2023 62nd IEEE Conference on Decision and Control (CDC)
December 13-15, 2023. Marina Bay Sands, Singapore

ALADIN-based Distributed Model Predictive Control with dynamic partitioning: An application to Solar Parabolic Trough Plants

P. Chanfreut[1], J. M. Maestre[2], D. Krishnamoorthy[1], E. F. Camacho[2]

Abstract— **This article presents a distributed model predictive controller with time-varying partitioning based on the augmented Lagrangian alternating direction inexact Newton method (ALADIN). In particular, we address the problem of controlling the temperature of a heat transfer fluid (HTF) in a set of loops of solar parabolic collectors by adjusting its flow rate. The control problem involves a nonlinear prediction model, decoupled inequality constraints, and coupled affine constraints on the system inputs. The application of ALADIN to address such a problem is combined with a dynamic clustering-based partitioning approach that aims at reducing, with minimum performance losses, the number of variables to be coordinated. Numerical results on a 10-loop plant are presented.**

I. INTRODUCTION

Over the last decades, solar energy technologies have become increasingly efficient and cost-effective, and they are now essential for the transition towards a sustainable power system. In 2021, solar power was ranked the top power generation source installed worldwide, and has recently surpassed the threshold of 1 terawatt of installed capacity [1]. While solar photovoltaics are being the kingpin for the growth of solar technologies, there are about 6 gigawatts of installed concentrating solar power (CSP), and over 1 gigawatt under construction [2]. Moreover, the incorporation of thermal energy storage systems makes CSP plants capable of dispatching power on demand, which is of particular interest to support other forms of renewable generation [3].

This paper focuses on solar parabolic trough plants, which represent the most extended CSP technology [4]. Parabolic trough plants obtain thermal energy by concentrating the solar rays on a tube through which circulates a heat transfer fluid (HTF). In this regard, the solar field consists of a set of parallel *loops*, which are rows of parabolic collectors with a tube running along their focal line. One of the main control problems that arise in this context is to control the HTF temperature around a given reference by manipulating its flow rate. While different control methods have been explored to address the latter, model predictive control (MPC) has received special attention both at research and commercial levels. See [5] (Chapter 5) for a review, and [6] and [7] for recent contributions.

Traditionally, all loops of collectors receive the same HTF flow. However, several works have pointed out that higher efficiencies can be attained by optimally allocating the flow that circulates specifically through each loop since they may exhibit disparate dynamics [8, 9]. This is because the loops may receive different irradiance levels due to cloud shading, have different optical efficiencies due to changes in their mirrors reflectivity, etc. The resulting MPC problem is a constrained optimization where the goal is then to optimally distribute the total HTF available in the plant. The sheer size of these plants, which may comprise more than 800 loops as in SOLANA [10], hinders the applicability of centralized MPC. Pursuing increased scalability, a number of articles have explored distributed MPC (DMPC) strategies where multiple agents control subsets of loops, e.g., [11, 12]. In addition, DMPC is also favourable in terms of monitoring and maintenance. For example, if some of the loops are not operating, it will only affect some of the agents, while the rest could continue operating normally.

Within the DMPC framework, dual decomposition and the alternating direction method of multipliers (ADMM) have been extensively used for coordinating control decisions [13, 14]. Both of them involve iterative (sub)gradient methods, which often require many iterations to converge to a solution. Moreover, their theoretical properties do not generally apply in the nonconvex setting [15]. Considering these issues, this paper explores the augmented Lagrangian alternating direction inexact Newton method (ALADIN) [15, 16], which has been recently studied for optimizing power transfers in electrical networks [17, 18]. Particularly, ALADIN combines ideas of augmented Lagrangian methods and sequential quadratic programming, and is designed to solve potentially nonconvex optimization problems in a distributed manner. In contrast to ADMM, ALADIN uses both gradient and Hessian information at every iteration, and has been shown to converge faster [16]. This is beneficial for the real-time control problem underlying our solar plants application.

The main contribution of this article is a DMPC based on ALADIN with time-varying system partitioning. The proposed controller optimizes the HTF flow rates in every loop to track reference outlet temperatures, and integrates clustering methods to further increase scalability with minimum performance losses. In this regard, the solar field is dynamically partitioned into clusters of similar loops to reduce the number of optimization variables, and thus simplify the distributed computations.

*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.

[1]P. Chanfreut and D. Krishnamoorthy are with the Department of Mechanical Engineering, Eindhoven University of Technology, The Netherlands. E-mails: p.chanfreut.palacio@tue.nl, d.krishnamoorthy@tue.nl

[2]J. M. Maestre and E. F. Camacho are with the Department of Systems and Automation Engineering, University of Seville, Spain. E-mails: pepemaestre@us.es, efcamacho@us.es

The rest of the article is organized as follows. Section II presents the system dynamics, the control objectives, and the associated centralized problem. Section III describes the clustering formation, formulates the DMPC problem, and presents the proposed ALADIN-based control algorithm. Finally, Section IV presents our simulation results.

Notation: Given two time steps k and $n \geq k$, and a variable x, $x(n|k)$ indicates the predicted value of x for time n realized at k. Given a set $\mathcal{S}$, say $\mathcal{S} = \{1, 2, ..., |\mathcal{S}|\}$, $[x_i]_{i\in\mathcal{S}} = [x_i]_{i=1}^{|\mathcal{S}|}$ is the vector $[x_1, x_2, ..., x_{|\mathcal{S}|}]^\top$. Also, $|\cdot|$ denotes the cardinality when referring to a set, and the absolute value when used with scalars. Capital caligraphic letters are used for sets, whereas bold letters represent sequences. Finally, $\mathbf{1}_m$ and $\mathbf{0}_m$ are the all-ones and all-zeros vectors of dimension $m \times 1$.

II. Problem Formulation

Consider a solar parabolic trough plant comprising a set of loops $\mathcal{N} = \{1, 2, ..., N_{\rm loops}\}$ equipped with inlet valves.

A. System dynamics

The dynamics of the HTF temperature at the outlet of any loop $i \in \mathcal{N}$, i.e., $T_i^{\rm out}$ [°C], can be modeled considering the variation of its internal energy as follows:[1]

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

where $T^{\rm in}$ [°C] is the inlet temperature, and q_i [m^3/s] represents the HTF flow rate in loop i. Also, C_i [J/ºC] is the thermal capacity of the loop, P_i [J/(m^3°C)] is related to its geometrical and thermal properties, $\hbar_i$ [W] weights the heat losses of loop i, and $\eta_i \mathscr{I}_i$ [W] considers the power received from the sun. In particular, η_i weights the optical and geometric efficiency of the collectors in i, and $\mathscr{I}_i = SI_i$, with S [m^2] being the loops' reflective surface and I_i [W/m^2] the direct normal irradiance. Finally, note that some of the parameters in (1) vary as a function of the temperature. In particular, we will consider the following:[2]

$$\begin{aligned} &\rho_i = 903 - 0.672 T_i^{\rm m}, \qquad P_i = \rho_i c_i, \\ &c_i = 1820 + 3.478 T_i^{\rm m}, \qquad C_i = \rho_i c_i A L, \\ &\hbar_i = S\left(0.00249 (T_i^{\rm m} - T^{\rm a})^2 - 0.06133 (T_i^{\rm m} - T^{\rm a})\right), \end{aligned} \qquad (2)$$

where $T_i^{\rm m} = (T_i^{\rm out} + T^{\rm in})/2$, $T^{\rm a}$ [°C] is the ambient temperature, A [m^2] is the cross sectional area of the tube, and L [m] is the loops length.

1) Cluster-based model: Similar to (1), a cluster of loops $\mathcal{C} \subseteq \mathcal{N}$ can be described by the following lumped parameter model:

$$C_{\mathcal{C}} \frac{dT_{\mathcal{C}}^{\rm out}}{dt} = \eta_{\mathcal{C}} \mathscr{I}_{\mathcal{C}} - q_{\mathcal{C}} P_{\mathcal{C}} (T_{\mathcal{C}}^{\rm out} - T^{\rm in}) - \hbar_{\mathcal{C}}, \qquad (3)$$

where $T_{\mathcal{C}}^{\rm out}$ denotes the outlet temperature of cluster $\mathcal{C}$, and $q_{\mathcal{C}} = \sum_{i\in\mathcal{C}} q_i$ is the total HTF pumped to the loops in $\mathcal{C}$.

[1]For the sake of clarity, the continuous time index is omitted in Subsections II-A and II-B.

[2]The definitions in (2) consider the HTF (Therminol 55) and heat losses of the ACUREX plant, which was located in the south of Spain [5].

In what follows, we will consider $T_{\mathcal{C}}^{\rm out} = \sum_{i\in\mathcal{C}} (q_i T_i^{\rm out})/q_{\mathcal{C}}$, whereas $T^{\rm in}$ is the same for all loops, and hence for all clusters. Also, parameters $C_{\mathcal{C}}$, $P_{\mathcal{C}}$ and $\hbar_{\mathcal{C}}$ are defined analogously to (2), and $\eta_{\mathcal{C}} \mathscr{I}_{\mathcal{C}} = \sum_{i\in\mathcal{C}} \eta_i \mathscr{I}_i$. Note that if $\mathcal{C} = \mathcal{N}$, then (3) provides a lumped model of the entire solar field; whereas if $\mathcal{C} = \{i\}$, model (3) is equivalent to (1).

B. Control objectives

The proposed controller should dynamically update flows q_i for all $i \in \mathcal{N}$ to track varying references on the loops outlet temperature while satisfying the following:

$$\sum_{i\in\mathcal{N}} q_i \leq Q_{\rm T}, \qquad (4a)$$

$$q^{\rm min} \leq q_i \leq q^{\rm max}, \quad T^{\rm min} \leq T_i^{\rm out} \leq T^{\rm max}, \ \forall i \in \mathcal{N}, \quad (4b)$$

where $Q_{\rm T}$ is the maximum available HTF flow in the plant, $q_{\rm min}$ and $q_{\rm max}$ denote respectively the minimum and maximum flows allowed in the loops, and $T^{\rm min}$ and $T^{\rm max}$ are similarly the minimum and maximum desired temperatures. Note that, as long as constraint (4a) is satisfied, the total available HTF can be unevenly distributed among the set of loops, e.g., higher flow rates can be pumped to loops receiving greater irradiance. Finally, the proposed controller should be scalable and approximate the optimal performance with reduced computational and communication burden.

C. Centralized MPC problem

In what follows, consider a discrete-time setting, let $\Delta t^{\rm s}$ be the integration step size, and let k be the discrete time index, i.e., step k refers to instant $k\Delta t^{\rm s}$. Likewise, let $\Delta t^{\rm c} = \delta^{\rm c} \Delta t^{\rm s}$ be the sampling time considered in the control models, where $\delta^{\rm c} \in \mathbb{N}^+$. Then, the centralized MPC problem underlying this article can be formulated as follows:

$$\min_{[\mathbf{q}_i(k)]_{i\in\mathcal{N}}} \sum_{i\in\mathcal{N}} \sum_{n\in\mathcal{H}} \left(w_{\rm e} e_i^2(n + \delta^{\rm c}|k) + w_{\rm q} q_i^2(n|k) \right)$$

s.t.

$$\begin{aligned} T_i^{\rm out}(n + \delta^{\rm c}|k) &= T_i^{\rm out}(n|k) \\ &+ \frac{\Delta t^{\rm c}}{C_i(n|k)} \Big(\eta_i(k) \mathscr{I}_i(k) - \hbar_i(n|k) \Big) \\ &- \frac{\Delta t^{\rm c}}{C_i(n|k)} q_i(n|k) P_i(n|k) \left(T_i^{\rm out}(n|k) - T^{\rm in}(k) \right), \end{aligned} \qquad (5a)$$

$$T_i^{\rm out}(k|k) = T_i^{\rm out}(k), \qquad (5b)$$

$$T^{\rm min} \leq T_i^{\rm out}(n + \delta^{\rm c}|k) \leq T^{\rm max}, \qquad (5c)$$

$$q^{\rm min} \leq q_i(n|k) \leq q^{\rm max}, \qquad (5d)$$

$$\sum\nolimits_{l\in\mathcal{N}} q_l(n|k) \leq Q_{\rm T}, \qquad (5e)$$

$$\forall i \in \mathcal{N}, \ \forall n \in \mathcal{H}, \qquad (5f)$$

where $e_i(n + \delta^{\rm c}|k) = T_i^{\rm out}(n+\delta^{\rm c}|k) - T^{\rm ref}(n+\delta^{\rm c})$ denotes the outlet temperature error of loop i, with $T^{\rm ref}(\cdot)$ being the reference temperature. Also, $\mathcal{H} = \{k, k + \delta^{\rm c}, k + 2\delta^{\rm c}, ..., k + \delta^{\rm c} N_{\rm p}\}$ is the set of time instants considered in the prediction horizon, with $N_{\rm p}$ being a tuning parameter, $\mathbf{q}_i(k) = [q_i(k|k), q_i(k + \delta^{\rm c}|k), ..., q_i(k + \delta^{\rm c} N_{\rm p}|k)]^\top$ is the flow rate sequence of loop i, and $w_{\rm e}$ and $w_{\rm q}$ are positive

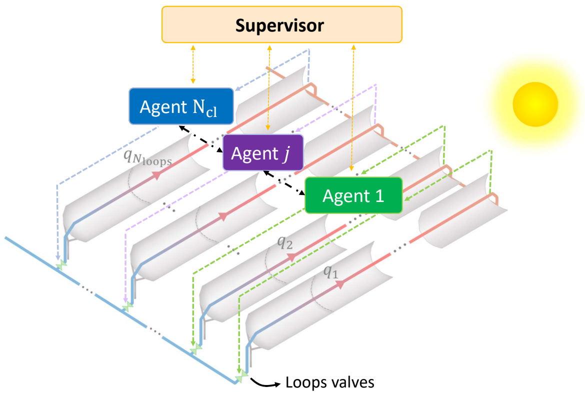

Fig. 1. Architecture of the proposed control approach. The agents control the flows in different clusters of loops, e.g., agent 1 controls loops 1 and 2.

definite weighting scalars. Likewise, (5a) is a discrete-time version of model (1), where $P_i(n|k)$, $C_i(n|k)$ and $\hbar_i(n|k)$ are computed considering (2) and the predicted mean temperature $T_i^{\mathrm{m}}(n|k) = (T_i^{\mathrm{out}}(n|k) + T^{\mathrm{in}}(k))/2$. Note that, for the sake of simplicity, the inlet temperature, effective irradiance, and ambient temperature, are assumed to mantain their value at k during the entire prediction horizon. Finally, notice also that prediction model (5a) introduces nonconvex terms both in the cost function and in constraint (5c), making (5) a potentially nonconvex optimization problem.

III. Clustering-based DMPC using ALADIN

Centralized problem (5) may involve a large number of loops and lacks convexity guarantees as mentioned above. Considering this issue, this article proposes the distributed control architecture illustrated in Fig. 1, whose main features are the following:

(i) The set of N_{loops} loops are *dynamically* partitioned by a supervisor into a set of non-overlapping clusters $\{\mathcal{C}_1,\ \mathcal{C}_2,\ \ldots,\ \mathcal{C}_{N_{\mathrm{cl}}}\}$, such that $\bigcup_{j=1}^{N_{\mathrm{cl}}} \mathcal{C}_j = \mathcal{N}$, where $N_{\mathrm{cl}} \leq N_{\mathrm{loops}}$ denotes the number of clusters.
(ii) Each cluster $\mathcal{C}_j$ is assigned to MPC agent j, which controls flow q_i for all $i \in \mathcal{C}_j$ during a given time period.
(iii) The set of MPC agents coordinate their decisions to optimize their collective performance using ALADIN, which is designed to address (potentially nonconvex) distributed problems.

The next subsections provide further details regarding the partition selection and the proposed control algorithm.

A. Partition selection

Inspired by [19], our proposed DMPC approach exploits similarities between the loops to reduce the control problem complexity. In particular, the solar field is dynamically partitioned into clusters of loops whose dynamics are approximately characterized by the same parameters. To this end, mean temperature $T_i^{\mathrm{m}}(k)$ and current effective irradiance $\eta_i(k)\mathscr{I}_i(k)$ are periodically collected for all $i \in \mathcal{N}$ so that we build the following data set:

$$\mathcal{D}(k) = \{[\eta_i(k)\mathscr{I}_i(k),\quad T_i^{\mathrm{m}}(k)]\}_{i\in\mathcal{N}}. \qquad (6)$$

Note that, given (5a), those loops for which these two features are equal will have identical prediction models. Using clustering methods, the loops in $\mathcal{N}$ can then be partitioned into a number of clusters, say $N_{\mathrm{cl}}(k) \leq N_{\mathrm{cl}}^{\max}$, according to the data in $\mathcal{D}(k)$. In this respect, $N_{\mathrm{cl}}^{\max}$ denotes the maximum number of clusters, which is directly related with the number of MPC agents available in the system. Without loss of generality, we consider the well-known centroid-based algorithm K-means [20], together with the *elbow* method to select the optimal number of clusters.

B. Clusters-based MPC problem

Let $\mathcal{P}(k) = \{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_{N_{\mathrm{cl}}(k)}\}$ be the partition selected at time k as described above. Then, we consider the following MPC problem to find the HTF to be pumped to every cluster:

$$\min_{[\mathbf{q}_{\mathcal{C}_j}(k)]_{\forall \mathcal{C}_j}} \sum_{j=1}^{N_{\mathrm{cl}}(k)} |\mathcal{C}_j| \sum_{n\in\mathcal{H}} \left(w_{\mathrm{e}} e_{\mathcal{C}_j}^2(n+\delta^{\mathrm{c}}|k) + w_{\mathrm{q}} q_{\mathcal{C}_j}^2(n|k)\right)$$

s.t.

$$\begin{aligned} &T_{\mathcal{C}_j}^{\mathrm{out}}(n+\delta^{\mathrm{c}}|k) = T_{\mathcal{C}_j}^{\mathrm{out}}(n|k) \\ &+ \frac{\Delta t^{\mathrm{c}}}{C_{\mathcal{C}_j}(n|k)}\left(\eta_{\mathcal{C}_j}(k)\mathscr{I}_{\mathcal{C}_j}(k) - \hbar_{\mathcal{C}_j}(n|k)\right) \\ &- \frac{\Delta t^{\mathrm{c}}}{C_{\mathcal{C}_j}(n|k)} q_{\mathcal{C}_j}(n|k) P_{\mathcal{C}_j}(n|k)\left(T_{\mathcal{C}_j}^{\mathrm{out}}(n|k) - T^{\mathrm{in}}(n|k)\right), \end{aligned} \qquad (7a)$$

$$T_{\mathcal{C}_j}^{\mathrm{out}}(k|k) = \sum_{i\in\mathcal{C}_j} (q_i(k-1) T_i^{\mathrm{out}}(k)) / \sum_{i\in\mathcal{C}_j} q_i(k-1), \qquad (7b)$$

$$T^{\min} \leq T_{\mathcal{C}_j}^{\mathrm{out}}(n+\delta^{\mathrm{c}}|k) \leq T^{\max}, \qquad (7c)$$

$$q_{\mathcal{C}_j}^{\min} \leq q_{\mathcal{C}_j}(n|k) \leq q_{\mathcal{C}_j}^{\max}, \qquad (7d)$$

$$\sum_{l=1}^{N_{\mathrm{cl}}(k)} q_{\mathcal{C}_l}(n|k) \leq Q_{\mathrm{T}}, \qquad (7e)$$

$$\forall j \in \{1, 2, \ldots, N_{\mathrm{cl}}(k)\},\ \forall n \in \mathcal{H}, \qquad (7f)$$

where $e_{\mathcal{C}_j}(n+\delta^{\mathrm{c}}|k) = T_{\mathcal{C}_j}^{\mathrm{out}}(n+\delta^{\mathrm{c}}|k) - T^{\mathrm{ref}}(n+1)$, and $\mathbf{q}_{\mathcal{C}_j}(k) = [q_{\mathcal{C}_j}(n|k)]_{n\in\mathcal{H}}$. Also, $q_{\mathcal{C}_j}^{\min} = |\mathcal{C}_j| q^{\min}$ and $q_{\mathcal{C}_j}^{\max} = |\mathcal{C}_j| q^{\max}$. Given the solution of (7), say $\mathbf{q}_{\mathcal{C}_j}^*(k)$ for all $\mathcal{C}_j \in \mathcal{P}(k)$, the HTF is uniformly distributed among the loops in every cluster. That is, the implemented flows are given by

$$q_i(t) = \frac{q_{\mathcal{C}_j}^*(k|k)}{|\mathcal{C}_j|},\ \forall i \in \mathcal{C}_j,\ \forall t \in [k, k+1, \ldots, k+\delta^{\mathrm{c}}). \qquad (8)$$

Remark 1. *Problem* (7) *has the same form as problem* (5) *but involves a reduced number of optimization variables. In particular, while the number of flow variables in* (5) *was* $N_{\mathrm{p}}N_{\mathrm{loops}}$*, here we deal with* $N_{\mathrm{p}}N_{\mathrm{cl}}(k)$*.*

Remark 2. *Since the clusters are chosen to aggregate loops with similar dynamics, the solution of* (7) *will approximate that of* (5)*. Particularly, we are replacing models of loops that are nearly identical with a single lumped description. This similarity among loops also motivates the uniform flow allocation indicated in* (8)*.*

1) Formulation to use ALADIN: As detailed in [15], ALADIN is designed to solve optimization problems with separable (potentially nonconvex) objective functions, decoupled inequality constraints, and coupled affine equality constraints.

Note that, by definition, the objective function in (7) is separable and can indeed be rewritten as

$$\sum_{j=1}^{N_{\mathrm{cl}}(k)} \underbrace{|\mathcal{C}_j| \sum_{n\in\mathcal{H}} (w_{\mathrm{e}} e^2_{\mathcal{C}_j}(n+\delta^{\mathrm{c}}|k) + w_{\mathrm{q}} q^2_{\mathcal{C}_j}(n|k)).}_{f_{\mathcal{C}_j}(\mathbf{e}_{\mathcal{C}_j}(k),\mathbf{q}_{\mathcal{C}_j}(k))}$$

Likewise, given (7a), variables $T^{\mathrm{out}}_{\mathcal{C}_j}(k+\kappa\delta^{\mathrm{c}}|k)$ for any $\kappa \in \{1,2,...,N_{\mathrm{p}}\}$ can be computed as

$$\begin{aligned} &T^{\mathrm{out}}_{\mathcal{C}_j}(k+\kappa\delta^{\mathrm{c}}|k) = T^{\mathrm{out}}_{\mathcal{C}_j}(k)\\ &+ \sum_{\tilde{n}=k}^{k+(\kappa-1)\delta^{\mathrm{c}}} \frac{\Delta t^{\mathrm{c}}}{C_{\mathcal{C}_j}(\tilde{n}|k)} \Big(\eta_{\mathcal{C}_j}(k)\mathscr{I}_{\mathcal{C}_j}(k) - \hbar_{\mathcal{C}_j}(\tilde{n}|k)\Big)\\ &- \sum_{\tilde{n}=k}^{k+(\kappa-1)\delta^{\mathrm{c}}} \frac{\Delta t^{\mathrm{c}}}{C_{\mathcal{C}_j}(\tilde{n}|k)} q_{\mathcal{C}_j}(\tilde{n}|k) P_{\mathcal{C}_j}(\tilde{n}|k) \Big(T^{\mathrm{out}}_{\mathcal{C}_j}(\tilde{n}|k) - T^{\mathrm{in}}(k)\Big). \end{aligned}$$

That is, they depend on $T^{\mathrm{out}}_{\mathcal{C}_j}(k)$, $T^{\mathrm{in}}(k)$, $T^{\mathrm{a}}(k)$, $\eta_{\mathcal{C}_j}(k)\mathscr{I}_{\mathcal{C}_j}(k)$, and on the sequence of flow rates from instant k up to $k+(\kappa-1)\delta^{\mathrm{c}}$. Then, the objective function in (7) can be written as a function of $z_{\mathcal{C}_j}(k) = [T^{\mathrm{out}}_{\mathcal{C}_j}(k), \eta_{\mathcal{C}_j}(k)\mathscr{I}_{\mathcal{C}_j}(k), T^{\mathrm{in}}(k), T^{\mathrm{a}}(k)]$, $\mathbf{T}^{\mathrm{ref}}(k) = [T^{\mathrm{ref}}(n+\delta^{\mathrm{c}})]_{n\in\mathcal{H}}$, and $\mathbf{q}_{\mathcal{C}_j}(k)$. Similarly, constraint (7c) is of the form $h_{\mathcal{C}_j}(z_{\mathcal{C}_j}(k), \mathbf{q}_{\mathcal{C}_j}(k)) \leq \mathbf{0}_{N_{\mathrm{p}}}$, where $h_{\mathcal{C}_j}(\cdot)$ is the corresponding constraint function. Finally, let us introduce a *sink* artificial loop, say loop 0, and define $\mathcal{C}_0 = \{0\}$ to keep the notation simple. Then, problem (7) can be reformulated as follows:

$$\min_{[\mathbf{q}_{\mathcal{C}_j}(k)]_{j=0}^{N_{\mathrm{cl}}(k)}} \sum_{j=1}^{N_{\mathrm{cl}}(k)} f_{\mathcal{C}_j}(z_{\mathcal{C}_j}(k), \mathbf{T}^{\mathrm{ref}}(k), \mathbf{q}_{\mathcal{C}_j}(k)) + f_{\mathcal{C}_0}(\mathbf{q}_{\mathcal{C}_0}(k)))$$

$$\begin{aligned} \text{s.t.}\quad & h_{\mathcal{C}_j}(z_{\mathcal{C}_j}(k), \mathbf{q}_{\mathcal{C}_j}(k)) \leq 0, \quad \forall \mathcal{C}_j \in \mathcal{P}(k), && \text{(9a)}\\ & q^{\min}_{\mathcal{C}_j}\mathbf{1}_{N_{\mathrm{p}}} \leq \mathbf{q}_{\mathcal{C}_j}(k) \leq q^{\max}_{\mathcal{C}_j}\mathbf{1}_{N_{\mathrm{p}}}, \quad \forall \mathcal{C}_j \in \mathcal{P}(k), && \text{(9b)}\\ & \mathbf{q}_{\mathcal{C}_0}(k) \geq \mathbf{0}_{N_{\mathrm{p}}}, && \text{(9c)}\\ & \sum\nolimits_{l=0}^{N_{\mathrm{cl}}(k)} \mathbf{q}_{\mathcal{C}_l}(k) = Q_{\mathrm{T}}\mathbf{1}_{N_{\mathrm{p}}}, && \text{(9d)} \end{aligned}$$

with $\mathbf{q}_{\mathcal{C}_0}(k)$ being the flow surplus over Q_{T} that the agents decide not to use. Likewise, $f_{\mathcal{C}_0}(\mathbf{q}_{\mathcal{C}_0}(k)))$ is a (possibly nonzero) cost associated with sending flow to the sink loop.

C. Distributed coordination using ALADIN

Problem (9) is an optimal resource allocation problem of the form of those that can be solved in a distributed manner by implementing ALADIN. This algorithm involves an iterative procedure that is briefly introduced below. Let subscript p enumerate the iterations, λ be the multiplier associated with constraint (9d), and consider some time step $k \in \{0, \delta^{\mathrm{c}}, 2\delta^{\mathrm{c}}, ...\}$. Also, consider a positive definite scaling matrix Σ, a termination tolerance ϵ, an initial guess for the primal variables $\mathbf{y}^0 = [\mathbf{y}^0_{\mathcal{C}_j}]_{j=0}^{N_{\mathrm{cl}}(k)}$, and some λ^0 and $\mu^0, \rho^0 > 0$. Then, flow sequences $\mathbf{q}_{\mathcal{C}_j}(k)$ for all $\mathcal{C}_j$ are computed by implementing the following steps starting from $p = 0$. See [15] and [16] for further details.

1. *Parallelizable decentralized step*: All agents $j \in \{1,2,...,N_{\mathrm{cl}}\}$ solve locally the following decoupled nonlinear problem:

$$\begin{aligned} \min_{\mathbf{q}_{\mathcal{C}_j}}\ & f_{\mathcal{C}_j}(z_{\mathcal{C}_j}, \mathbf{T}^{\mathrm{ref}}, \mathbf{q}_{\mathcal{C}_j}) + (\lambda^p)^\top \mathbf{q}_{\mathcal{C}_j} + \frac{\rho^p}{2}\|\mathbf{q}_{\mathcal{C}_j} - \mathbf{y}^p_{\mathcal{C}_j}\|^2_\Sigma\\ \text{s.t.}\ & h_{\mathcal{C}_j}(z_{\mathcal{C}_j}, \mathbf{q}_{\mathcal{C}_j}) \leq 0, && \text{(10a)}\\ & q^{\min}_{\mathcal{C}_j}\mathbf{1}_{N_{\mathrm{p}}} \leq \mathbf{q}_i \leq q^{\max}_{\mathcal{C}_j}\mathbf{1}_{N_{\mathrm{p}}}, && \text{(10b)} \end{aligned}$$

   where, for clarity, we have omitted time index k. For the sink artificial loop, we consider additional agent $j = 0$, which solves a similar problem considering $f_{\mathcal{C}_0}(\mathbf{q}_{\mathcal{C}_0})$.
2. Let $\mathbf{q}^p_{\mathcal{C}_j}$ be the solution of (10) for the j-th cluster. Then, if $\|\sum_{j=0}^{N_{\mathrm{cl}}} \mathbf{q}^p_{\mathcal{C}_j} - Q_{\mathrm{T}}\| \leq \epsilon$ and $\|\sum_{j=0}^{N_{\mathrm{cl}}}(\mathbf{q}^p_{\mathcal{C}_j} - \mathbf{y}^p_{\mathcal{C}_j})\| \leq \epsilon$, exit the algorithm.
3. *Sensitivity evaluations*: All agents j compute gradients $g^p_{\mathcal{C}_j} = \nabla f_{\mathcal{C}_j}(\cdot)$, a positive definite Hessian approximation $H^p_{\mathcal{C}_j}$, and constraints Jacobian $G^p_{\mathcal{C}_j}$ [15].
4. *Coordination step*: Solve the following overall quadratic program (QP):

$$\begin{aligned} \min_{s,\Delta\mathbf{q}}\ & \sum_{j=0}^{N_{\mathrm{cl}}} \left(\frac{1}{2}\|\Delta\mathbf{q}_{\mathcal{C}_j}\|^2_{H^p_{\mathcal{C}_j}} + (g^p_{\mathcal{C}_j})^\top \Delta\mathbf{q}_{\mathcal{C}_j}\right) + r(s, \lambda^p, \mu^p)\\ \text{s.t.}\ & \sum\nolimits_{j=0}^{N_{\mathrm{cl}}} (\mathbf{q}^p_{\mathcal{C}_j} + \Delta\mathbf{q}_{\mathcal{C}_j}) - Q_{\mathrm{T}} = s, && \text{(11a)}\\ & G^p_{\mathcal{C}_j}\Delta\mathbf{q}_{\mathcal{C}_j} = 0, \forall \mathcal{C}_j \in \mathcal{P}, && \text{(11b)} \end{aligned}$$

   where $\Delta\mathbf{q} = [\Delta\mathbf{q}_{\mathcal{C}_j}]_{j=0}^{N_{\mathrm{cl}}}$ and $r(\cdot) = \lambda^{p\top} s + \mu^p/2\|s\|^2$.
5. Finally, update the primal and dual variables as follows:

$$\begin{aligned} \mathbf{y}^{p+1} &= \mathbf{y}^p + \beta^p_1(\mathbf{q}^p - \mathbf{y}^p) + \beta^p_2 \Delta\mathbf{q}^p,\\ \lambda^{p+1} &= \lambda^p + \beta^p_3(\lambda^p_{\mathrm{QP}} - \lambda^p), \end{aligned} \tag{12}$$

   where $\mathbf{q}^p = [\mathbf{q}^p_{\mathcal{C}_j}]_{j=0}^{N_{\mathrm{cl}}}$, $\Delta\mathbf{q}^p$ is obtained from the solution of (11), and λ^p_{QP} is the multiplier associated with constraint (11a). Likewise, factors β_1, β_2, and β_3 are computed following [15].

D. Pseudocode

Finally, the pseudocode of the proposed algorithm is summarized in Algorithm 1. Recall that Δt^{c} is the control time step and that the system is simulated using a discrete-time version of (1) for all $i \in \mathcal{N}$, where the integration step size is Δt^{s}. Likewise, the inlet temperature dynamics are modeled considering the following transfer function:

$$\frac{T^{\mathrm{in}}(s)}{T^{\mathrm{out}}(s) - 80} = \frac{1}{600s+1}, \tag{13}$$

where T^{out} is the overall outlet temperature of the solar field, and $T^{\mathrm{out}} - 80$ºC approximates the outlet temperature of the steam generator. In this regard, for all instants k, we consider $T^{\mathrm{out}}(k) = \sum_{i\in\mathcal{N}} q_i(k-1)T^{\mathrm{out}}_i(k) / \sum_{i\in\mathcal{N}} q_i(k-1)$.

Algorithm 1 Control algorithm

Define a maximum number of clusters $N_{\mathrm{cl}}^{\max}$, an initial partition $\mathcal{P}(0)=\{\mathcal{C}_1,\dots,\mathcal{C}_{N_{\mathrm{cl}}(0)}\}$, with $N_{\mathrm{cl}}(0)\le N_{\mathrm{cl}}^{\max}$, and let the partition be updated every $\Delta t^{\mathrm{cl}}=\delta^{\mathrm{cl}}\Delta t^{\mathrm{s}}$. Also, assign each $\mathcal{C}_j$ to agent j, and the sink loop to agent 0. Then, at all instants k, proceed as follows:

```
if k ∈ {0, δ^c, 2δ^c, ...} then
    if k ∈ {δ^cl, 2δ^cl, ...} then
        Update the clusters as described in Section III-A
        and define partition P(k) = {C_1, ..., C_{N_cl(k)}}.
    else
        Set P(k) ← P(k − 1).
    end if
    All MPC agents j ∈ {0, 1, ..., N^cl(k)} solve problem (7) in a distributed
    manner by using ALADIN algorithm as described in Section III-C. As a solution,
    the agents find the flow rates to be pumped to each cluster C_j during
    interval [k, k + δ^c).
    For each cluster C_j, define q*_i(t) = q*_{C_j}(k|k)/|C_j| for
    all i ∈ C_j and t ∈ [k, k + δ^c).
end if
Implement flow rates q*_i(k) for all loops i ∈ N.
```

IV. Simulation Results

In this section, we simulate Algorithm 1 on a 10-loop solar parabolic plant with different values of $N_{\mathrm{cl}}^{\max}$ and Δt^{cl}, and considering the parameters in Table I. All simulations were carried out in a 1.8 GHz Intel® Core™ i7/16GB RAM computer using Matlab®, software CasADi [21], and toolbox ALADIN-α [16]. Also, we used `ipopt` and `MA57` for solving (10) and (11), respectively. The partitions were found using function `kmeans` with the Calinski-Harabasz index.

As a reference, the results are compared with those obtained considering statically the *finest* and *coarsest* partition of the system. The former corresponds to running Algorithm 1 with initial singleton partition $\mathcal{P}(0)=\{\{1\},\{2\},\dots,\{10\}\}$ and $\Delta t^{\mathrm{cl}}=\infty$. By contrast, the latter corresponds to $\mathcal{P}(0)=\{1,2,\dots,10\}$ and $\Delta t^{\mathrm{cl}}=\infty$, i.e., a single controller uses a lumped parameter model of the entire solar field and distributes equally the flow among all loops. For the sake of clarity, these two approaches will be denoted as DMPC$_{\mathrm{fin}}$ and MPC$_{\mathrm{coar}}$, respectively.

The simulations consider a 7-hour period of a cloudy day in which the irradiance and ambient temperature evolve as in [19, Fig. 3]. The outlet temperatures and flows evolution are shown in Fig. 2 (top) for $N_{\mathrm{cl}}^{\max}$=5 and Δt^{cl}=2.5 min.

TABLE I

PARAMETERS USED IN THE SIMULATIONS

	Value	Unit		Value	Unit
$q^{\min}$, $q^{\max}$	0.2, 2	l/s	Δt^{s}, Δt^{c}	0.5, 30	s
$T^{\min}$, $T^{\max}$	220, 305	°C	w_{e}, w_{q}	0.001, 1	-
A	$5.067{\cdot}10^{-4}$	m^2	N_{p}	5	-
L	142	m	ϵ	$1\cdot10^{-5}$	-
S	267.4	m^2	Q_{T}	9	l/s

As can be seen, the loops outlet temperatures follow closely the reference, and the flows decrease as the irradiance falls. However, the system performance underwent a significant deterioration when using MPC$_{\mathrm{coar}}$ (Fig. 2 (bottom)). Note that in the latter case all loops receive the same flow, and hence there is no chance of adjusting it to the space-varying conditions. Particularly, given (8), the maximum flow that the loops can get with MPC$_{\mathrm{coar}}$ is $Q_{\mathrm{T}}/10=0.9$ l/s, whereas in the DMPC case we obtained $\max_{i,k} q_i(k)=0.95$ l/s. Also, if the overall outlet temperature approaches $T^{\max}$, controller MPC$_{\mathrm{coar}}$ increases the flow in all loops, which decreases the temperature even of those that were below the reference.

The system performance is also numerically compared in Table II, which provides the cumulative costs in different simulations, i.e., $J_{\mathrm{cc}}=\sum_{k\in\mathcal{K}}\sum_{i=1}^{10}\left(w_{\mathrm{e}}e_i^2(k)+w_{\mathrm{q}}q_i^2(k)\right)$, together with the maximum incurred temperature errors, i.e.,

$$\bar{e} \;=\; \max_{k\in\tilde{\mathcal{K}},i\in\{1,\dots,10\}} |T_i^{\mathrm{out}}(k)-T^{\mathrm{ref}}(k)|.$$

Above, $\mathcal{K}$ represents the set of all simulated time instants, and $\tilde{\mathcal{K}}\subset\mathcal{K}$ contains the instants after the first 5 minutes. Note that $\tilde{\mathcal{K}}$ is used not to account for the errors at the beginning of the simulations, which are mainly influenced by the choice of the initial state. In addition, Table II indicates the mean number of loops per cluster. As expected, finer partitions and reduced Δt^{cl} resulted in better performance. Note also that the temperature errors could be reduced if accurate irradiance estimations are available.

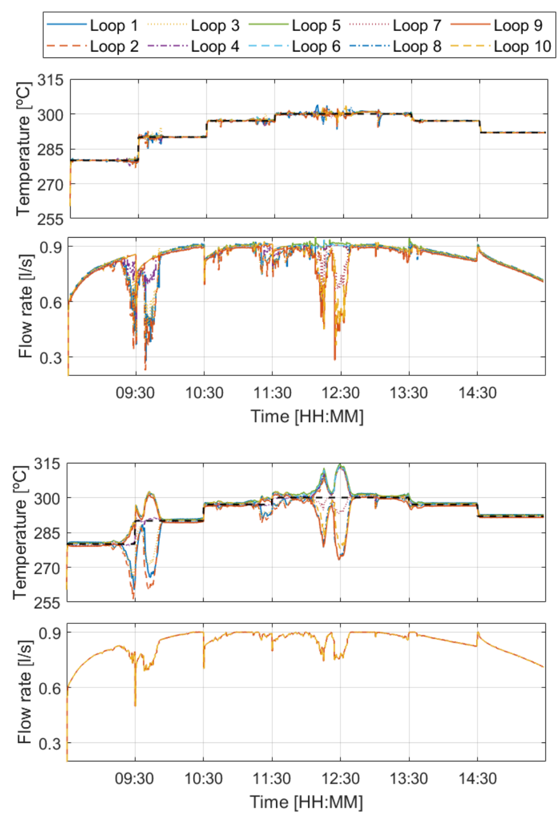

Fig. 2. Loops outlet temperature and HTF flow rate using the proposed DMPC with $N_{\mathrm{cl}}^{\max}=5$ and $\Delta t^{\mathrm{cl}}=2.5$ min (top), and using MPC$_{\mathrm{coar}}$ (bottom). The dashed black lines indicate the reference temperature.

TABLE II

CUMULATIVE PERFORMANCE COSTS AND CLUSTERS SIZE

			J_{cc}	$\bar{e}$	Mean no. of loops/cluster
Static part.	DMPC$_{\mathrm{fin}}$		172.08	9.84	1
	MPC$_{\mathrm{coar}}$		$8.22{\cdot}10^3$	29.83	10
Time-varying partition	$N_{\mathrm{cl}}^{\max}$	Δt^{cl} [min]			
	8	1.5	176.89	9.86	1.39
	6	1.5	195.02	10.23	2.03
	5	1.5	215.44	10.60	2.51
	5	2.5	228.59	10.63	2.53
	3	5.0	749.88	19.28	4.15

Regarding the computation times, Fig. 3 shows the values of the following indexes:

$$\bar{\tau}^{\mathrm{NLP}} = \sum_{k\in\mathcal{K}^{\mathrm{c}}} \sum_{j=1}^{N_{\mathrm{cl}}(k)} \tau_{\mathcal{C}_j}^{\mathrm{NLP}}(k)/|\mathcal{K}^{\mathrm{c}}|, \quad \bar{\tau}^{\mathrm{QP}} = \sum_{k\in\mathcal{K}^{\mathrm{c}}} \tau^{\mathrm{QP}}(k)/|\mathcal{K}^{\mathrm{c}}|,$$

$$\bar{\tau}^{\mathrm{sum}} = \bar{\tau}^{\mathrm{NLP}} + \bar{\tau}^{\mathrm{QP}} + \sum_{k\in\mathcal{K}^{\mathrm{c}}} \sum_{j=1}^{N_{\mathrm{cl}}(k)} \tau_{\mathcal{C}_j}^{\mathrm{sens}}(k)/|\mathcal{K}^{\mathrm{c}}|,$$

which are associated with different steps of ALADIN algorithm. Above, $\tau_{\mathcal{C}_j}^{\mathrm{NLP}}(k)$ and $\tau_{\mathcal{C}_j}^{\mathrm{sens}}(k)$ denote respectively the time spent by agent j solving nonlinear problem (10) and computing the sensitives at time step k. In addition, $\tau^{\mathrm{QP}}(k)$ refers to the time spent solving QP (11), and $\mathcal{K}^{\mathrm{c}}$ is the set of instants in which the flow rates are updated. As reflected in Fig. 3, finer partitions involve a greater number of variables to coordinate, and led to higher computation times. Notice also that, although steps 1 and 3 of ALADIN can be performed in parallel, increasing the number of distributed agents also demands more communication links.

V. CONCLUSIONS

A DMPC with time-varying partitioning for optimizing the HTF flow rates in solar parabolic trough plants has been presented. In this regard, clustering methods are considered for dynamically partitioning the solar field into clusters of similar loops, which are subsequently assigned to a set of MPC agents. The article formulates the associated DMPC problem so that it can be addressed implementing ALADIN algorithm, and illustrates its effectiveness via simulations. In particular, it is shown that the proposed approach can closely approximate that of a DMPC with static finer partitions while reducing the number of variables to be coordinated. Future research will include a comparison with ADMM, as well as exploring bi-level ALADIN. Also, we will extend our results to larger plants, and consider the optimization of the setpoint so as to maximize the net electricity production.

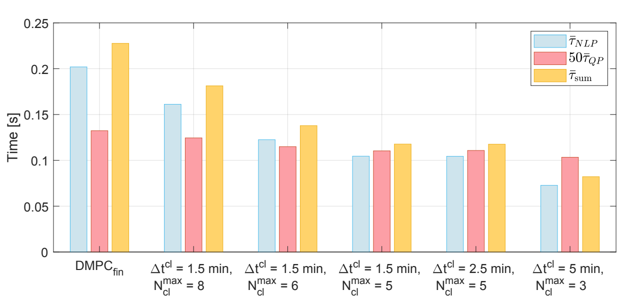

Fig. 3. Computation times of different steps of ALADIN algorithm. For sake of clarity, $\bar{\tau}^{\mathrm{QP}}$ is scaled by 50.

REFERENCES

[1] SolarPower Europe. (2022) Global market outlook. https://www.solarpowereurope.org/insights/market-outlooks/global-market-outlook-for-solar-power-2022.

[2] Renewable Energy Policy Network for the 21st Century (REN21). Renewables 2022: Global Status Report. https://www.ren21.net/reports/global-status-report.

[3] G. Alva, Y. Lin, and G. Fang, "An overview of thermal energy storage systems," *Energy*, vol. 144, pp. 341–378, 2018.

[4] A. B. Awan, M. Khan, M. Zubair, and E. Bellos, "Commercial parabolic trough CSP plants: Research trends and technological advancements," *Solar Energy*, vol. 211, pp. 1422–1458, 2020.

[5] E. F. Camacho, M. Berenguel, F. R. Rubio, and D. Martínez, *Control of Solar Energy Systems*. Springer London, 2012.

[6] S. J. Navas, F. R. Rubio, P. Ollero, and J. M. Lemos, "Optimal control applied to distributed solar collector fields with partial radiation," *Solar Energy*, vol. 159, pp. 811–819, 2018.

[7] T. Gholaminejad and A. Khaki-Sedigh, "Stable deep Koopman model predictive control for solar parabolic-trough collector field," *Renewable Energy*, vol. 198, pp. 492–504, 2022.

[8] M. Abutayeh, R. V. Padilla, M. Lake, Y. Y. Lim, J. Garcia, M. Sedighi, Y. C. S. Too, and K. Jeong, "Effect of short cloud shading on the performance of parabolic trough solar power plants: motorized vs manual valves," *Renewable energy*, vol. 142, pp. 330–344, 2019.

[9] A. J. Gallego, A. J. Sánchez, J. Escaño, and E. F. Camacho, "Nonlinear model predictive control for thermal balance in solar trough plants," *European Journal of Control*, vol. 67, p. 100717, 2022.

[10] Solana Generating Station. https://solarpaces.nrel.gov/project/solana-generating-station.

[11] J. R. D. Frejo and E. F. Camacho, "Centralized and distributed model predictive control for the maximization of the thermal power of solar parabolic-trough plants," *Solar Energy*, vol. 204, pp. 190–199, 2020.

[12] A. Sánchez-Amores, J. Martinez-Piazuelo, J. M. Maestre, C. Ocampo-Martinez, E. F. Camacho, and N. Quijano, "Coalitional model predictive control of parabolic-trough solar collector fields with population-dynamics assistance," *Applied Energy*, vol. 334, p. 120740, 2023.

[13] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein *et al.*, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends® in Machine learning*, vol. 3, no. 1, pp. 1–122, 2011.

[14] P. Giselsson, M. D. Doan, T. Keviczky, B. De Schutter, and A. Rantzer, "Accelerated gradient methods and dual decomposition in distributed MPC," *Automatica*, vol. 49, no. 3, pp. 829–833, 2013.

[15] B. Houska, J. Frasch, and M. Diehl, "An augmented Lagrangian based algorithm for distributed nonconvex optimization," *SIAM Journal on Optimization*, vol. 26, no. 2, pp. 1101–1127, 2016.

[16] A. Engelmann, Y. Jiang, H. Benner, R. Ou, B. Houska, and T. Faulwasser, "ALADIN-α - An open-source MATLAB toolbox for distributed non-convex optimization," *Optimal Control Applications and Methods*, vol. 43, no. 1, pp. 4–22, 2022.

[17] A. Engelmann, Y. Jiang, T. Mühlpfordt, B. Houska, and T. Faulwasser, "Toward distributed OPF using ALADIN," *IEEE Transactions on Power Systems*, vol. 34, no. 1, pp. 584–594, 2018.

[18] Y. Jiang, P. Sauerteig, B. Houska, and K. Worthmann, "Distributed optimization using ALADIN for MPC in smart grids," *IEEE Trans. on Control Systems Technology*, vol. 29, no. 5, pp. 2142–2152, 2020.

[19] P. Chanfreut, J. M. Maestre, A. J. Gallego, A. M. Annaswamy, and E. F. Camacho, "Clustering-based model predictive control of solar parabolic trough plants," *Renewable Energy*, vol. 216, p. 118978, 2023.

[20] M. Ahmed, R. Seraj, and S. M. S. Islam, "The K-means algorithm: A comprehensive survey and performance evaluation," *Electronics*, vol. 9, no. 8, p. 1295, 2020.

[21] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, "CasADi – A software framework for nonlinear optimization and optimal control," *Mathematical Programming Computation*, vol. 11, no. 1, pp. 1–36, 2019.