, M. Binotti, P. Silva, E. Macchi, G. Manzolini, Comparison of two linear collectors in solar thermal plants: parabolic trough versus fresnel, J. Sol. Energy Eng. 135 (1) (2013) 011001.

[9] I.L. García, J.L. Álvarez, D. Blanco, Performance model for parabolic trough solar thermal power plants with thermal storage: Comparison to operating plant data, Sol. Energy 85 (10) (2011) 2443–2460.

[10] M. Liu, N.S. Tay, S. Bell, M. Belusko, R. Jacob, G. Will, W. Saman, F. Bruno, Review on concentrating solar power plants and new developments in high temperature thermal energy storage technologies, Renew. Sustain. Energy Rev. 53 (2016) 1411–1432.

[11] G. Alva, Y. Lin, G. Fang, An overview of thermal energy storage systems, Energy 144 (2018) 341–378.

[12] H. Bouziane, B. Benhamou, Assessment of the impact of thermal energy storage operation strategy on parabolic trough solar power plant performance, Renew. Energy 202 (2023) 713–720.

[13] T. Eddine, M.-S. Mecibah, Parabolic trough solar thermal power plant: Potential, and projects development in Algeria, Renew. Sustain. Energy Rev. 21 (2013) 288–297.

[14] W. Fuqiang, C. Ziming, T. Jianyu, Y. Yuan, S. Yong, L. Linhua, Progress in concentrated solar power technology with parabolic trough collector system: A comprehensive review, Renew. Sustain. Energy Rev. 79 (2017) 1314–1328.

[15] M.T. Islam, N. Huda, A. Abdullah, R. Saidur, A comprehensive review of state-of-the-art concentrating solar power (CSP) technologies: Current status and research trends, Renew. Sustain. Energy Rev. 91 (2018) 987–1018.

[16] The World Bank, Concentrating Solar Power, Clean Power on Demand 24/7, Washington, DC, 2021.

[17] K. Vignarooban, X. Xu, A. Arvay, K. Hsu, A.M. Kannan, Heat transfer fluids for concentrating solar power systems–a review, Appl. Energy 146 (2015) 383–396.

[18] E.F. Camacho, M. Berenguel, F.R. Rubio, D. Martínez, Control of Solar Energy Systems, Springer London, 2012, http://dx.doi.org/10.1007/978-0-85729-916-1.

[19] G. Andrade, D. Pagano, J.D. Alvarez, M. Berenguel, A practical NMPC with robustness of stability applied to distributed solar power plants, Sol. Energy 92 (2013) 106–122.

[20] S.J. Navas, F.R. Rubio, P. Ollero, J.M. Lemos, Optimal control applied to distributed solar collector fields with partial radiation, Sol. Energy 159 (2018) 811–819.

[21] A.O. López-Bautista, A. Flores-Tlacuahuac, M.A. Gutiérrez-Limón, Robust model predictive control for a nanofluid based solar thermal power plant, J. Process Control 94 (2020) 97–109.

[22] T. Gholaminejad, A. Khaki-Sedigh, Stable deep koopman model predictive control for solar parabolic-trough collector field, Renew. Energy 198 (2022) 492–504.

[23] B. Kouvaritakis, M. Cannon, Model Predictive Control, Vol. 38, Springer International Publishing, Springer, Switzerland, 2016.

[24] J.B. Rawlings, D.Q. Mayne, M. Diehl, Model Predictive Control: Theory, Computation, and Design, Vol. 2, Nob Hill Publishing Madison, WI, 2017.

[25] S.J. Qin, T.A. Badgwell, A survey of industrial model predictive control technology, Control Eng. Pract. 11 (7) (2003) 733–764.

[26] D.Q. Mayne, Model predictive control: Recent developments and future promise, Automatica 50 (12) (2014) 2967–2986.

[27] P.D. Christofides, R. Scattolini, D.M. de la Pena, J. Liu, Distributed model predictive control: A tutorial review and future research directions, Comput. Chem. Eng. 51 (2013) 21–41.

[28] B.T. Stewart, A.N. Venkat, J.B. Rawlings, S.J. Wright, G. Pannocchia, Cooperative distributed model predictive control, Systems Control Lett. 59 (8) (2010) 460–469.

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

[30] Y. Wei, S. Li, Y. Zheng, Enhanced information reconfiguration for distributed model predictive control for cyber-physical networked systems, Internat. J. Robust Nonlinear Control 30 (1) (2020) 198–221.

[31] J. Barreiro-Gomez, C. Ocampo-Martinez, N. Quijano, Time-varying partitioning for predictive control design: Density-games approach, J. Process Control 75 (2019) 1–14.

[32] F. Fele, J.M. Maestre, E.F. Camacho, Coalitional control: Cooperative game theory and control, IEEE Control Syst. Mag. 37 (1) (2017) 53–69.

[33] OCONTSOLAR, ERC advanced grant, 2018, https://www.europeandissemination.eu/ocontsolar-project/2563.

[34] M. Abutayeh, R.V. Padilla, M. Lake, Y.Y. Lim, J. Garcia, M. Sedighi, Y.C.S. Too, K. Jeong, Effect of short cloud shading on the performance of parabolic trough solar power plants: motorized vs manual valves, Renew. Energy 142 (2019) 330–344.

[35] E. Masero, J.R.D. Frejo, J.M. Maestre, E.F. Camacho, A light clustering model predictive control approach to maximize thermal power in solar parabolic-trough plants, Sol. Energy 214 (2021) 531–541.

[36] A. Sánchez, A. Gallego, J. Escano, E. Camacho, Thermal balance of large scale parabolic trough plants: A case study, Sol. Energy 190 (2019) 69–81.

[37] E. Camacho, A. Gallego, Optimal operation in solar trough plants: A case study, Sol. Energy 95 (2013) 106–117.

[38] İbrahim Halil Yılmaz, A. Mwesigye, Modeling, simulation and performance analysis of parabolic trough solar collectors: A comprehensive review, Appl. Energy 225 (2018) 135–174.

[39] S. Wei, X. Liang, T. Mohsin, X. Wu, Y. Li, A simplified dynamic model of integrated parabolic trough concentrating solar power plants: Modeling and validation, Appl. Therm. Eng. 169 (2020) 114982.

[40] V. Shaferman, M. Schwegel, T. Glück, A. Kugi, Continuous-time least-squares forgetting algorithms for indirect adaptive control, Eur. J. Control 62 (2021) 105–112.

[41] E. Lavretsky, Combined/composite model reference adaptive control, IEEE Trans. Automat. Control 54 (11) (2009) 2692–2697.

[42] R. Xu, D. Wunsch, Survey of clustering algorithms, IEEE Trans. Neural Netw. 16 (3) (2005) 645–678.

[43] M. Ahmed, R. Seraj, S.M.S. Islam, The k-means algorithm: A comprehensive survey and performance evaluation, Electronics 9 (8) (2020) 1295.

[44] Y. Liu, Z. Li, H. Xiong, X. Gao, J. Wu, Understanding of internal clustering validation measures, in: 2010 IEEE International Conference on Data Mining, IEEE, 2010, pp. 911–916.

[45] T. Caliński, J. Harabasz, A dendrite method for cluster analysis, Comm. Statist. Theory Methods 3 (1) (1974) 1–27.

[46] B. Houska, J. Frasch, M. Diehl, An augmented Lagrangian based algorithm for distributed nonconvex optimization, SIAM J. Optim. 26 (2) (2016) 1101–1127.

[47] J.A.E. Andersson, J. Gillis, G. Horn, J.B. Rawlings, M. Diehl, CasADi – A software framework for nonlinear optimization and optimal control, Math. Program. Comput. 11 (1) (2019) 1–36.