Multi-Agent Coordination of Thermostatically Controlled Loads by Smart Power Sockets for Electric Demand Side Management

Mauro Franceschelli, *Member, IEEE*, Alessandro Pilloni, *Member, IEEE*, and Andrea Gasparri, *Senior Member, IEEE*

***Abstract*— This article presents a multi-agent control architecture and an online optimization method based on a dynamic average consensus to coordinate the power consumption of a large population of thermostatically controlled loads (TCLs). Our objective is to penalize peaks of power demand, smooth the load profile, and enable demand-side management. The proposed architecture and methods exploit only local measurements of power consumption via smart power sockets (SPSs) with no access to their internal temperature. No centralized aggregator of information is exploited, and agents preserve their privacy by cooperating anonymously only through consensus-based distributed estimation. The interactions among devices occur through an unstructured peer-to-peer (P2P) network over the Internet. Methods for parameter identification, state estimation, and mixed logical modeling of TCLs and SPSs are included. The architecture is designed from a multi-agent and plug-and-play perspective, in which existing household appliances can interact with each other in an urban environment. Finally, a novel low-cost testbed is proposed along with numerical tests and experimental validation.**

***Index Terms*— Distributed predictive control, electric demand-side management (DSM), multi-agent systems (MASs), online randomized optimization, thermostatically controlled loads (TCLs).**

I. Introduction

DEMAND-SIDE management (DSM) aims to manage the electric power demand to match baseload power generation, thus reducing the use of costly and polluting peaker power plants [1]. Although statistics may vary depending on the country, in the USA, the 40% of the total electric demand is due to residential consumption [2] of which the largest share is due to electric heating and cooling achieved by the so-called thermostatically controlled loads (TCLs), such as water heaters, freezers, radiators, and air conditioners.

Manuscript received December 6, 2019; accepted February 9, 2020. Manuscript received in final form February 11, 2020. This work was supported in part by the CoNetDomeSys Project, Italian Ministry of Research and Education, under call SIR 2014, under Grant RBSI14OF6H, and in part by the POR SARDEGNA FSE 2014–2020-Asse III, Azione 10.5.12, Avviso di chiamata per il finanziamento di Progetti di ricerca Anno 2017. Recommended by Associate Editor (C. Manzie). *(Corresponding author: Mauro Franceschelli.)*

Mauro Franceschelli and Alessandro Pilloni are with the Department of Electric and Electronic Engineering, Unviersity of Cagliari, 09123 Cagliari, Italy (e-mail: mauro.franceschelli@unica.it; alessandro.pilloni@unica.it).

Andrea Gasparri is with the Department of Engineering, University of Roma TRE, 00146 Rome, Italy (e-mail: gasparri@dia.uniroma3.it).

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCST.2020.2974181

The coordination of the power consumption of these TCLs during daily peaks of urban power consumption is crucial to reduce costs [3]. However, due to the time-correlation of consumers' power consumption combined with the penetration of volatile renewable generation have exposed the inadequacy of unidirectional DSM strategies [4].

To provide some flexibility in controlling the urban power demand, several strategies that focus on actively modulating the ON/OFF power consumption profiles of TCLs to provide ancillary services to the grid have been proposed, see [5]–[8].

A. Literature Review

Often in the literature of DSM, the keywords "distributed control" and "multi-agent" are associated with distributed decision making by agents consisting of smart homes or devices interacting with a centralized information aggregator which collects data and updates the local setpoints.

Within this framework, in [9], a DSM feedback scheme for air conditioners, where broadcasts of thermostat set-points offset changes to the TCLs, is proposed. This approach looks well suited for building automation applications since it requires readings from a common power distribution connection cabinet, where the total aggregate demand is measured.

In [10], the problem of optimally dispatching a set of energy resources is addressed as a convex optimization problem where the "ratio-consensus" algorithm enables the distributed decision-making process to occur in an unbalanced directed graph.

In [11], a distributed algorithm to control a network of TCLs to match, in real time, the aggregated power consumption of the population of TCLs with the predicted power supply is proposed. It consists of a consensus-based optimization of the sum of TCL temperature differences with respect to the desired reference. There, each TCL executes an instance of a consensus algorithm periodically to estimate the total TCLs' aggregate demand. Based on this estimation and on a centralized forecast of power consumption, the desired power consumption is assigned to each TCL.

Similarly, measurements on the TCL's internal temperatures are considered in [7]–[12], where a game-theoretic approach which optimizes a quadratic objective function of the TCLs' temperature profile is considered. The decision making is made by the TCLs and a centralized aggregator of information

computes average state values to be used as feedback. Finally, we mention [13] and [14], which instead dispense of the presence of any information aggregator. There, motivated by the peak-to-average (PAR) ratio minimization problem in a network of TCLs, a distributed optimization method to solve min–max problems characterized by local convex constraints are presented.

All mentioned strategies require costly "smart" versions of TCLs with advanced communication, actuation, and sensing capabilities that are not implemented on off-the-shelf TCLs. This increases the cost of the DSM infrastructure, which efficacy ultimately depends on the number of users.

Furthermore, the existence of centralized information aggregators renders the whole system vulnerable to denial of service (DoS) cyber-attacks. Moreover, issues with the privacy of the consumers whose consumption is continuously monitored need to be considered [15].

In this article, we propose a different framework where smart devices (agents), TCLs in particular, cooperate within a peer-to-peer (P2P) network autonomously and anonymously with a small set of neighboring agents in the network graph. The proposed method aims to exploit only local and asynchronous anonymous interactions among the agents to optimize trough their emergent behavior a global objective function defined by their power consumption. In particular, we chose an objective function that incentivizes the shaving of peak power consumption and reduction of electric load variations by the network of TCLs. Through the proposed framework, the power consumption of the network can be modulated, thus enabling the shaping of the electric load profile without any direct control action on any device or the sharing of power consumption information with a centralized coordinator.

The proposed method is paired with a multi-agent control architecture which is intended to exploit the cheapest possible hardware to enable cooperation among devices, i.e., a smart power socket (SPS), suitable to retrofit existing TCLs such as domestic water heaters, thus greatly reducing the cost of the infrastructure needed for the electric DSM program and therefore significantly reducing the cost of adoption by the users. This design choice makes the control problem more challenging because on each device the identification of the TCL dynamics, the estimation of its state on its control has to be carried out with only power consumption measurements and ON/OFF control capability. For instance, in [16], the problem of identifying the dynamics of the TCL was addressed by either considering its internal temperature measurable or at least assuming that the temperature range of the thermostat (maximum and minimum temperature) was known in absolute terms. In this article, we only exploit power consumption measurement to develop a real plug-and-play approach that does not require any system configuration by the user.

Since only randomized, asynchronous, and anonymous local interactions among neighboring agents are considered, our strategy has the advantages of being robust against DoS cyber-attacks since there is no centralized decision maker or information aggregator to provide a vulnerability to attack. In addition, the agents are coupled only by a dynamic average consensus estimation process, which is robust to switching network topologies, i.e., link failure, and do not need to be re-initialized if an agent leaves the network, i.e., node failure.

Finally, scalability for implementation on large-scale networks is ensured by exploiting local control actions that do not increase in complexity with the size of the network and no network-wide event synchronization is required.

Summarizing, the main contributions are as follows.

1) A multi-agent DSM architecture for the coordination of anonymous networks of TCL via SPS.
2) A method for model identification for TCLs based only on power consumption measurements.
3) An observer for the estimation of the TCL internal state based only on power consumption measurements.
4) A distributed, randomized, online optimization method for the cooperative constrained optimization of the power consumption by the network of TCLs controlled by SPSs.
5) A low-cost experimental testbed based on off-the-shelf hardware and purpose-built software.
6) An experimental validation of the proposed method.

B. Structure of the Article

In Section II, a model of a TCL and SPS is presented together with the adopted notation. In Section III the considered DSM problem is formulated. Section IV presents the details of proposed DSM architecture. In Section V the proposed online distributed optimization method is presented and some of its convergence properties are characterized. Testbed and experiments are discussed in Section VI. Concluding remarks are in Section VII.

II. Modeling of Thermostatically Controlled Loads and Smart Power Sockets

Consider a multi-agent system (MAS) consisting of a population $\mathcal{V} = \{1, \ldots, n\}$ of agents. Each agent "i" consists of a TCL with its power outlet plugged into an SPS adapter.

SPSs are provided with processing and WiFi communication capabilities. Each SPS is connected to a P2P network over the Internet. At time $t_k \in \mathbb{R}^+$, the network topology is described by the graph $\mathcal{G}(t_k) = (\mathcal{V}, \mathcal{E}(t_k))$, where $\mathcal{E}(t_k) \subseteq \{\mathcal{V} \times \mathcal{V}\}$ is the set of active links. Each socket can monitor the active power $p_i(t_k) \in \mathbb{R}^+$ absorbed by the associated TCL, while enabling the power supply to the TCL by actuating its own switch state $s^i(t_k) \in \{0, 1\}$ to ON, i.e., $s^i(t_k) = 1$. Otherwise $s^i(t_k) = 0$ forces the TCL to be OFF as well, which further implies $p_i(t_k) = 0$.

Let $\Delta\tau = t_{k+1} - t_k > 0$ be the constant sampling interval between two time instants, and following [9]–[14], the TCL dynamics is well approximated at discrete instants of time by:

$$T^i(t_{k+1}) = T^i(t_k) \cdot e^{-\alpha^i \Delta\tau} + \left(1 - e^{-\alpha^i \Delta\tau}\right)\left(T^i_\infty + \frac{q^i}{\alpha^i} u^i(t_k) + w(t_k)\right) \quad (1)$$

where $T^i \in \mathbb{R}^+$ denotes the thermostatically controlled temperature of the TCL. $\alpha^i > 0$ is the heat exchange coefficient with the operating environment, the temperature T^i_∞ of which

is assumed to be slowly varying. $w(t_k) \in \mathbb{R}$ models unknown disturbances, for instance, it may represent the temperature drop due to the refill process with cold water on an electric water heater after a water drawing event generated by the user.

Finally, $q^i > 0$ and $u^i(t_k) \in \{0, 1\}$ denote the nominal heat generated by the electric heating element and its actual state. In particular, since the SPS switch is connected in series with the TCL power supply, $u^i(t_k)$ can be rewritten as follows:

$$u^i(t_k) = s^i(t_k) \cdot h^i(t_k) \tag{2}$$

where $h^i(t_k) \in \{0, 1\}$ denotes the internal TCL thermostat relay control the value of which is updated according to

$$h^i(t_{k+1}) : \begin{cases} 0, & \text{if } T^i(t_k) \geq T^i_{\max} \\ h^i(t_k), & \text{if } T^i(t_k) \in \left(T^i_{\min}, T^i_{\max}\right) \\ 1, & \text{if } T^i(t_k) \leq T^i_{\min} \end{cases} \tag{3}$$

and which forces $T^i(t_k)$ to remain within a neighborhood of the operating range, with $T^i_{\min} < T^i_{\max}$. From (2), necessary condition to have $u^i = 1$ is $s^i = 1$, not vice-versa. Follows that the agent dynamics are not only hybrid and subjected to the nonconvex mixed-integer constraints (2) but also underactuated with respect to SPS switch control variable s^i.

Moreover, let $p^i > 0$ be the measured nominal amount of power absorbed by the TCL "i" when its heater element is ON. In the remainder, for simplicity's sake, its actual power consumption is approximated by

$$p^i(t_k) = p^i \cdot u^i(t_k) = p^i \cdot \left(s^i(t_k) \cdot h^i(t_k)\right). \tag{4}$$

In practice, the measurement of $p^i(t_k)$ is noisy. Moreover, unmodelled dynamics affect the heater response, thus making (3) no more instantaneous, as instead approximated, but much more similar to the step response of a high-order system with dominant poles. Thus, to evaluate $u^i(t_k) \in \{0, 1\}$ in accordance with (2) and (3), from $p^i(t_k)$, we consider

$$u^i(t_k) : \begin{cases} 1, & \text{if } p^i(t_k) > \epsilon^i_n \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

where $\epsilon^i_n > 0$ denotes a threshold to discriminate when the TCL is ON or OFF, in spite of the measurement noise and unmodelled actuator dynamics. Its value is chosen well above the noise measurement level when the TCL is OFF, and well below p^i in (4), when the TCL is ON, such that (5) meets the models (2) and (3). Experiments indicate as good as a threshold of 5 Watts. Finally, the total aggregated power demand associated with the population of TCL-plus-SPS agents is

$$\begin{aligned} P(t_k) &= \sum_{i \in \mathcal{V}} p^i(t_k) = \sum_{i \in \mathcal{V}} p^i \cdot u^i(t_k) \\ &= \sum_{i \in \mathcal{V}} p^i \cdot \left(s^i(t_k) \cdot h^i(t_k)\right). \end{aligned} \tag{6}$$

III. Multi-Agent Oriented Demand Side Management Coordination Objective

We propose a fully distributed coordination strategy, desynchronization. In which enables a population of TCL-plus-SPS agents to optimize, over a receding horizon time window $\tau(t_k) = [t_k + \Delta\tau, t_k + L\Delta\tau]$ of length $L\Delta\tau$ where parameter L is the number of time slots, a globally coupled quadratic cost

$$J(t_k) = \frac{1}{L} \sum_{\ell=1}^{L} (P_\ell(t_k))^2 = \frac{1}{L} \sum_{\ell=1}^{L} \left(\sum_{i \in \mathcal{V}} p_i \cdot u^i_\ell(t_k) \right)^2 \tag{7}$$

which penalizes the peaks of power demand while promoting the TCL consumption desynchronization. In (7)

$$P_\ell(t_k) = \sum_{i \in \mathcal{V}} p^i \cdot u^i_\ell(t_k) \tag{8}$$

denotes the prediction performed at time t_k on the power demand expected for the population of TCLs at the future time $t_{k+\ell} = t_k + \ell\Delta\tau$, $\ell = 1, 2, \ldots, L$.

Remark 1: The objective of (7) can be motivated by a scenario where the energy provider changes the electricity cost proportionally to the demand in real time, to smooth, and shave-off peaks in the power demand. Let $c_\ell(t_k) = \bar{c}_\ell P_\ell(t_k)$, $\bar{c}_\ell > 0$, be the cost predicted at time t_k for time $t_{k+\ell}$, then (7) reduces to $J(t_k) = (1/L) \sum_{\ell=1}^{L} c_\ell(t_k) P_\ell(t_k)$. ■

Remark 2: Since (7) is a sum over the receding horizon time window of the squared expected power consumption $P_\ell(t_k)$, the reduction in the PAR [17] of the power consumption in the network is expected. However, to prevent disservices to users, each TCL abides by the constraint $T^i(t_k) \in [T^i_{\min}, T^i_{\max}]$, which represents its desired temperature range. Thus, the averaged power demand by the network is not allowed to change significantly. It follows that the optimization of (7) incentives the shaving of consumption peaks while reducing the electric load variations with respect to the averaged demand, and thus, promotes, as a by-product, the TCLs' consumption desynchronization. ■

The network topology is modeled by a connected graph, which is unknown to the agents and possibly time varying. Moreover, to preserve the privacy, each agent has only access to information on a small set of anonymous neighboring agents, and the timing and order of their state updates is randomized.

It follows that (8) and, thus, also (7) are unknown to the agents. Thus, to enable their cooperation, the quantity (8) needs to be estimated in a distributed fashion. We further point out that $J(t_k)$ is not a separable function and that the local optimization constraints are mixed-integer linear, and thus, nonconvex.

Finally, since TCLs are retrofitted by means of an SPS, the SPS switch can override only the thermostat ON state by turning it OFF, not vice-versa, see [9]–[14] where the TCLs are fully actuated.

Despite these challenges, we provide a method to enable each agent to optimize through local interactions the objective (7) by controlling the planning schedule for its own SPS switch

$$\mathbf{s}^i(t_k) = \left[s^i_1 \cdots s^i_\ell \cdots s^i_L\right]^\mathsf{T} = \left[s^i(t_{k+1}) \cdots s^i(t_{k+L})\right]^\mathsf{T} \in \{0, 1\}^L. \tag{9}$$

IV. Proposed Multi-Agent Control Architecture

The proposed multi-agent control architecture consists of: 1) methods to identify and observe the agent's dynamics by measuring only power consumption; 2) modeling of the agents as mixed logic dynamic (MLD) systems; and 3) the TCL cooperation protocol (Algorithm 1).

A. Method for TCL Power Consumption Model Identification

In our scenario, $T^i(t_k)$ and T^i_∞ are not available, and $T^i_{\min}$ and $T^i_{\max}$ are unknown. Thus, differently from [16], it is not possible to derive any temperature information in absolute terms. However, from (3), we observed that the falling (resp. raising) edges on $p^i(t_k)$ in (4) informs us that the $T^i(t_k)$ is equal to $T^i_{\max}$ (resp. $T^i_{\min}$) due to the thermostatic control. Since for our purposes it is sufficient to predict $u^i(t_{k+\ell})$ on the basis of the last time instant in which $u^i(t_k)$ switched ON/OFF, the estimation of $T^i(t_k)$ can be relaxed up to a scale and bias factor.

1) TCL Modeling for DSM Purposes: Let us consider a TCL located in an environment where the temperature T^i_∞ is slowly varying, and thus, approximated by a constant for our purpose. This assumption is reasonable in urban areas, where TCLs are often located indoor. Let us define a so-called "virtual temperature" as

$$y^i(t_k) = \beta^i \cdot \left(T^i(t_k) - T^i_\infty\right) \tag{10}$$

where $\beta^i > 0$ is an unknown parameter that linearly maps $T^i(t_k) \in [T^i_\infty, T^i_{\max}]$ to $y^i(t_k) \in [y^i_\infty = 0, y^i_{\max} = 1]$. By substituting (10) into (1), and by assuming, similarly to [16], the TCL operating in retention mode during the identification procedure, which means with no significant temperature drops due to exogenous disturbances [i.e., $w^i \approx 0$ in (1)], it yields

$$y^i(t_{k+1}) = y^i(t_k) \cdot e^{-\alpha^i \Delta\tau} + \frac{\beta^i q^i}{\alpha^i}\left(1 - e^{-\alpha^i \Delta\tau}\right) u^i(t_k). \tag{11}$$

Let us now define ΔT^i_{off} (resp. ΔT^i_{on}) as the discharge (resp. charge) interval of time to be waited to bring T^i from $T^i_{\max}$ (resp. $T^i_{\min}$) to $T^i_{\min}$ (resp. $T^i_{\max}$) when $w^i(t_k) = 0$, and $s^i(t_k) = 1 \ \forall \ t_k \geq 0$. Then, note that the observation of the falling (resp. raising) edges on $p^i(t_k)$ in (4) through the SPS, not only provide a measurement of ΔT^i_{off} (resp. ΔT^i_{on}) but it also gives information on that $T^i(t_k)$ and $y^i(t_k)$ reached, respectively, $T^i_{\max}$ and $y^i_{\max}$ (resp. $T^i_{\min}$ and $y^i_{\min}$).

2) Identification Procedure: The proposed method is based on the measurement of ΔT^i_{off} and ΔT^i_{on} to estimate the parameters of (11). If $T^i(t_k) = T^i_{\max}$, i.e., $y^i(t_k) = 1$) at $t_k = 0$, from (3), (11) reduces to

$$y^i(\Delta T^i_{\text{off}}) = e^{-\alpha^i \Delta T^i_{\text{off}}} = y^i_{\min} \implies \alpha^i = \frac{-\ln\left(y^i_{\min}\right)}{\Delta T^i_{\text{off}}}. \tag{12}$$

On the contrary, if $T^i(t_k) = T^i_{\min}$ at $t_k = 0$, from (3), it yields

$$y^i\left(\Delta T^i_{\text{on}}\right) = y^i_{\min} \cdot e^{-\alpha^i \Delta T^i_{\text{on}}} + \frac{\beta^i q^i}{\alpha^i}\left(1 - e^{-\alpha^i \Delta T^i_{\text{on}}}\right) = 1. \tag{13}$$

Then, by substituting (12) into (13), we have

$$\eta^i = \frac{\beta^i q^i}{\alpha^i} = \frac{1 - y^i_{\min} \cdot e^{-\alpha^i \Delta T^i_{\text{on}}}}{1 - e^{-\alpha^i \Delta T^i_{\text{on}}}} = \frac{1 - y^i_{\min} \cdot e^{\ln(y^i_{\min}) \frac{\Delta T^i_{\text{on}}}{\Delta T^i_{\text{off}}}}}{1 - e^{\ln(y^i_{\min}) \frac{\Delta T^i_{\text{on}}}{\Delta T^i_{\text{off}}}}}. \tag{14}$$

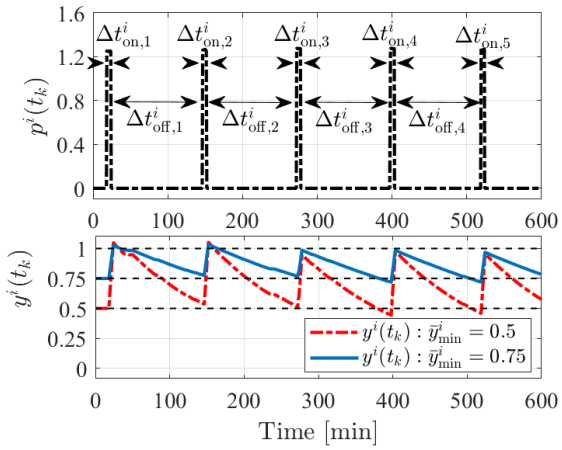

Fig. 1. Top: Consumption data set of 10 h of a real water heater with $p^i \approx 1.25$ kW acquired at night, when water drawing events are less likely. Bottom: output of model (11) corresponding to the power consumption shown at the top, with the parameters estimated with two different choices of $\bar{y}^i_{\min}$.

It follows that, since the parameter $y^i_{\min}$ is neither known nor available from any measurement, the set of equations (12) and (14) has infinitely many solutions. Nonetheless, (12), (14) admits a unique solution for each value of $\bar{y}^i_{\min} \in (0, 1)$. Let $\bar{\alpha}^i$ and $\bar{\eta}^i = \beta^i q^i / \bar{\alpha}^i$ be the solution of (12), (14) for $y^i_{\min} = \bar{y}^i_{\min}$, then the triple $(\bar{y}^i_{\min}, \bar{\alpha}^i, \bar{\eta}^i)$ completely describe the states of (11), which, if correctly initialized, provides a correspondence between the falling (resp. raising) edges of a measured sequence of $u^i(t_k)$ used as input in (11), and the fact that $y^i(t_k)$ and, thus, $T^i(t_k)$ reach, with a given rate due to ΔT^i_{off} and ΔT^i_{off}, the values of $\bar{y}^i_{\min}$ and $T^i_{\min}$ (resp. 1 and $T^i_{\max}$).

Remark 3: The parameters of (1) cannot be estimated with no information on $T^i_{\min}$, $T^i_{\max}$, and T^i_∞. On the contrary, the parameters of the model (11) can be estimated by fixing an arbitrary value of $y^i_{\min}$. The choice of $\bar{y}^i_{\min}$ does not affect the duration of the predicted charge/discharge time cycles of $y^i(t_k)$, which correspond to those of $T^i(t_k)$, due to (12) and (14). If at least two of the three absolute temperatures $T^i_{\min}$, $T^i_{\max}$, and T^i_∞ were available, the parameters of (1) could be also estimated. ■

Remark 4: The identification strategy does not require knowledge of β^i, which is unknown and constant, in accordance with the assumption of a slow change of T^i_∞. This is reasonable since, in urban areas, most of the TCLs are located indoor. However, for long-term operations, it could be necessary to update the parameters to account for the drift of T^i_∞ due to the seasonal changes. Finally, since α^i is positive, then (11) is bounded-input bounded-output stable. ■

Due to unmodelled dynamics such as convection, or disturbances, a single observation of an ON–OFF cycle is not a robust measure to estimate ΔT^i_{off} and ΔT^i_{on}. Let $\Delta T^i_{\text{on},k}$ and $\Delta T^i_{\text{off},k}$, accordingly with the top plot of Fig. 1, be the kth pair of ON–OFF time intervals of a data set of consumption acquired for identification purposes, experimental tests indicate that four or five cycles of operation in retention mode is enough to obtain reasonable statistical information through the observed power consumption.

Experimental Validation: On the bottom of Fig. 1, it is shown that the forced response of model (11) fed with the

TABLE I
DESCRIPTION OF DISCRETE EVENTS OF THE HYBRID VIRTUAL TEMPERATURE OBSERVER SHOWN IN FIG. 2

Events	Triggering conditions	Effects on continuous states
E_{off}	$\left(s^i(t_{k-2})=1\right) \wedge \left(s^i(t_{k-1})=1\right) \wedge \left(s^i(t_k)=1\right) \wedge \left(u^i(t_{k-1})=1\right) \wedge \left(u^i(t_k)=0\right)$	$y^i(t_k) := y^i_{\max}\ ,\ T^i_r := 0$
E_{on}	$\left(s^i(t_{k-2})=1\right) \wedge \left(s^i(t_{k-1})=1\right) \wedge \left(s^i(t_k)=1\right) \wedge \left(u^i(t_{k-1})=0\right) \wedge \left(u^i(t_k)=1\right)$	$y^i(t_k) := y^i_{\min}\ ,\ T^i_r := 0$
Timeout	$T^i_r \geq \bar{T}^i_r$	none
Out of Bound 1	$y^i(t_k) < \beta_1 \cdot y^i_{\min}$	$y^i(t_k) := \beta_1 \cdot y^i_{\min}$
Out of Bound 2	$y^i(t_k) > \beta_2 \cdot y^i_{\max}$	$y^i(t_k) := \beta_2 \cdot y^i_{\max}$

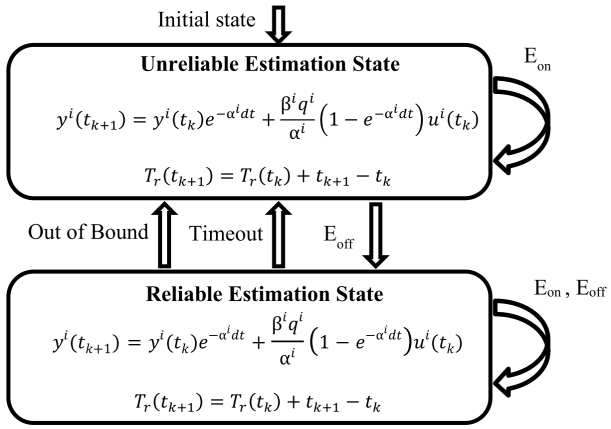

Fig. 2. Local hybrid virtual temperature observer.

normalized measured power consumption $u^i(t_k)$, shown at top of Fig. 1. The model is initialized at $\bar{y}^i_{\min}$ in correspondence of the first rising edge of $p^i(t_k)$. Parameters are estimated with (12) and (14), and the median value of the measured intervals $\Delta T^i_{\text{on},k}$ and $\Delta T^i_{\text{off},k}$, shown at the top of Fig. 1, for two different values of $\bar{y}^i_{\min}$. This test shows a good matching between the falling (resp. raising) edges of $p^i(t_k)$ and the fact that the virtual temperature $y^i(t_k)$ reaches coherently the value of $\bar{y}^i_{\min}$ (resp. 1), with a good precision. It can be further noted that the choice of $\bar{y}^i_{\min}$ does not affect the duration of the charge–discharge cycle. Thus, model (1), if correctly initialized enables the prediction the future TCL power consumption.

B. Hybrid Virtual Temperature Observer

Here, it is presented an observer enabling the estimation of the virtual temperature $y^i(t_k)$ on the basis of the measurement of power consumption, which is related to the thermostatic control input to the system. The proposed observer, thus, estimates $y^i(t_k)$ based on its open-loop dynamics, previously identified, and discrete events related to changes in the state of the thermostatic control, which periodically reset to zero its estimation error.

The block diagram of the proposed observer is shown in Fig. 2. It consists of a hybrid system with a continuous state $y^i(t_k)$ representing the estimated virtual temperature, a timer $T^i_r(t_k)$, and two discrete states, referred as "reliable estimation state" (RE) and "unreliable estimation state" (UE).

This distinction has been introduced to provide robustness on the estimation process.

The "RE" state identifies a state where the power consumption of the device is in agreement with the identified open-loop dynamics of the virtual temperature and, therefore, an estimation of the internal state of the TCL can be carried out based on the model.

The "UE" state identifies a state where the measured power consumption does not agree with the predicted consumption, and therefore, the open-loop TCL dynamics can not be used to carry out an estimation of the virtual temperature because significant and unknown perturbations to its temperature occurred. The "UE" state ends only when the event E_{off} is observed and the actual state of the device can be inferred exactly, thus changing the state to a "RE" state and enabling a prediction of future power consumption.

In fact, since both s^i and u^i play the role of inputs in (1), and w^i is unknown, no robust state feedback asymptotic observation strategies can be designed, and only open-loop estimation can be performed. Let $\wedge$ be the "AND" logical operator, the events between the discrete states and their set/reset effects on $y^i(t_k)$ and T^i_r are listed in Table I.

At initialization, the observer is set to the UE state. By (2) and (3), whenever the TCL "autonomously" changes state from $u^i(t_{k-1}) = 1$ to $u^i(t_k) = 0$, thus, without the switch to zero of $s^i(t_k)$, it can be inferred that $T^i(t_k)$ has reached $T^i_{\max}$ which corresponds to $y^i(t_k) = 1$. Similarly, when it changes autonomously from $u^i(t_{k-1}) = 0$ to $u^i(t_k) = 1$, $y^i(t_k) = \bar{y}^i_{\min}$. In accordance with Table I these events are "E_{off}" and "E_{on}."

We further refer to these events as "synchronization events" because, whenever triggered, the estimated virtual temperature is set to the corresponding correct value, thus resetting to zero any estimation error due to uncertainties or exogenous perturbations.

From the UE state, whenever "E_{off}" is triggered, the current estimation of $y^i(t_k)$ is reliable and will remain so until an "out of bound" or a "timeout" event occurs. The "out of bound" event is introduced to detect that the occurrence of a significant temperature drop $w^i(t_k)$ on (1), e.g., due to the cold water refill on a water heater after a hot-water drain event. This event is triggered whenever the estimated temperature exceeds its bounds by a certain percentage due to an unexpected TCL consumption, which literally brings $y^i(t_k)$ out of its bounds.

The "timeout" event accounts parameters uncertainties, long-term drifts on T^i_∞, or the cumulative effects of not significant in the small-term exogenous perturbations $w^i(t_k)$.

This event is triggered whenever the timer T^i_r measures that an interval of time greater than $\bar{T}^i_r$ has passed since the last synchronization event. The "timeout" event is needed because

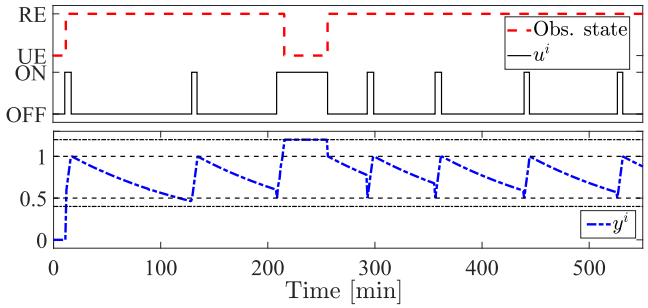

Fig. 3. Tests of the observer on a real water heater operating autonomously. Top: actual discrete observer state and thermostat state. Bottom: actual estimated virtual temperature.

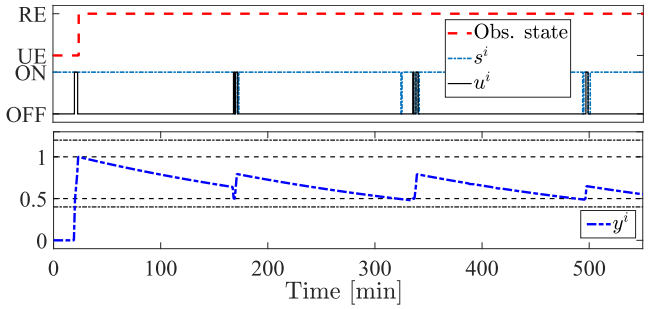

Fig. 4. Tests of the observer on a real water heater operating under the execution of the proposed DSM strategy. Top: actual discrete observer state and thermostat state and SPS switch state. Bottom: actual estimated virtual temperature.

in between synchronization events, the estimation of the virtual temperature is open loop, and thus, we set the observer to the UE state if no synchronization events occur within a maximum time window. $\bar{T}_r^i$ is experimentally set equal to $3\Delta T_{\text{off}}^i$.

Finally, notice that when the event "E_{on}" is triggered from the UE state, the new value of the estimated virtual temperature is considered to not be reliable because the temperature might be much lower T_{min}^i due to perturbations $w^i(t_k)$.

Remark 5: Detection of the events "E_{on}" and "E_{off}" depends on the most recent two states $s(t_{k-1})$ and $s(t_{k-2})$ of the SPS switch and the value of $u^i(t_{k-1})$, to discriminate whether $u^i(t_k)$ changed state due to the TCL control logic (3), or by the SPS due to (2). ■

Experimental Validation: In Figs. 3 and 4, the proposed observer is tested on two real-domestic water heaters with parameters identified in accordance with Section IV-A, $\epsilon_n^i = 5$ W, $\beta_1 = 0.8$ and $\beta_2 = 1.2$ and influenced by hot water drawing. In Fig. 3, the TCL is operating with $s^i(t_k) = 1\ \forall\ t_k$. We can observe that the virtual temperature estimation is reliable, from the first falling edge on $u^i(t_k)$ up to ≈ 210 min. On the contrary, at around 210 m due to hot water consumption and cold water refill, the observer detects an "out of bound 2" event. Then, it switches to the UE state while saturating $y^i(t_k)$. At around 250 m, "E_{off}" is triggered, and the observer sets itself back to the RE. From ≈ 250 to ≈ 350 m, the observer is on the RE state. In this case, estimation errors seem higher since $u^i(t_k)$ switches ON earlier than the expected time ($\approx \Delta T_{\text{off}}^i$). This is due to cold water refilling the tank and heat exchange in fluids occurring by convection currents [18], not modeled in neither (1) nor in (11). However, since events E_{off} and E_{on} reset $y^i(t_k)$ to the correct value, the estimation error in the RE state is kept small and bounded.

In Fig. 4, the response of the observer during the execution of the proposed DSM strategy during the experimental test described in Section VI, validating the observer also when the SPS is actuated the TCL with the DSM strategy.

C. Mixed Logic Dynamical Modeling of TCL-Plus-SPS Agents

To enable numerical control and optimization, the constrained hybrid dynamics of the ith TCL and SPS are written as an MLD system, i.e., as a set of mixed-integer linear inequalities, denoted $\chi^i(t_k)$.

Let us consider the identified dynamics of the virtual temperature (11), and let $A_i = e^{-\bar{\alpha}^i \Delta\tau}$ and $B_i = \bar{\eta}^i(1 - e^{-\bar{\alpha}^i \Delta\tau})$. Similarly, following the notation in (9), let us define

$$\begin{aligned}\mathbf{u}^i(t_k) &= \left[u_1^i \cdots u_\ell^i \cdots u_L^i\right]^\intercal = \left[u^i(t_{k+1}) \cdots u^i(t_{k+L})\right]^\intercal \\ \mathbf{y}^i(t_k) &= \left[y_1^i \cdots y_\ell^i \cdots y_L^i\right]^\intercal = \left[y^i(t_{k+1}) \cdots y^i(t_{k+L})\right]^\intercal \end{aligned} \quad (15)$$

which entries denote the control input and the virtual temperature over the prediction horizon from time t_{k+1} to t_{k+L}.

From (11) it yields $y_1^i = A_i y^i(t_k) + B_i u^i(t_k)$. Thus, the profile of the virtual temperature from time t_{k+2} to time t_{k+L}, evaluated at time t_k, as function of $\mathbf{u}^i(t_k)$, is

$$\begin{bmatrix} y_2^i \\ y_3^i \\ \vdots \\ y_L^i \end{bmatrix} = \underbrace{\begin{bmatrix} B_i & 0 & \cdots & 0 \\ A_i B_i & B_i & \cdots & 0 \\ \vdots & & \ddots & \\ A_i^{L-2} B_i & A_i^{L-3} B_i & \cdots & B_i \end{bmatrix}}_{\mathbf{F}^i} \underbrace{\begin{bmatrix} u_1^i \\ u_2^i \\ \vdots \\ u_L^i \end{bmatrix}}_{\mathbf{u}^i(t_k)} + \underbrace{\begin{bmatrix} A_i \\ A_i^2 \\ \vdots \\ A_i^{L-1} \end{bmatrix}}_{\mathbf{G}^i} y_1^i. \quad (16)$$

For completeness sake, by noting that y_1^i is fully determined by $u^i(t_k)$ and $y^i(t_k)$, which are not decision variables in the interval $[t_k, t_k + \Delta\tau)$, according to (16), it holds

$$\mathbf{y}^i(t_k) = \begin{bmatrix} 0 & \mathbf{0} \\ \mathbf{0} & \mathbf{F}^i \end{bmatrix} \begin{bmatrix} 0 \\ \mathbf{u}^i(t_k) \end{bmatrix} + \begin{bmatrix} 1 \\ \mathbf{G}^i \end{bmatrix} \left(B_i u^i(t_k) + A_i y^i(t_k)\right). \quad (17)$$

Let us now discuss how to model the hybrid agent behavior due to (2) and (3) into a set of linear integer constraints, such that each s_ℓ^i in (9), and each u_ℓ^i in (17) are coordinated to safely maintain each y^i within its operating range. First, we define

$$\begin{aligned} g_\ell^i &\triangleq y_\ell^i - y_{\text{min}}^i \\ f_\ell^i &\triangleq y_\ell^i - y_{\text{max}}^i. \end{aligned} \quad (18)$$

Clearly, $g_\ell^i \geq 0$ implies $y_\ell^i \geq y_{\text{min}}^i$ and $f_\ell^i \geq 0$ implies $y_\ell^i \geq y_{\text{max}}^i$. Let us define the logical *connectives* "$\wedge$" (and), "$\vee$" (OR), "$\sim$" (NOT), $\rightarrow$ (implies) and $\leftrightarrow$ (if and only if), see [19] for details.

The value of y_ℓ^i with respect to its bounds is now uniquely associated with two boolean dummy variables $\delta_{1,\ell}^i$ and $\delta_{2,\ell}^i$, as

$$[\delta_{1,\ell}^i = 1] \leftrightarrow [g_\ell^i \leq 0] \quad (19)$$

$$[\delta_{2,\ell}^i = 1] \leftrightarrow [f_\ell^i \leq 0]. \quad (20)$$

If, for instance, $y_\ell^i \in \left(y_{\text{min}}^i, y_{\text{max}}^i\right]$ then $\delta_{1,\ell}^i = 0$ and $\delta_{2,\ell}^i = 1$.

Now, on the basis of (2) and (3), particularized for $t_{k+\ell}$, a *truth table* of u^i_ℓ with respect $u^i_{\ell-1}$, s^i_ℓ and $\delta^i_{1,\ell-1}$, $\delta^i_{2,\ell-1}$ is computed. From that, the next equivalent logical statements have been derived

$$[s^i_\ell = 0] \rightarrow [u^i_\ell = 0] \tag{21}$$
$$[\delta^i_{1,\ell-1} = 0] \wedge [u^i_{\ell-1} = 0] \rightarrow [u^i_\ell = 0] \tag{22}$$
$$[\delta^i_{1,\ell-1} = 0] \wedge [\delta^i_{2,\ell-1} = 0] \rightarrow [u^i_\ell = 0] \tag{23}$$
$$[s^i_\ell = 1] \wedge [u^i_{\ell-1} = 1] \wedge [\delta^i_{2,\ell-1} = 1] \rightarrow [u^i_\ell = 1] \tag{24}$$
$$[s^i_\ell = 1] \wedge [\delta^i_{1,\ell-1} = 1] \rightarrow [u^i_\ell = 1]. \tag{25}$$

Due to space limitations, the intermediate steps involved in their evaluation are not provided. They can, however, be found in the archival version of this article available online (see [20]).

Although (24) and (25) force $u^i_\ell = 1$, if $y^i_{\ell-1} < y^i_{\min}$ ($\delta^i_{1,\ell-1} = 1$), to enable the TCL to be ON, a constraint enabling the power supply to the TCL by means of $s^i_\ell = 1$ is necessary, as follows:

$$\left[\delta^i_{1,\ell-1} = 1\right] \rightarrow \left[s^i_\ell = 1\right]. \tag{26}$$

Let us now translate the constraints (19)–(26) to the desired set of linear inequalities $\chi^i(t_k)$. First, note that the virtual temperature vector $\mathbf{y}^i(t_k)$ is bounded elementwise, and thus, (18) admits minimum and maximum points defined as follows:

$$\begin{aligned} m^i_f &= \min_{\ell=1,\dots,L} f^i_\ell, \quad m^i_g = \min_{\ell=1,\dots,L} g^i_\ell \\ M^i_f &= \max_{\ell=1,\dots,L} f^i_\ell, \quad M^i_g = \max_{\ell=1,\dots,L} g^i_\ell. \end{aligned} \tag{27}$$

Then, by invoking [19, Properties (2d) and (4e)], the logical implications in (19)–(26) can be formulated, for each $\ell = 1, \dots, L$, as the next set of linear mixed integer inequalities

$$(19) \equiv \begin{cases} g^i_\ell \le M^i_g \cdot \left(1 - \delta^i_{1,\ell}\right) \\ g^i_\ell \ge m^i_g \cdot \delta^i_{1,\ell} \end{cases} \tag{28}$$
$$(20) \equiv \begin{cases} f^i_\ell \le M^i_f \cdot \left(1 - \delta_{2,\ell}\right) \\ f^i_\ell \ge m^i_f \cdot \delta_{2,\ell} \end{cases} \tag{29}$$
$$(21) \equiv s^i_\ell \ge u^i_\ell \tag{30}$$
$$(22) \equiv u^i_\ell \ge \delta_{1,\ell-1} + u^i_{\ell-1} \tag{31}$$
$$(23) \equiv u^i_\ell \ge \delta_{1,\ell-1} + \delta_{2,\ell-1} \tag{32}$$
$$(24) \equiv u^i_\ell \ge s^i_\ell + u^i_{\ell-1} + \delta_{2,\ell-1} - 2 \tag{33}$$
$$(25) \equiv u^i_\ell \ge s^i_\ell + \delta_{1,\ell-1} - 1 \tag{34}$$
$$(26) \equiv s^i_\ell \ge \delta_{1,\ell-1}. \tag{35}$$

Now, since the estimation on $y^i(t_k)$ needed to initialize (15) is provided by the open-loop observer in Fig. 2 under uncertain and possibly perturbed conditions, and because the occurrence of the synchronous events is preferred since they reset the estimation error, the introduction of a constraint which forces the SPS switch to be ON for at least a minimum fraction $S^i_{\%,\text{on}}$ of the receding horizon time-window $\tau(t_k)$ is included

$$\sum_{\ell=1}^{L} s^i_\ell \ge L \cdot S^i_{\%,\text{on}}, \quad S^i_{\%,\text{on}} \in [0, 100]. \tag{36}$$

Thanks to (36), the occurrences of synchronization events E_{on} and E_{off} in the hybrid observer is greater, and thus, its corresponding estimation is reliable more often, improving the performance of the overall control architecture. In our setup $S^i_{\%,\text{on}} = 0.6$. Set $S^i_{\%,\text{on}} = 1$ makes agent i not cooperative, even if its expected consumption will be counterpart on the aggregated expected demand in (8) to be optimized.

Finally, by combining (16) and (28)–(36) with $\ell = 1, \dots, L$, we get the set of local MLD constraints $\chi^i(t_k)$ associated with each agent i in the optimization of the functional (7).

Definition 6 (Local Constraint Set $\chi^i(t_k)$): The SPS schedule plan $\mathbf{s}^i$, by each agent i is constrained by the set of mixed linear inequalities $\chi^i(t_k)$, which includes inequalities from (28) to (36) and the equality (16) with the addition of $\mathbf{s}^i_\ell, \mathbf{u}^i_\ell, \delta^i_{1,\ell}, \delta^i_{2,\ell} \in \{0, 1\}$ and $y^i_\ell \in \mathbb{R}^+$ for $\ell = 1, \dots, L$. ■

The set $\chi^i(t_k)$ consists of $1 + 11\ L$ linear inequalities and $5L$ variables. Only the L elements of $\mathbf{s}^i(t_k)$ in (9) are control variables, consisting on the actual planning for the SPS switch; s^i_1 denotes the switch command to be actuated at t_{k+1}.

D. Protocols for Dynamic Average Consensus

One of the key ideas of the proposed architecture is to exploit local interaction protocols for dynamic average consensus to enable each agent to estimate online the profile of the time-varying future planned global average power consumption P_ℓ in (8) of the TCLs. The estimation computed at time t_k of the average power consumption at time $t_{k+\ell} = t_k + \ell\Delta\tau$, is

$$\begin{aligned} \bar{P}_\ell(t_k) &= \frac{1}{n} \cdot P_\ell(t_k) = \frac{1}{n} \sum_{i \in \mathcal{V}} p^i_\ell(t_k) \\ &= \frac{1}{n} \sum_{i \in \mathcal{V}} p^i \cdot u^i_\ell, \quad \ell = 1, \dots, L \end{aligned} \tag{37}$$

where $p^i_\ell(t_k) = p^i \cdot u^i_\ell(t_k)$ is the power consumption plan of the ith agent, available at time t_k. In the remainder, $p^i_\ell(t_k)$ is considered as the local time-varying reference input of each agent that executes the dynamic consensus algorithm.

In particular, let $\bar{P}^i_\ell(t_k)$ be the estimation at time t_k of the average power consumption $\bar{P}_\ell$ (37) expected at time $t_{k+\ell} = t_k + \ell\Delta\tau$, by agent i. To distributedly estimate $\bar{P}_\ell$, for each time slot ℓ over the receding horizon window $\tau(t_k)$, L local instances of dynamic average consensus algorithm are needed. Here, the generic dynamic consensus updated rule is denoted as

$$\begin{aligned} \bar{P}^i_\ell(t + dt) = \text{D\_Consensus\_Update}\Big(& \bar{P}^j_\ell(t)|_{j \in \mathcal{N}^i}, \\ & \bar{P}^i_\ell(t),\ p^i_\ell(t_k)\Big) \end{aligned} \tag{38}$$

where dt is time required to execute an iteration of (38) according to the network capabilities. Due to space limitations, implementation details on (38) are omitted. The reader is referred to [21] for a comprehensive treatment of the topic. Roughly speaking, each agent updates its estimation $\bar{P}^i_\ell$ through local interaction within its neighborhood $\mathcal{N}^i$, by exchanging their local estimation $\bar{P}^j_\ell$. Among the many existing dynamic consensus algorithms, e.g., [22]–[26], we adopted the solution proposed in [24] because:

1) it can be easily tuned to achieve a desired trade-off between steady error and maximum tracking error;
2) it is robust to re-initialization due to changes in the network topology or size;
3) it can be implemented with randomized asynchronous state updates, and thus, it is resilient against communication failures or agent logout during the algorithm execution.

The performance of the dynamic consensus can be evaluated in terms of the maximum tracking error with respect to the average of the reference signals

$$e_{\text{track}}(t) = \max_{i\in\mathcal{V}} \left| \bar{P}_\ell^i(t) - \bar{P}_\ell(t) \right| = \max_{i\in\mathcal{V}} |\xi_\ell^i| \le \xi \qquad (39)$$

and its convergence rate, which can be improved by increasing the number of iterations per second. In our scenario, the timescale dt of the iterations of the dynamic consensus is in the range of the milliseconds.

Remark 7: The adopted dynamic consensus algorithm, proposed in [24], requires a communication topology represented by a connected undirected graph and bounded exogenous reference signals as working assumptions. It achieves a bounded maximum tracking error, characterized in [24]. The randomized version of dynamic average consensus in [24] considers a time-varying graph $\mathcal{G}(t_k)$, which consists at each instant of time of a set of random edges selected from a connected undirected graph. ■

V. TCL Cooperation Protocol

We now present the TCL cooperation protocol, detailed in Algorithm 1, which enables cooperation among the TCLs by minimizing the global objective (7) under the local constraints $\chi^i(t_k)$ derived in Section IV-C. These constraints uniquely determine u_ℓ^i for $\ell = 1,\dots,L$, as function of the only decision variable, the SPS switch state planning s_ℓ^i. This on the basis of the current virtual temperature estimation $y^i(t_k)$ obtained by the local observer, the current boolean state of the TCL $u^i(t_k)$, the system parameters identified with the method in Section IV-A, and the estimation on the average TCL power consumption profile of the network $\bar{P}_\ell(t_k)$ in (37), predicted via dynamic average consensus.

The "TCL Cooperation Protocol," consists of a local state update rule executed by each agent indefinitely. Each agent owns a local prediction $\bar{P}_\ell^i$ of the future average TCL power consumption of the network over the horizon $\tau(t_k)$.

From Task (a) of Algorithm 1, each agent updates its prediction $\bar{P}_\ell^i$ of $\bar{P}_\ell$ by executing in (38) the multistage dynamic consensus algorithm [24]. In parallel, and following Task (b) of Algorithm (1), each agent attempts to minimize, with probability μ, the local objective (40), consisting of the local SPS switch scheduling weighted by the predicted average TCL power consumption of the network over $\tau(t_k)$, subject to the local constraints $\chi^i(t_k)$. Notice that, the set $\chi^i(t_k)$ is not only nonconvex because it includes integer variables, but also time varying because it depends upon the current state of the TCL at the time in which the optimization takes place.

Although the optimization problem in (40) is in general NP-hard due to mixed-integer linear programming, in our setting for each agent i, the number of variables to be optimized is relatively small as only local constraints over a short time horizon $\tau(t_k)$ are involved. For example, about 20/60 steps into the future may account for 30 m to 2–3 hours of operations, depending on the tuning of the algorithm. Thus, the complexity of Algorithm 1 does not increase with the size of the network, but only with respect to the size of the time horizon L. This can be shown by noticing that the local optimization executed by each agent involves only its own state and its own prediction on the averaged aggregated consumption, while cooperation with other agents is achieved only through the execution of the

Algorithm 1 TCL Cooperation Protocol

Algorithm's Parameters:
dt: Maximum execution time of a dynamic consensus update;
Δt: Maximum execution time of the local optimization;
$\Delta\tau$: Time length of an optimization time-slot;
L: Number of time slots of the time window horizon;
$\tau(t_k) = L\cdot\Delta\tau$: Optimization's receding horizon time window;
ξ: maximum tracking error of dynamic consensus algorithm;
ε: small positive constant;
μ_i: probability of execution of a local optimization;

Algorithm Inputs:
$p^i(t_k)$: Measured local TCL power consumption;
$y^i(t_k)$: Estimated virtual temperature;
Obs_State$(t_k) = \{\text{UR}, \text{RE}\}$: Observer discrete state;

Algorithm Outputs:
$\mathbf{s}^i(t_k) = [s_1^i \cdots s_L^i]^\intercal \in \{0,1\}^L$: Local SPS scheduling plan;
$\mathbf{u}^i(t_k) = [u_1^i \cdots u_L^i]^\intercal \in \{0,1\}^L$: Local TCL scheduling;
$\mathbf{p}^i(t_k) = \mathrm{p}^i \cdot \mathbf{u}^i(t_k)$: Local Expected power consumption;

Initialize counter: $k = 0$;
Each agent i executes in parallel the next tasks:

- **Task (a).** Every dt seconds:
  1. Gather $\bar{P}_\ell^j$ for $\ell = 1,\dots,L$, from neighbors $j \in \mathcal{N}^i(t_k)$;
  2. Update state variables $\bar{P}_\ell^i$ for $\ell = 1,\dots,L$, according to (38);
- **Task (b).** Every Δt seconds:
  1. Update the measurement of power consumption $p^i(t_k)$;
  2. Update the state of the virtual temperature observer and collect the virtual temperature $y_i(t_k)$;
  3. **If** Obs_State$(t_k) = \{\text{RE}\}$ **then**, with probability μ_i update the ON/OFF scheduling according to an approximate solution of the next problem by a time-constrained (Δt) heuristic:

$$[\mathbf{s}^{i,\star}, \mathbf{u}^{i,\star}] = \operatorname{argmin}_{\mathbf{s}^i\in\chi(t_k)} \quad J^i(t_k) = \textstyle\sum_{\ell=1}^L P_\ell^i p_i u_\ell^i$$
$$J^{i,\star} = \textstyle\sum_{\ell=1}^L P_\ell^i p_i u_\ell^{i,\star}. \qquad (40)$$

  If a solution $[\mathbf{s}^{i,\star}, \mathbf{u}^{i,\star}]$ is found within Δt seconds and

$$|J^i(t_k) - J^{i,\star}| = \gamma_i \ge \frac{\xi p_i}{L} \sum_{\ell=1}^{L} |u_\ell^{i,\star} - u_\ell^i| + \varepsilon, \qquad (41)$$

  then set $\mathbf{s}^i(t_k) := \mathbf{s}^{i,\star}$, $\mathbf{u}^i(t_k) := \mathbf{u}^{i,\star}$
  Endif
  else $\mathbf{s}^i(t_k) := \mathbf{s}^i(t_k)$, $\mathbf{u}^i(t_k) := \mathbf{u}^i(t_k)$, i.e., do nothing.
  Endif
- **Task (c).** Every $\Delta\tau$ seconds:
  1. Set the current SPS state equal to: $s^i(t_{k+1}) := s_1^i(t_k)$;
  2. Shift the receding horizon time window by $\Delta\tau$:

$$\mathbf{u}^i(t_{k+1}) := \begin{bmatrix} u_2^i(t_k) & \cdots & u_\ell^i(t_k) & \cdots & u_{L-1}^i(t_k) & 1 \end{bmatrix}^\intercal; \qquad (42)$$
$$\mathbf{s}^i(t_{k+1}) := \begin{bmatrix} s_2^i(t_k) & \cdots & s_\ell^i(t_k) & \cdots & s_{L-1}^i(t_k) & 1 \end{bmatrix}^\intercal; \qquad (43)$$

Let $k := k+1$
Endif

dynamic consensus protocol which is designed for large-scale networks.

It should be noticed that approximate solutions to local optimization problems are sufficient to execute the heuristic. Here, a standard branch and bound solver with a limited maximum execution time have been considered. If a given solution does not improve on the current scheduling, it is simply discarded and the existing scheduling is kept out until a better one is found in future iterations. In Section VI, numerical simulations and experiments carried out for our case study are also discussed.

A. Convergence Analysis

First, we note that a feasible solution to the local optimization problem (40) always exists and such solution is $s^i(t_{k+\ell}) = 1$, for $\ell = 1, \ldots, L$, which corresponds to the SPS switch always ON. This can be easily proven by verifying that all MLD constraints are always satisfied by such solution.

Let $J^+(t_k)$ be the value of the global objective $J(t_k)$ in (7) after one agent updates its own local control action $\mathbf{s}^i(t_k)$ during the execution of Task (b) of Algorithm 1.

Theorem 8 (Online Optimization): Consider a network of TCL-plus-SPS executing Algorithm 1. If the dynamic consensus algorithm (38) executed in Task (a) of Algorithm 1 has maximum tracking error (39) less than ξ, and inequality (41) holds, then the global objective value decreases by

$$J^+(t_k) \leq J(t_k) - n\varepsilon \tag{44}$$

where $\varepsilon > 0$ is a small constant and n is the number of agents.

Proof: The global objective to be optimized online is

$$J(t_k) = \frac{1}{L}\sum_{\ell=1}^{L}\left(\sum_{i\in\mathcal{V}} p_i u_\ell^i\right)^2. \tag{45}$$

Now, the average power consumption estimated by each agent with the dynamic consensus algorithm (38) has, in general, a time-varying estimation error ξ_ℓ^i with respect to the real average power consumption $\bar{P}_\ell$ in (37), such that

$$\bar{P}_\ell = \frac{\sum_{i\in\mathcal{V}} p_i u_\ell^i}{n} = \bar{P}_\ell^i + \xi_\ell^i. \tag{46}$$

Thus, we can rewrite (45) as

$$\begin{aligned} J(t_k) &= \frac{1}{L}\sum_{\ell=1}^{L}\left(\sum_{i\in\mathcal{V}} p_i u_\ell^i\right)\left(\sum_{i\in\mathcal{V}} p_i u_\ell^i\right) \\ &= \frac{1}{L}\sum_{\ell=1}^{L} n\bar{P}_\ell\left(\sum_{i\in\mathcal{V}} p_i u_\ell^i\right) \\ &= n\sum_{i\in\mathcal{V}}\frac{1}{L}\sum_{\ell=1}^{L}\left(\bar{P}_\ell^i + \xi_\ell^i\right) p_i u_\ell^i(t_k). \end{aligned} \tag{47}$$

Now, by letting $J^i(t_k) = (1/L)\sum_{\ell=1}^{L}\left(\bar{P}_\ell^i + \xi_\ell^i\right) p_i u_\ell^i$, it holds $J(t_k) = n\sum_{i\in\mathcal{V}} J^i(t_k)$. Therefore, if exact knowledge of $\bar{P}_\ell = \bar{P}_\ell^i + \xi_\ell^i$ was available to each agent, we could guarantee the optimization of the global objective function over the nonconvex set of constraints (up to a local minimum) by updating the local ON/OFF scheduling as

$$[\mathbf{s}^{i,\star}, \mathbf{u}^{i,\star}] = \underset{\mathbf{s}^i\in\chi(t_k)}{\operatorname{argmin}} \quad \frac{1}{L}\sum_{\ell=1}^{L}\bar{P}_\ell\, p^i u_\ell^i. \tag{48}$$

Instead, since the agents do not have access to global information on the network, i.e., they do not have access to the updated values of $\bar{P}_\ell$ during the iterations, an estimation of $\bar{P}_\ell$ is employed. Since we consider a dynamic consensus process which has bounded tracking error less than ξ, it holds

$$\max_{i\in\mathcal{V}} |\bar{P}_\ell^i - \bar{P}_\ell| \leq \xi \quad \forall i \in \mathcal{V}, \quad \ell = 1, \ldots, L. \tag{49}$$

During the execution of Task (b) of Algorithm 1, each agent optimizes its local objective $J^i(t_k)$ in (40) on the basis of its own local estimation on the predicted average power consumption of the network $\bar{P}_\ell^i$, that is,

$$\tilde{J}^i(t_k) = \frac{1}{L}\sum_{\ell=1}^{L}\bar{P}_\ell^i\, p^i u_\ell^i = J^i(t_k) + \frac{1}{L}\sum_{\ell=1}^{L}\xi_\ell^i\, p^i u_\ell^i. \tag{50}$$

Thus, let

$$[\mathbf{s}^{i,\star}, \mathbf{u}^{i,\star}] = \underset{\mathbf{s}^i\in\chi_i(t_k)}{\operatorname{argmin}}\, \tilde{J}^i(t_k) = \underset{\mathbf{s}^i\in\chi_i(t_k)}{\operatorname{argmin}}\, \frac{1}{L}\sum_{\ell=1}^{L}\bar{P}_\ell^i\, p^i u_\ell^i \tag{51}$$

and $\tilde{J}^{i,\star} = (1/L)\sum_{\ell=1}^{L}\bar{P}_\ell^i\, p^i u_\ell^{i,\star}$. Now, denote with γ_i the computed decrement of the local objective function affected by estimation errors

$$\tilde{J}^{i,\star} - \tilde{J}^i(t_k) = -\gamma_i. \tag{52}$$

We now compute a sufficient value of γ_i which guarantees the optimization of the global objective despite persistent estimation errors. An actual decrement of the local objective after one agent executes Task (b) of Algorithm 1 is obtained if

$$J^{i,\star}(t_k) - J^i(t_k) < -\varepsilon. \tag{53}$$

By rewriting (52) and substituting $\tilde{J}^i(t_k)$ by exploiting (50)

$$J^{i,\star}(t_k) - J^i(t_k) = -\gamma_i + \frac{1}{L}\sum_{\ell=1}^{L}\xi_\ell^i\, p_i\left(u_\ell^{i,\star} - u_\ell^i\right). \tag{54}$$

Then, by putting together the inequalities in (53) and (54), it follows that to ensure (53), it suffices that:

$$\gamma_i > \frac{1}{L}\sum_{\ell=1}^{L}\xi_\ell^i\, p_i\left(u_\ell^{i,\star} - u_\ell^i\right) + \varepsilon. \tag{55}$$

By considering an upper bound to the estimation error ξ_ℓ^i by exploiting (49), it holds $|\xi_\ell^i| \leq \xi$ for all $\ell = 1, \ldots, L$, and for all $i \in \mathcal{V}$. Thus, we can rewrite (55) as the next inequality

$$\gamma_i > \frac{\xi p_i}{L}\sum_{\ell=1}^{L}\left|u_\ell^{i,\star} - u_\ell^i\right| + \varepsilon. \tag{56}$$

Therefore, if (56) holds, the global objective function $J^+(t_k)$ after one agent executes Task (b) of Algorithm 1 is

$$J^+(t_k) = n\left(\left(\sum_{j\in\mathcal{V}\setminus i} J^j(t_k)\right) + J^{i,\star}\right) \tag{57}$$

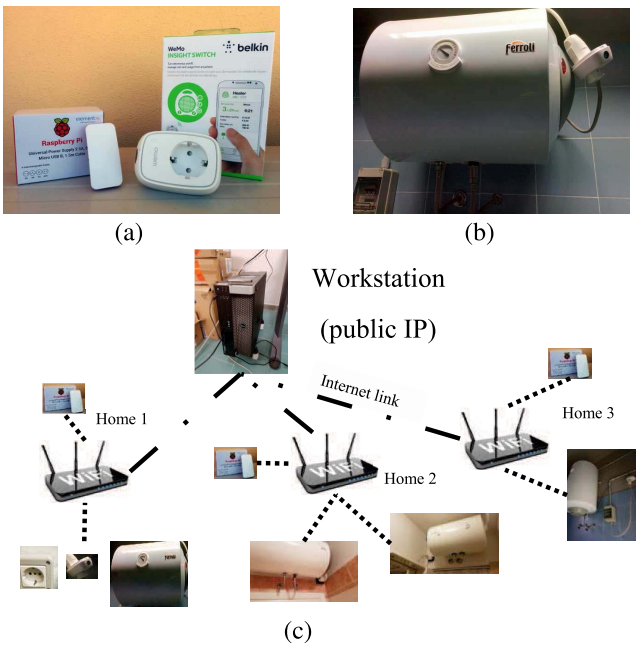

Fig. 5. The infrastructure behind the CoNetDomeSys testbed. (a) A Raspberry Pi Zero W and a WeMo Insight Switch. (b) A Water heater equipped with a WeMo Insight Switch. (c) CoNetDomeSys testbed's cyber-physical infrastructure.

and decreases, with respect to its value before the local update, by at least

$$J^+(t_k) < n\left(\left(\sum_{j\in\mathcal{V}\setminus i} J^j(t_k)\right) + J^i(t_k) - \varepsilon\right) = J(t_k) - n\varepsilon \tag{58}$$

thus proving the statement of Theorem 8. ■

VI. Testbed Description and Experiments

The "CoNetDomeSys" testbed, short for "Cooperative Network of Domestic Systems," is a low cost IoT-oriented experimental demonstrator for fast prototyping and testing of DSM algorithms on large populations of domestic appliances controlled and monitored by means of SPS plug adapters.

The testbed is designed around off-the-shelf, low-cost hardware components. The only requirement is having Internet Wi-Fi facilities. Its core component is the WeMo Insight Switch [27], shown in Fig. 5(a). The choice of which was dictated by the availability of Open-APIs to integrate purpose-built software, thus enabling remote monitoring and control. It is a 220-V/16-A/50-Hz power outlet adapter provided with

1) a power consumption sensing unit with a rated resolution of 1 mW and tested a maximum sampling frequency of 1 Hz;
2) a latching switch to remotely/manually enable the power supply to the appliance plugged within its power outlet;
3) a controller with WiFi 2.4 GHz 802.11n capabilities.

A number of SPSs were delivered to volunteers in the city of Cagliari (Italy) who agreed to participate in this experimental campaign. A domestic TCL is plugged into each SPS as shown, e.g., in Fig. 5(b). In each domestic environment, a Raspberry Pi Zero W [28], shown on the left of Fig. 5(a), is also connected to the same LAN where the SPSs are connected to.

Two Java software applications were also developed: one installed on each Raspberry Pi Zero W and implementing a client–server communication and control infrastructure over the Telnet protocol for the SPSs; one installed in a workstation with a public IP address located in our laboratory.

The application running on the Raspberry, for each SPS within the same LAN, queries the measured consumption $p^i(t_k)$ while enabling the control of the switch $s^i(t_k)$. Then, it forwards measurements/receives control commands to/from the workstation. In Fig. 5(c), the testbed's cyber-physical infrastructure is pictured.

The workstation application collects the received data from the population of SPSs and sends back asynchronous actuation commands. It is also integrated with a MATLAB interface for fast prototyping of distributed coordination algorithms. Both communication and actuation delays are below 1 s.

A network topology is assigned to the agents, and each can only share information anonymously with its neighbors. The topology and size of the network are unknown to the agents.

The workstation is a Dell Precision t5810 equipped with an Intel Xeon E5-1620 v3 (10M Cache, 3.50 GHz), 64 GB of RAM, and Windows 10 Pro installed.

The processing carried out by the testbed is centralized and takes place in a single workstation. Therefore, the testbed validates the responsiveness experimentally to the DSM of the TCLs while to enable fast distributed algorithm prototyping and testing, in the MATLAB environment, the processing and communications among agents are simulated via software.

Parameters and Setting of the Experimental Scenario

To validate the proposed approach on the CoNetDomeSys testbed, a small scale scenario involving domestic TCLs located in a set of private homes of volunteers has been considered.

To limit the number of volunteers required to carry out the test, while executing the DSM cooperation strategy on a network of at least 100 agents, a set of numerically simulated TCLs equipped with SPS, modeled accordingly with (1)–(4), is also considered. In particular, we had access to 12 electric water heaters and 3 electric radiators in 10 different locations.

Then, we introduced 85 numerically simulated agents with parameters identified from other real TCLs involved in the experiment, thus yielding a total population of $n = 100$ agents.

The parameters of each TCL were identified with the method described in Section IV-A and by letting, for all agents, the free parameter $\bar{y}^i_{\min} = 0.5$. The list of parameters used for the tuning of model (11) is provided in Table II.

In our scenario, we considered a P2P network topology represented by a random undirected Erdös-Rényi graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$, with edge existence probability $3\log(n)/n$. Tests were carried out in real-time, and in accordance with Algorithm 1, the time-related parameters were set as follows.

1) Task (a), dynamic consensus: $dt = 10$ ms.

TABLE II
TEST PARAMETERS

#	TCL	$\bar{\alpha}^i$ [s^{-1}]	$\bar{\eta}^i$ [–]	p^i [kW]
12	Water Heater	$(0.3 \div 5.3) \times 10^{-4}$	$6.17 \div 48.25$	$1.2 \div 1.5$
3	Radiator	$(1.5 \div 3.5) \times 10^{-3}$	$1.6 \div 5.5$	$1.2, 2.0$
85	Simulated TCL	$(1.3 \div 2.4) \times 10^{-4}$	$21 \div 25$	$1.0 \div 2.0$

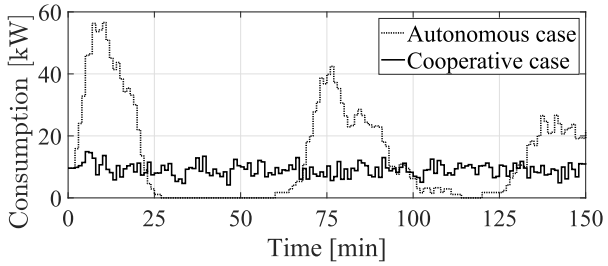

Fig. 6. Comparison between the power consumption of 100 TCLs in the autonomous case and in the cooperative case in the mixed scenario (16 real TCLs).

2) Task (b), local optimization: $\Delta t = 1$ s.
3) Task (c), SPS actuation time slots: $\Delta\tau = 1$ min.

A receding horizon time window $\tau(t_k)$ of 40 min was chosen, and thus, $L = 40$ time slots with length $\Delta\tau = 1$ min each. The probability of executing the local optimization every Δt on each device was set to $\mu^i = 1/30$ for all agents. The expected number of local optimization rounds (40), executed by each agent in one time-slot $\Delta\tau$, is $\mu^i \Delta\tau \Delta t = 2$.

Local optimization problems were solved with the MATLAB MILP solver "`intlinprog`." The solver is set to use a time-constrained "branch and bound" search with maximum execution of 5 s. The average time to execute an optimization was about 0.03 s. In the large majority of cases, optimal solutions were found within the time limit.

In our test with 100 agents, we considered a time span of 550 min (≈9 h). Numerically simulated TCLs and SPS were given random initial conditions with virtual temperature $y^i(t_0)$ in the interval $[0.5, 0.7]$ and $s^i(t_0) = 1$ and $h^i(t_0) = 0$.

This ensured a scenario, referred as "autonomous case" where, if no DSM strategies are executed on the SPSs (thus, $s^i(t_k) = 1 \ \forall \ t_k \in \mathbb{R}^+$), about 90% of the simulated TCLs would need to turn ON their heating element within the first 25 min of the experimentation, thus showing as highlighted by the dotted power consumption profile of Fig. 6, significant peaks. Clearly, a DSM strategy able to desynchronize the consumption of the TCLs, in spite of their local temperature constraints, and in the absence of any absolute temperature information is needed.

Experimental Validation: In contrast with the "autonomous case," the scenario where the network of SPSs executes Algorithm 1 is referred to as "cooperative case." A comparison between the actual power consumption by the considered network of 100 mixed TCLs (15 real and 85 simulated) in the two cases is shown in Fig. 6. The details of the behavior of the first 8 simulated TCLs are shown in Fig. 7.

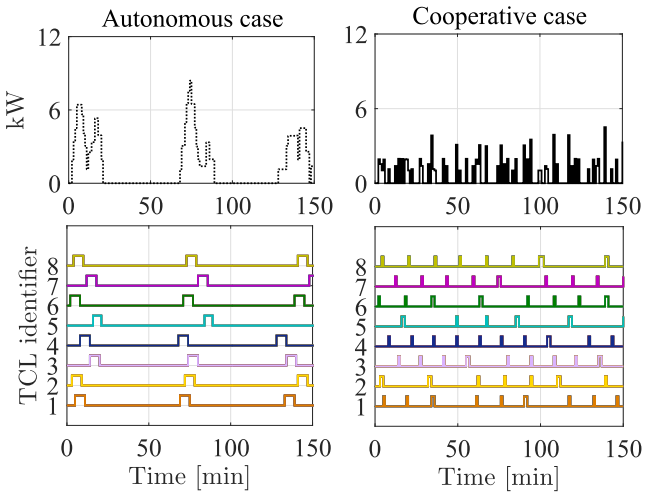

Fig. 7. Comparison between the power consumption of 8 simulated TCLs in the autonomous case and the cooperative case. The state of each TCL is in different colors.

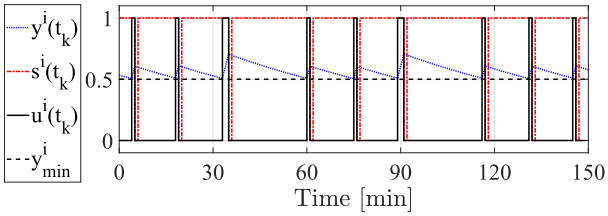

Fig. 8. Evolution of virtual temperature $y^i(t_k)$, SPS state $s^i(t_k)$, and TCL state $u^i(t_k)$ of one agent in Algorithm 1.

In spite of the adverse agent's initialization, the same in both case studies, the actual power consumption associated with the "cooperative case" is shaved off with respect to the "autonomous case." In particular, from the bottom plots of Fig. 7, it can be noted that, compared with the "autonomous case" (left), in the "cooperative case" (right) the discrete state of each TCL $u^i(t_k)$ switches with increased frequency, while maintaining, in accordance with Remark 2, similar average power demand.

Figs. 4 and 8 are shown, respectively, for a real TCL and for a simulated one, the details of the evolution of the discrete state of the SPS and TCL with the corresponding virtual temperature during the execution of Algorithm 1.

It can be seen that the switch of the SPS is mostly in the ON state accordingly with (36), where $S^i_{\%,\text{on}} = 0.6$. Thus, there is no significant issue neither due the excessive switching of the electromechanical switch (which might reduce the lifespan of the SPS), nor with respect to the presence of estimation errors on $y^i(t_k)$, which are promptly reset to zero by the observer at each occurrence of the events E_{on}, or E_{off}. This is evident in Fig. 4, where temperature drops due to user behavior are present.

From the discussion earlier, we conclude that the proposed strategy ensures peak consumption reduction and promotes, in general, the desynchronization of the TCLs power consumption.

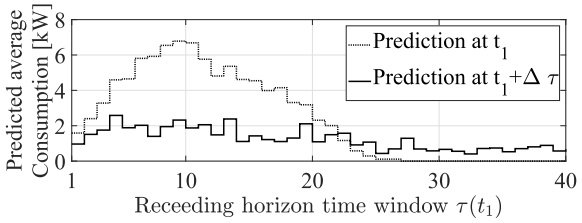

Fig. 9. Comparison of the predicted averaged consumption $\bar{P}_\ell^i$ by agent $i = 1$, $\ell = 1, \ldots, 40$, at time t_1 and time $t_1 + \Delta\tau$.

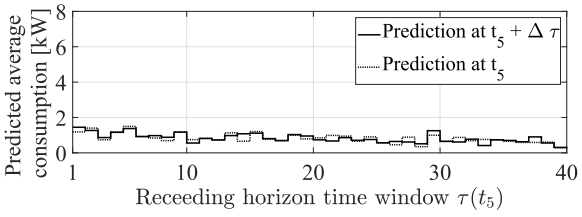

Fig. 10. Comparison of the predicted averaged consumption $\bar{P}_\ell^i$ by agent $i = 1$, $\ell = 1, \ldots, 40$, at time t_5 and time $t_5 + \Delta\tau$.

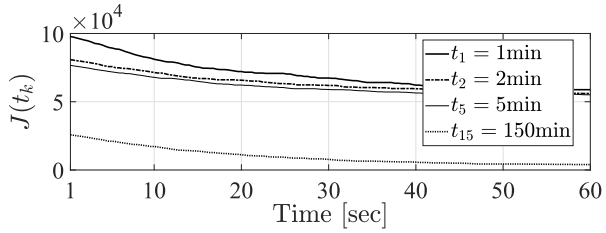

Fig. 11. Evolution of the global objective function during the execution of the DSM strategy at different intervals of time t_k.

In Fig. 9, a comparison between the predicted average power consumption $\bar{P}_\ell^i$, for $\ell = 1, \ldots, 40$, estimated with the dynamic consensus algorithm in (38) by agent 1 at two different instants of time, at $t_1 = 1$ min and at $t_2 = t_1 + \Delta\tau = 2$ min by which time each agent is expected to have solved (40) at least twice in accordance with the chosen value of μ^i, is shown. The same comparison is also evaluated at $t_5 = 5$ min and at $t_6 = 6$ min in Fig. 10.

It can be seen that major changes in the predicted power consumption by the network occur mostly during the startup phase since the agents are initialized in such a way that their power consumption is synchronized. Then, after the time window $\tau(t_k)$ recedes by a few time slots, the network reaches a steady solution for its average consumption where only small changes occur in the local schedule plans.

The tests show that even a few iterations of Task (b) of Algorithm 1 are sufficient to significantly improve the global objective, i.e., reduce peak demand and load variations. Since the approach is real time and based on feedback, errors due to local TCL parameter uncertainty and estimation errors are mitigated and averaged within the whole network, thus providing robustness.

In Fig. 11, it is shown a validation of the result in Theorem 8. It shows that the execution of the TCL cooperation protocol with randomized local optimizations, i.e., Task (b) of Algorithm 1, provides a decrement on the global objective (7) despite estimation errors and real dynamical evolution of the TCLs.

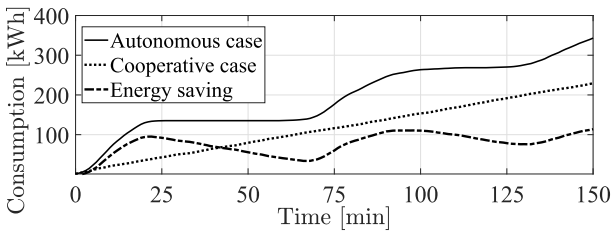

Fig. 12. Energy consumption profiles, integrated in a 150-min time window, in the autonomous case and cooperative case.

Finally, in Fig. 12, it is shown a comparison between the energy consumption profiles, integrated within a 150-min time window, with the TCLs in autonomous operation and during the execution of the TCL cooperation protocol. It can be seen that as a by-product of the proposed cooperation strategy, also total power consumption by the network is reduced, thus realizing energy savings. This occurs because the modulation of the ON/OFF state of the TCLs forces their temperature to be closer to the lower limit of the desired temperature range $[T^i_{\min}, T^i_{\max}]$, thus reducing thermal losses with respect to the ambient temperature. The linear growth of the cooperative case in Fig. 12 is explained by the almost uniform and constant power consumption integrated over time that our method is able to achieve, thus yielding an almost linear growth in the experimental case study.

VII. Conclusion

In this article, we presented: 1) a multi-agent DSM architecture for the coordination of anonymous networks of TCLs, not tailored for DSM task, enabled by the use of SPSs; 2) a method for power consumption model identification of TCLs based only on power consumption measurements; 3) a hybrid observer for the estimation of the TCL internal state based only on consumption measurements; 4) a method for distributed online constrained optimization method of the power consumption of a network of TCLs controlled by SPSs; 5) a low-cost experimental testbed based on off-the-shelf hardware along with a dedicated software; and 6) an experimental validation of the proposed method.

Future work will build upon the proposed architecture by generalizing to virtual power plants.

References

[1] Office of Energy. Government of Western Australia. (2010). *Demand Management*. [Online]. Available: https://www.webcitation.org/

[2] U.S. Energy Information Administration. USA, DoE, Energy Website. DOE.GOV. (2018). *Use of Electricity—Energy Explained, Your Guide to Understanding Energy*. Accessed: Oct. 8, 2016. [Online]. Available: https://www.webcitation.org/

[3] C. Li, X. Yu, W. Yu, G. Chen, and J. Wang, "Efficient computation for sparse load shifting in demand side management," *IEEE Trans. Smart Grid*, vol. 8, no. 1, pp. 250–261, Jan. 2017.

[4] P. Palensky and D. Dietrich, "Demand side management: Demand response, intelligent energy systems, and smart loads," *IEEE Trans Ind. Informat.*, vol. 7, no. 3, pp. 381–388, Aug. 2011.

[5] H. Hao, B. M. Sanandaji, K. Poolla, and T. L. Vincent, "Aggregate flexibility of thermostatically controlled loads," *IEEE Trans. Power Syst.*, vol. 30, no. 1, pp. 189–198, Jan. 2015.
[6] S. H. Tindemans, V. Trovato, and G. Strbac, "Decentralized control of thermostatic loads for flexible demand response," *IEEE Trans. Control Syst. Technol.*, vol. 23, no. 5, pp. 1685–1700, Sep. 2015.
[7] S. Grammatico, B. Gentile, F. Parise, and J. Lygeros, "A mean field control approach for demand side management of large populations of thermostatically controlled loads," in *Proc. Eur. Control Conf. (ECC)*, Jul. 2015, pp. 3548–3553.
[8] M. Franceschelli, A. Gasparri, and A. Pisano, "Coordination of electric thermal systems for distributed demand-side management: A gossip-based cooperative approach," in *Proc. Eur. Control Conf. (ECC)*, Jun. 2016, pp. 623–630.
[9] J. H. Braslavsky, C. Perfumo, and J. K. Ward, "Model-based feedback control of distributed air-conditioning loads for fast demand-side ancillary services," in *Proc. 52nd IEEE Conf. Decis. Control*, Dec. 2013, pp. 6274–6279.
[10] A. D. Dominguez-Garcia, S. T. Cady, and C. N. Hadjicostis, "Decentralized optimal dispatch of distributed energy resources," in *Proc. IEEE 51st IEEE Conf. Decis. Control (CDC)*, Dec. 2012, pp. 3688–3693.
[11] H. Xing, Y. Mou, Z. Lin, and M. Fu, "Fast distributed power regulation method via networked thermostatically controlled loads," in *Proc. IFAC World Congr.*, 2014, pp. 5439–5444.
[12] S. Grammatico, F. Parise, M. Colombino, and J. Lygeros, "Decentralized convergence to Nash equilibria in constrained deterministic mean field control," *IEEE Trans. Autom. Control*, vol. 61, no. 11, pp. 3315–3329, Nov. 2016.
[13] I. Notarnicola, M. Franceschelli, and G. Notarstefano, "A duality-based approach for distributed min-max optimization with application to demand side management," in *Proc. IEEE 55th Conf. Decis. Control (CDC)*, Dec. 2016, pp. 1877–1882.
[14] I. Notarnicola, M. Franceschelli, and G. Notarstefano, "A duality-based approach for distributed min–max optimization," *IEEE Trans. Autom. Control*, vol. 64, no. 6, pp. 2559–2566, Jun. 2019.
[15] L. J. Ratliff, R. Dong, H. Ohlsson, A. A. Cardenas, and S. S. Sastry, "Privacy and customer segmentation in the smart grid," in *Proc. 53rd IEEE Conf. Decis. Control*, Dec. 2014, pp. 2136–2141.
[16] M. Shad, A. Momeni, R. Errouissi, C. P. Diduch, M. E. Kaye, and L. Chang, "Identification and estimation for electric water heaters in direct load control programs," *IEEE Trans. Smart Grid*, vol. 8, no. 2, pp. 947–955, Nov. 2015.
[17] A.-H. Mohsenian-Rad, V. W. S. Wong, J. Jatskevich, R. Schober, and A. Leon-Garcia, "Autonomous demand-side management based on game-theoretic energy consumption scheduling for the future smart grid," *IEEE Trans. Smart Grid*, vol. 1, no. 3, pp. 320–331, Dec. 2010.
[18] S. Kakaç, *Boilers, Evaporators, and Condensers*. Hoboken, NJ, USA: Wiley, 1991.
[19] A. Bemporad and M. Morari, "Control of systems integrating logic, dynamics, and constraints," *Automatica*, vol. 35, no. 3, pp. 407–427, Mar. 1999.
[20] M. Franceschelli, A. Pilloni, and A. Gasparri, "Multi-agent coordination of thermostatically controlled loads by smart power sockets for electric demand side management," 2019, *arXiv:1908.11318*. [Online]. Available: http://arxiv.org/abs/1908.11318
[21] S. S. Kia, B. Van Scoy, J. Cortes, R. A. Freeman, K. M. Lynch, and S. Martinez, "Tutorial on dynamic average consensus: The problem, its applications, and the algorithms," *IEEE Control Syst.*, vol. 39, no. 3, pp. 40–72, Jun. 2019.
[22] M. Zhu and S. Martínez, "Discrete-time dynamic average consensus," *Automatica*, vol. 46, no. 2, pp. 322–329, Feb. 2010.
[23] S. S. Kia, J. Cortés, and S. Martínez, "Dynamic average consensus under limited control authority and privacy requirements," *Int. J. Robust Nonlinear Control*, vol. 25, no. 13, pp. 1941–1966, Sep. 2015.
[24] M. Franceschelli and A. Gasparri, "Multi-stage discrete time and randomized dynamic average consensus," *Automatica*, vol. 99, pp. 69–81, Jan. 2019.
[25] E. Montijano, J. I. Montijano, C. Sagüés, and S. Martínez, "Robust discrete time dynamic average consensus," *Automatica*, vol. 50, no. 12, pp. 3131–3138, Dec. 2014.
[26] B. Van Scoy, R. Freeman, and K. Lynch, "A fast robust nonlinear dynamic average consensus estimator in discrete time," in *Proc. IFAC Workshop Distr. Estim. Control Networked Sys.*, 2015, vol. 48, no. 22, pp. 191–196.
[27] Belkin. (2019). *Wemor Insight Smart Plug*. [Online]. Available: https://www.belkin.com/us/p/P-F7C029/
[28] Raspberry Pi Foundation. (2019). *Raspberry PI Zero W*. [Online]. Available: https://www.raspberrypi.org/products/raspberry-pi-zero-w/

Mauro Franceschelli (Member, IEEE) received the Laurea degree *(cum laude)* in electronic engineering and the Ph.D. degree from the University of Cagliari, Cagliari, Italy, in 2007 and 2011, respectively.

He is currently a Senior Assistant Professor (RTD-B) with the Department of Electrical and Electronic Engineering, University of Cagliari, Cagliari, Italy. In 2013, he received a fellowship from the National Natural Science Foundation of China (NSFC), Grant No. 61450110086, at Xidian University, Xi'an, China. In 2015, he received the Italian grant "Scientific Independence of Young Researchers" (SIR), with project "CoNetDomeSys," code RBSI14OF6H, funded by the Italian Ministry of Education, University and Research (MIUR) under the 2014 call. His research interests include consensus problems, gossip algorithms, multi-agent systems, multirobot systems, distributed optimization, and electric demand-side management.

Dr. Franceschelli has been a member of the Conference Editorial Board for the IEEE Control Systems Society since 2019. He has been serving as an Associate Editor for the IEEE Conference on Automation Science and Engineering (CASE) since 2015, the IEEE American Control Conference (ACC) since 2019, and the IEEE Conference on Decision and Control (CDC) since 2020.

Alessandro Pilloni (Member, IEEE) received the Laurea (M.S.) degree (Hons.) in electronic engineering and the Ph.D. degree in industrial engineering with Doctor Europaeus mention from the University of Cagliari, Cagliari, Italy, in 2010 and 2014, respectively.

He held visiting positions at the Universitat Politècnica de València, Valencia, Spain, the University of Leicester, Leicester, U.K., and the University of Belgrade School of Electrical Engineering, Belgrade, Serbia. He is currently an Assistant Professor (RTD-A) with the Department of Electrical and Electronic Engineering, University of Cagliari, Cagliari, Italy. He has also served as a system engineer and a technical advisor for small- and medium-sized enterprises within its regional territory. He has coauthored more than 34 articles published on international journals, conference proceedings, and books. His research area includes nonlinear control and variable structure control theory and their applications to uncertain and/or distributed parameter systems, distributed optimization and multi-agent control with a focus on microgrid control and demand-side management.

Andrea Gasparri (Senior Member, IEEE) received the Laurea degree *(cum laude)* in computer science and the Ph.D. degree in computer science and automation from Roma Tre University, Rome, Italy, in 2004 and 2008, respectively. He is currently a Professor with the Department of Engineering, Roma Tre University. He received the Italian Grant FIRB Futuro in Ricerca 2008 for the project Networked Collaborative Team of Autonomous Robots funded by the Italian Ministry of Research and Education. He is currently the coordinator of the project "PANTHEON" focused on robotics for precision farming supported by the European Community within the H2020 framework (under grant agreement number 774571). His research interests are in the areas of robotics, sensor networks, networked multi-agent systems, and precision farming.

Dr. Gasparri has been a member of the Conference Editorial Board for the IEEE Control Systems Society since 2017. He has been an Associate Editor of the IEEE TRANSACTIONS ON CYBERNETICS since 2017.