A decentralized framework for the optimal coordination of distributed energy resources

Miguel F. Anjos, *Member, IEEE,* Andrea Lodi, and Mathieu Tanneau

***Abstract*—Demand-response aggregators are faced with the challenge of how to best manage numerous and heterogeneous Distributed Energy Resources (DERs). This paper proposes a decentralized methodology for optimal coordination of DERs. The proposed approach is based on Dantzig-Wolfe decomposition and column generation, thus allowing to integrate any type of resource whose operation can be formulated within a mixed-integer linear program. We show that the proposed framework offers the same performance guarantees as a centralized formulation, with the added benefits of distributed computation. The practical efficiency of the algorithm is demonstrated through extensive computational experiments, on a set of 1120 instances generated using data from Ontario energy markets. The proposed approach was able to solve all test instances to proven optimality, while achieving significant speed-ups over a centralized formulation solved by state-of-the-art optimization software.**

***Index Terms*—Column generation, Dantzig-Wolfe decomposition, demand response aggregation, distributed energy resources, mixed-integer linear programming, smart grid**

I. Introduction

SMART grids hold the promise of more reliable and sustainable power grids, through the integration of communication technologies, advanced computation and intelligent controls. Among smart grid-enabled paradigms, Demand Response (DR) programs [1] enable end-users to dynamically adjust their electricity consumption, in response to price signals or incentives. The present work focuses on the DR potential of Distributed Energy Resources (DERs), which include flexible loads, distributed generation, and distributed energy storage.

Because of their small size, individual resources have a negligible marginal impact at the grid level. This has motivated the introduction of aggregators [2], that act as intermediaries between the grid and resources, thus enabling the latters to participate in traditional energy markets. Therefore, aggregators must address the challenges associated with coordinating numerous and heterogeneous resources, whose operation may involve discrete decisions. To that end, various coordination mechanisms have been investigated in the literature [3]–[5]. Among them, pricing strategies [5] rely on each resource optimizing its own individual objective, in response to price signals such as Time-of-Use rates. However, it was pointed out in [6] that pricing strategies may lead to unstable market behaviours. Conversely, the approach considered in this paper seeks to jointly coordinate the operation of all resources, so as to optimize a global objective.

The authors are with École Polytechnique de Montréal, Department of Mathematics and Industrial Engineering, Montréal, QC H3T 1J4, Canada, & Groupe d'études et de recherche en analyse des décisions (GERAD), Montréal, QC H3T 1J4, Canada (e-mail: miguel-f.anjos@polymtl.ca).

Such a global approach can be formulated as a centralized optimization problem, as was explored in [7]–[10], wherein a central controller manages each individual resource. Although a centralized formulation offers the strongest optimality guarantees, it quickly becomes intractable when the number of resources increases. Moreover, the disclosure of private information by the resources raises privacy concerns, and puts additional burden on communication requirements. Decentralized methods, on the other hand, distribute the computational effort among resources by leveraging local computing power. This typically results in better scalability, reduced communication overheads, and improved privacy. Several distributed heuristics were proposed, for example in [11]–[14], but provide weaker performance guarantees. Nevertheless, classical decomposition techniques allow to solve the centralized problem in a distributed way, thus offering the same performance guarantees, with the added benefit of decentralized computation.

A large body of literature has focused on dual decomposition and Lagrangian-based methods. Dual decomposition yields a separable structure, which in turn enables distributed implementations. The related works [15]–[20] thus differ mainly in which algorithm is used to optimize the dual Lagrangian. In [15], [16], the authors consider an augmented Lagrangian-based relaxation, which is formulated as a consensus problem and solved with the Alternating Direction Method of Multipliers (ADMM). However, ADMM does not handle discrete variables, that are used to model on-off constraints. In a similar fashion, standard Lagrangian relaxation is used in [17]–[20]. A classical sub-gradient algorithm is investigated in [17] and cutting-planes methods are studied in [18], but discrete variables were not considered.

Although the works in [19], [20] do consider mixed-integer variables, they rely on recovery heuristics to obtain feasible solutions. A bundle method is used in [19], while a double smoothing of the dual Lagrangian is applied in [20], allowing the use of a more efficient gradient-based algorithm. Overall, the main drawback of Lagrangian-based approaches is the recovery of feasible solutions when strong duality does not hold, which is generally the case for mixed-integer problems.

Alternatively, a primal approach, namely Dantzig-Wolfe (DW) decomposition, was investigated in [21]–[23]. In [21], [22], DW decomposition is applied to demand-response problems that arise from peak-load management applications. However, only a few types of devices were considered, with no discrete variables involved. Furthermore, a column-generation heuristic is used in [23] to schedule residential heating systems. Nevertheless, this heuristic approach does not provide a guarantee of optimality.

Our contribution is twofold. On one hand, we propose a column generation-based decentralized methodology, to optimally coordinate DERs whose operation may involve mixed-integer variables. This methodology is motivated by DR aggregation of residential customers. On the other hand, we demonstrate the practical efficiency of the proposed algorithm through extensive computational experiments, on a set of 1120 instances generated using data from Ontario energy markets.

The rest of the paper is organised as follows. In section II, we present operational models for devices and households, and formulate the aggregation problem. In section III, we present a column-generation algorithm to solve the aggregation problem in a distributed fashion. Finally, computational results are reported in section IV.

Unless specified otherwise, durations are expressed in hours (h), power and energy quantities in kilowatt (kW) and kilowatt-hour (kWh) respectively, and temperatures in degrees Celsius. Energy prices are given in dollars per kilowatt-hour ($/kWh).

II. Problem formulation

We consider a set $\mathcal{R} = \{1, ..., R\}$ of households. Each household r contains a set of devices $\mathcal{D}_r = \{1, ..., D_r\}$, whose operation is scheduled over a time horizon $\mathcal{T} = \{0, ..., T-1\}$, with Δ_τ the duration of each time step. Households interact with an aggregator a, in order to minimize the total cost of purchasing electricity from the grid. The aggregator and households are assumed to behave rationally, and households dot not interact directly with each others. Finally, the environment is assumed to be deterministic.

A. Device constraints

In what follows, $p_{d,t}$ denotes the algebraic power consumed by a device d during time step t. A negative consumption indicates the device generates power. The sequence $\mathbf{p}_d = (p_{d,0}, \cdots, p_{d,T-1})$ is called the device's load profile. Following the classification in [24], devices are grouped into six classes: uncontrollable loads, curtailable loads, uninterruptible loads, deferrable loads, thermal loads, and energy storage.

Uncontrollable loads include devices whose operation cannot be altered. Such a device d consumes a deterministic amount of power $(\tilde{P}_{d,0}, ..., \tilde{P}_{d,T-1})$. Therefore, its load profile is simply

$$p_{d,t} = \tilde{P}_{d,t}. \qquad \forall t \in \mathcal{T} \tag{1}$$

Curtailable loads refer to devices whose operation at a given time t can be altered, independently of past and future operations. The operation of a curtailable load d is modelled as

$$p_{d,t} - u_{d,t}\tilde{P}_{d,t} = 0, \qquad \forall t \in \mathcal{T} \tag{2}$$

$$u_d \in \mathbb{B}, \tag{3}$$

where $(\tilde{P}_{d,0}, ..., \tilde{P}_{d,T-1})$ is the device's load profile in the absence of curtailment. An on-off constraint is modelled through binary variables $u_{d,t}$. When the device is off, the binary variable $u_{d,t}$ takes value 0, which forces $p_{d,t} = 0$. If the device is on, then $u_{d,t} = 1$ and $p_{d,t} = \tilde{P}_{d,t}$. Continuous curtailment is modelled by relaxing the integrality constraint on u_d. This model can be further extended by considering several operation modes, each with minimum and maximum power levels.

Uninterruptible loads are comprised of devices such as dishwashers and clothes dryers, whose operation is typically composed of a finite number of cycles. A cycle's start-up time is flexible but, once started, it cannot be interrupted. For simplicity, we consider a single cycle of duration $L_d \leq T$, with power requirement $(\tilde{P}_{d,0}, ..., \tilde{P}_{d,L-1})$. The device's operation is then defined by

$$p_{d,t} - \sum_{l=0}^{L-1} v_{d,t-l}\tilde{P}_{d,l} = 0, \qquad \forall t \in \mathcal{T} \tag{4}$$

$$\sum_{t=0}^{T-L} v_{d,t} = 1, \tag{5}$$

$$v_d \in \mathbb{B}, \tag{6}$$

where $v_{d,t}$ takes value 1 if the cycle is started at time t, and 0 otherwise. The device's load profile is given by (4), while (5) ensures the cycle is started exactly once. For readability, we use the convention $v_{d,t} = 0, \forall t < 0$. This framework naturally extends to several cycles with precedence constraints as proposed in [25].

Deferrable loads include devices such as an electric vehicle's (EV) charger, whose operation may be interrupted and resumed later. However, contrarily to curtailable loads, consumption may be shifted to an earlier or later period of time. Deferrable loads are modelled as follows:

$$e_d^{\text{tot}} = \Delta_\tau \sum_{t \in \mathcal{T}} p_{d,t}, \tag{7}$$

$$E_d^{\min} \leq e_d^{\text{tot}} \leq E_d^{\max}, \tag{8}$$

$$u_{d,t}P_d^{\min} \leq p_{d,t} \leq u_{d,t}P_d^{\max}, \qquad \forall t \in \mathcal{T} \tag{9}$$

$$u_d \in \mathbb{B}. \tag{10}$$

The device's total energy consumption e_d^{tot} is given by (7) and must be between minimum and maximum levels as stated in (8). On-off constraints are modelled by (9)-(10).

Thermal loads encompass devices like space heaters and air conditioners, that aim at keeping a system's (e.g., a room) temperature within a certain range. Their operation is modelled by

$$\Theta_d^{\min} \leq \Theta_{d,t} \leq \Theta_d^{\max}, \qquad \forall t \in \mathcal{T} \tag{11}$$

$$\frac{\mu}{c}(\Theta_{d,t}^{\text{ext}} - \Theta_{d,t}) + \frac{\eta}{c}p_{d,t} = \frac{\Theta_{d,t+1} - \Theta_{d,t}}{\Delta_\tau}, \qquad \forall t \in \mathcal{T} \tag{12}$$

$$u_{d,t}P_d^{\min} \leq p_{d,t} \leq u_{d,t}P_d^{\max}, \qquad \forall t \in \mathcal{T} \tag{13}$$

$$u_d \in \mathbb{B}. \tag{14}$$

The system's temperature $\Theta_{d,t}$ must be kept within an acceptable range as stated in (11). Its evolution is given by the first-order approximation in (12) of a thermodynamic model, where μ is a conduction coefficient, η is the device's thermal efficiency, and c is the system's heat capacity. Finally, the device's on-off constraints are modelled in (13)-(14). This simple model may be extended to include, for example, time-varying temperature requirements.

Energy storage devices can store energy and release it later. For simplicity, we consider the case of batteries, for which an operational model is

$$E_d^{\min} \leq e_{d,t} \leq E_d^{\max}, \qquad \forall t \in \mathcal{T} \quad (15)$$
$$p_{d,t} - p_{d,t}^{\text{ch}} + p_{d,t}^{\text{dis}} = 0, \qquad \forall t \in \mathcal{T} \quad (16)$$
$$\Delta_\tau \left(\eta_d^{\text{ch}} p_{d,t}^{\text{ch}} - \frac{1}{\eta_d^{\text{dis}}} p_{d,t}^{\text{dis}} \right) = e_{d,t} - e_{d,t-1}, \qquad \forall t \in \mathcal{T} \quad (17)$$
$$u_{d,t}^{\text{ch}} P_d^{\text{ch, min}} \leq p_{d,t}^{\text{ch}} \leq u_{d,t}^{\text{ch}} P_d^{\text{ch, max}}, \qquad \forall t \in \mathcal{T} \quad (18)$$
$$u_t^{\text{dis}} P_d^{\text{dis, min}} \leq p_{d,t}^{\text{dis}} \leq u_{d,t}^{\text{dis}} P_d^{\text{dis, max}}, \qquad \forall t \in \mathcal{T} \quad (19)$$
$$u_{d,t}^{\text{ch}} + u_{d,t}^{\text{dis}} \leq 1, \qquad \forall t \in \mathcal{T} \quad (20)$$
$$u_d^{\text{ch}}, u_d^{\text{dis}} \in \mathbb{B}. \quad (21)$$

The battery's state of charge is denoted $e_{d,t}$, while $p_{d,t}^{\text{ch}}$ and $p_{d,t}^{\text{dis}}$ denote charging and discharging power, respectively. The internal dynamics of the battery are captured by (17), where η_d^{ch} and η_d^{dis} are charging and discharging efficiencies. Finally, on-off constraints (18)-(21) ensure that the battery cannot be simultaneously charged and discharged.

Finally, renewable generation can be modelled as a negative load. Depending on systems' specifications, it may be either uncontrollable or curtailable.

B. Household and aggregator constraints

We now consider a given household $r \in \mathcal{R}$ and denote $p_{r,t}$ its net load at time t. Additional constraints at the household level are formulated as

$$P_r^{\min} \leq p_{r,t} \leq P_r^{\max}, \qquad \forall t \in \mathcal{T} \quad (22)$$
$$p_{r,t} = \sum_{d \in \mathcal{D}_r} p_{d,t}. \qquad \forall t \in \mathcal{T} \quad (23)$$

Constraints (22) state that the household's net load $p_{r,t}$, as defined in (23), is bounded below and above. This is most typically related to a circuit breaker's operating range.

Similarly, the aggregator must ensure that the aggregated load $p_{a,t}$ remains within physical limitations

$$P_a^{\min} \leq p_{a,t} \leq P_a^{\max}, \qquad \forall t \in \mathcal{T} \quad (24)$$
$$p_{a,t} = \sum_{r \in \mathcal{R}} p_{r,t}. \qquad \forall t \in \mathcal{T} \quad (25)$$

If households were operated independently, i.e., without interacting with the aggregator, constraints (24) may be violated. In practice, this could result in equipment damage, or localised blackouts.

C. Aggregation problem

The aggregation problem consists here in minimizing the total cost of purchasing energy from the grid, while satisfying all operational constraints. It is formulated as the following Mixed-Integer Linear Program (MILP):

$$\min \quad \sum_{t \in \mathcal{T}} \Pi_t \Delta_\tau p_{a,t} \quad (26)$$
$$\text{s.t.} \quad P_a^{\min} \leq p_{a,t} \leq P_a^{\max} \qquad \forall t \quad (27)$$
$$p_{a,t} = \sum_{r \in \mathcal{R}} p_{r,t} \qquad \forall t \quad (28)$$
$$P_r^{\min} \leq p_{r,t} \leq P_r^{\max} \qquad \forall r, t \quad (29)$$
$$p_{r,t} = \sum_{d \in \mathcal{D}_r} p_{d,t} \qquad \forall r, t \quad (30)$$
$$\cdots \qquad \forall r, \forall d \in \mathcal{D}_r \quad (31)$$

with Π_t being the price of electricity at time t. For ease of reading, we do not explicitly re-write device-specific constraints in (31).

We now discuss how to write the aggregation problem (26)-(31) in a more general form. For each household r, let x_r be the vector obtained by concatenating all decision variables specific to that household. Here, x_r would be composed of a household's net load, plus the decision variables of each device in that household. In what follows, we refer to x_r as the operational schedule of household r. Operational constraints for household r and its devices, corresponding to (29)-(31), are then written $D_r x_r = e_r$ without loss of generality. Integrality requirements are written $x_r \in X_r$. Finally, a household's operating cost is written $c_r^T x_r$ for a given cost vector c_r.

Similarly, let y denote the vector of decision variables that are specific to the aggregator. For the case at hand, y corresponds to the aggregated load $p_{a,t}$. The aggregator's operating cost is written $q^T y$, while constraints (27)-(28) are written

$$My + \sum_r A_r x_r = b.$$

The aggregation problem can therefore be written in the general compact form

$$\min_{y, x_1, \ldots, x_R} \quad q^T y + \sum_{r \in \mathcal{R}} c_r^T x_r \quad (32)$$
$$\text{s.t.} \quad My + \sum_{r \in \mathcal{R}} A_r x_r = b \quad (33)$$
$$B_r x_r = e_r \qquad \forall r \quad (34)$$
$$x_r \in X_r. \qquad \forall r \quad (35)$$

Constraints (33) induce a coupling between the households, and are thus referred to as linking constraints. Conversely, constraints (34)-(35) are separable by household, and are referred to as local constraints. This formulation is not restricted to the aforementioned use case, but allows to integrate any type of distributed resource r, whose operation can be formulated within a MILP. Likewise, more elaborate constraints and objective may be considered for the aggregator.

Mixed-integer linear programs such as (32)-(35) can be solved to proven optimality using standard optimization software. Although MILPs are NP-hard in general, practical instances can often be solved efficiently. Indeed, several of our test instances, with up to hundreds of thousands of variables,

were solved to optimality in a few minutes with a state-of-the art solver. Nevertheless, a centralized formulation obviously becomes intractable when dealing with large systems, both due to memory requirements and the eventual curse of dimensionality.

III. Distributed Column-Generation framework

We now present a decentralized framework, to solve the aggregation problem to global optimality. To underline the generality of the proposed methodology, we use the notation introduced in II-C.

A. Dantzig-Wolfe decomposition

For each resource $r \in \mathcal{R}$, define the set of its feasible operational schedules as

$$\mathcal{X}_r := \{x_r | B_r x_r = e_r, x_r \in X_r\}. \tag{36}$$

For simplicity, we assume $\mathcal{X}_r$ is bounded. Consequently, its convex hull $\text{conv}(\mathcal{X}_r)$ is a polytope, whose finite set of extreme points is denoted by Ω_r. The Minkowski-Weil theorem then allows to express x_r as a convex combination of these extreme points

$$x_r = \sum_{\omega \in \Omega_r} \theta_{r,\omega} \omega, \tag{37}$$

where the weights $\theta_{r,\omega}$ are non-negative and sum to 1.

This change of variable leads to the following extended formulation:

$$\min_{y,\theta} \quad q^T y + \sum_{r \in \mathcal{R}, \omega \in \Omega_r} c_{r,\omega} \theta_{r,\omega} \tag{38}$$

$$s.t. \quad My + \sum_{r \in \mathcal{R}, \omega \in \Omega_r} \theta_{r,\omega} a_{r,\omega} = b \tag{39}$$

$$\sum_{\omega \in \Omega_r} \theta_{r,\omega} = 1 \qquad \forall r \tag{40}$$

$$\theta \geq 0 \tag{41}$$

$$\sum_{\omega \in \Omega_r} \theta_{r,\omega} \omega = x_r \qquad \forall r \tag{42}$$

$$x_r \in X_r, \qquad \forall r \tag{43}$$

with the notation $c_{r,\omega} = c_r^T \omega$ and $a_{r,\omega} = A_r \omega$. The objective (38) and linking constraints (39) are simply re-writing of (32) and (33), respectively. Thereby, each extreme point ω of $\text{conv}(\mathcal{X}_r)$ is associated to a variable $\theta_{r,\omega}$, and a column $a_{r,\omega}$ which corresponds to a load schedule for resource r. The local operation constraints (34) in the original formulation are now implicit, through the definition of each Ω_r and the convexity constraints (40), (41). Finally, integrality constraints (43) are formulated on the original variables x_r, so that the equivalence with the compact formulation (32)-(35) is preserved. In what follows however, integrality constraints are relaxed, and the resulting problem (38)-(41) is referred to as the master problem.

Unlike in to a centralized formulation, the structure of the master problem is now independent of the nature of the resources. Indeed, the linking constraints only involve variables corresponding to a resource's net load. Therefore, using DW decomposition yields a technology-agnostic formulation.

B. Distributed column generation

Compared to the compact formulation, the master problem contains fewer constraints, but an exponentially large number of variables. Column generation is an iterative method wherein only a limited subset of variables is considered in a restricted master problem. Additional variables are generated at each iteration by solving an auxiliary pricing sub-problem. We focus here on the methodology's core structure, and refer to [26] for more advanced considerations on column generation and branch-and-price algorithms, as well as their relation to Lagrangian relaxation.

First consider the Restricted Master Problem (RMP)

$$\min_{y,\theta} \quad q^T y + \sum_{r \in \mathcal{R}, \omega \in \bar{\Omega}_r} c_{r,\omega} \theta_{r,\omega} \tag{44}$$

$$\text{s.t.} \quad My + \sum_{r \in \mathcal{R}, \omega \in \bar{\Omega}_r} \theta_{r,\omega} a_{r,\omega} = b \tag{45}$$

$$\sum_{\omega \in \bar{\Omega}_r} \theta_{r,\omega} = 1 \qquad \forall r \tag{46}$$

$$\theta \geq 0, \tag{47}$$

where $\bar{\Omega}_r \subset \Omega_r$ denotes, for each resource r, the subset of columns that are currently considered. The RMP is initialized with a small number of columns, some of which may be artificial to ensure feasibility.

At the beginning of each iteration, the RMP is solved to optimality. Let π denote the vector of dual variables associated to linking constraints (45), and σ_r the dual variable associated to convexity constraint (46) for resource r. For given r and $\omega \in \Omega_r$, the reduced cost of $\theta_{r,\omega}$ is then

$$\bar{c}_{r,\omega} = c_r^T \omega - \pi^T A_r \omega - \sigma_r. \tag{48}$$

Therefore, a variable θ_{r,ω^*} with smallest reduced cost is given by the pricing step

$$\omega^* \in \arg\min_{\omega \in \Omega_r} \left(c_r^T \omega - \pi^T A_r \omega - \sigma_r \right). \tag{49}$$

However, explicitly iterating over all Ω_r is prohibitively expensive. Nevertheless, since each $\omega \in \Omega_r$ is an extreme point of $\text{conv}(\mathcal{X}_r)$, performing the pricing step (49) is equivalent to solving

$$\omega^* \in \arg\min_{x_r} \quad (c_r^T - \pi^T A_r) x_r - \sigma_r \tag{50}$$

$$\text{s.t.} \quad B_r x_r = e_r \tag{51}$$

$$x_r \in X_r. \tag{52}$$

The pricing sub-problem (50)-(52) is a small MILP, which we assume can be solved efficiently. If the identified variable θ_{r,ω^*} has negative reduced cost, it is added to the RMP. Otherwise, all variables $\theta_{r,\omega}, \omega \in \Omega_r$ have non-negative reduced cost. Optimality in the master problem is reached when this is the case for all resources, i.e., all variables in the master problem have non-negative reduced cost.

Moreover, since all sub-problems are independent of each other, they can be solved in parallel, thus yielding a distributed algorithm. At each iteration, the aggregator solves the RMP and broadcasts the dual variables to the resources. Each resource then computes a vector ω_r^* with smallest reduced

Table I
DEVICES CLASSIFICATION AND OWNERSHIP RATES

Device	Classification	Own. rate (%)
Dishwasher	Uninterruptible load	65
Clothes washer	Uninterruptible load	90
Clothes dryer	Uninterruptible load	75
Electric heating	Thermal load	60
Electric vehicle	Deferrable load	ξ^{dep}
Home battery	Energy storage	ξ^{dep}
Rooftop solar	Curtailable load	ξ^{dep}
Others	Uncontrollable load	100

cost, and columns with negative reduced cost are added to the RMP.

Finally, a branch-and-price algorithm is generally used to solve the extended formulation (38)-(43), with the master problem being the root node of the branching tree. Nevertheless, we implemented a heuristic procedure to recover feasible solutions from the root solution, similar to a depth-first strategy. At each step, a small, random subset of the resources is selected, and the associated integer variables are rounded. The corresponding relaxation is then solved by column-generation, and the procedure is repeated until all integer variables are fixed. This recovery heuristic was found to obtain optimal integer solutions for all test instances. Therefore, we did not formally implement a full branch-and-price scheme.

IV. COMPUTATIONAL RESULTS

We now report computational results for the proposed column-generation algorithm (CG), using the centralized formulation (MILP) as a baseline. Tests are carried out on a set of 1120 instances, generated using data from Ontario energy markets. We first describe our test methodology in IV-A and IV-B. Numerical results are analysed in IV-C, and the robustness of the formulation is further assessed in IV-D.

A. Numerical instantiation

For all simulations, the time-horizon begins at 5am Monday, January 18^{th} 2016, and its length T ranges from 24h to 96h. The duration of a time-step is $\Delta_\tau = 1$h. The number of considered households, R, varies between 1024 and 8192. Ontario's provincial load, production and pricing data are obtained from [27].

The set of considered devices and their ownership rates are given in Table I (adapted from [24]). The ownership rate of a device is interpreted as the probability that that device be present in a given household. Devices that are not explicitly considered in this work are aggregated into one uncontrollable load for each household. In addition, we consider correlation among devices. It is assumed that only households with a clothes washer own a clothes dryer. Because available data regarding the penetration of EVs, batteries and rooftop solar in Ontario is very scarce, we assumed that households own either none or the three of them. Several deployment scenarios are considered, corresponding to ownership rates $\xi^{\text{dep}} = 0\%, 33\%, 66\%$ and 100%.

Dishwashers, clothes washers and clothes dryers are modelled as uninterruptible loads. Each appliance's operation consists of one cycle per day, which consumes a constant power $\tilde{P} = 1$. Cycles' durations are 2h for dishwashers, 2h for clothes washers, and 3h for clothes dryers.

For electric heating systems, the outside temperature Θ^{ext} is obtained by perturbing a reference profile Θ^{ref} that consists of hourly readings from a weather station in the Toronto area [28]

$$\Theta^{\text{ext}}_{d,t} = \Theta^{\text{ref}}_t + 0.5 \times \varepsilon_{d,t}, \qquad \forall t \in \mathcal{T}$$

with $\varepsilon \sim \mathcal{N}(0,1)$. Numerical parameters for the thermodynamics model are identical among households: $\eta = 1$, $\mu = 0.2$, $c = 3$, and $P^{\min} = 0$, $P^{\max} = 10$. The inside temperature must be kept in the range $[\Theta^{\min}_d, \Theta^{\max}_d] = [18, 22]$.

The charging of electric vehicles is modelled as a deferrable load with a daily energy requirement of $E^{\min}_d, E^{\max}_d = 10$ as reported in [24]. Charging must happen between 8pm and 5am, and charger limitations are $P^{\min}_d = 1.1$, $P^{\max}_d = 7.7$.

Battery specifications are based on the Tesla powerwall [29]. Energy capacity is $E^{\max}_d = 13.5$, with minimum state of charge $E^{\min}_d = 0$. Charging and discharging limitations are $P^{\text{ch, min}}_d, P^{\text{dis, min}}_d = 0$ and $P^{\text{ch, max}}_d, P^{\text{dis, max}}_d = 5$. Efficiencies are $\eta^{\text{ch}}_d, \eta^{\text{dis}}_d = 0.95$, yielding a 90% round-trip efficiency.

Rooftop solar is modelled as a (negative) curtailable load. The output of a PV system d is given by

$$\tilde{P}^{\text{PV}}_{d,t} = \gamma_r \times \tilde{Q}^{\text{PV}}_t \times \zeta_{d,t}, \qquad \forall t \in \mathcal{T}$$

where the normalized output $\tilde{Q}^{\text{PV}}_t$ is Ontario's hourly PV output Q^{PV}_t, divided by its average value over the considered time period. The household-specific scaling factor γ_r is drawn from a uniform distribution $\mathcal{U}(0.5, 1.5)$, and $\zeta \sim \mathcal{U}(0,1)$ is a random noise.

For each household r, the corresponding uncontrollable load d has the following load profile:

$$\tilde{P}_{d,t} = \gamma_r \times \max\left(0, \tilde{Q}^{\text{Ont}}_t + 0.05 \times \varepsilon_{d,t}\right), \qquad \forall t \in \mathcal{T}$$

where $\tilde{Q}^{\text{Ont}}$ is the normalized Ontario hourly provincial load, and $\varepsilon \sim \mathcal{N}(0,1)$ is a white noise.

Finally, we set $P^{\min}_r = 0$, $P^{\max}_r = 10$ for the households' net load constraints. Similarly, the total aggregated load is bounded by $P^{\min}_a = 0$ and $P^{\max}_a = 7.5 \times R$. The price of electricity Π_t is the Hourly Ontario Energy Price (HOEP).

B. Implementation details

Experiments were performed on a 2×Xeon E5-2650V4 2.2Ghz, 254 GB RAM computer running Linux. Our implementation was coded in Python 2.7, with CPLEX 12.7 as the linear solver. All CPLEX runs used default parameters and a single thread. Accordingly, an instance is considered solved to optimality when the reported optimality gap is smaller than 10^{-4}, which is the default threshold for CPLEX.

In order to smooth out performance variations, we generated ten different instances for each tuple (R, T, ξ^{dep}). This resulted in a testbed of 1120 instances, and we compare the results obtained by the proposed method (CG), and the centralized

formulation (MILP). For the MILP, we used CPLEX with default parameters, a single thread and a time limit of one hour. All instances were found to be feasible. Due to limited computing resources, the column-generation sub-problems were solved serially rather than in parallel. Nevertheless, the duration of each iteration in the distributed setting is given by the RMP computing time, plus the maximum solving time among sub-problems, plus computation overheads. In practice, each sub-problem would indeed be solved locally by the corresponding household.

Finally, the RMP is initialized by computing, for each resource, the column corresponding to a minimum-peak load schedule. In order to ensure feasibility in the master, slack and surplus variables are added to linking constraints. These artificial variables have sufficiently large cost, and are thus automatically set to zero once a feasible solution is found. Furthermore, we used a partial pricing strategy that consists in adding at most $0.1 \times R$ columns to the master at each iteration. The 0.1 ratio was found to achieve good performance across a wide range of instances. Similarly, for the recovery heuristic, a random 10% of the resources are selected at each step.

C. Results analysis

Performance statistics are presented for MILP and CG in Tables II and III respectively. Since both methods behaved consistently across the different values of ξ^{dep}, we only report results for the case $\xi^{\text{dep}} = 0.66$. For the MILP formulation, Table II reports the number of binary and continuous variables (Bin. and Cont. respectively, in millions) of the problem, the proportion of solved instances, total and root computing times (in seconds), the number of nodes in the branch-and-bound tree, and the root gap (in %). For the CG method, Table III reports the number of columns generated (Col., in thousands), the total number of CG iterations to reach optimality (Iter.), the proportion of instances solved to proven optimality, computing times (in seconds), and the root gap (in %). Root gaps are given by

$$\text{gap} = \frac{|z^* - \underline{z}|}{|z^*|},$$

where z^* is the value of the best known integer solution found by either algorithm, and $\underline{z}$ is the value of the root relaxation. Finally, all reported averages are geometric means, except for the proportion of solved instances.

As expected, CPLEX solved the MILP formulation for smaller instances, but systematically reached the time limit for the larger ones. More specifically, CPLEX failed to find a feasible solution for 162 instances, roughly corresponding to the instances with more than two million binary variables. Note that the quality of the lower bound for MILP is not an issue here. On the opposite, the solver's capability of exploring the branch-and-bound nodes in a reasonable amount of time, in order to provide good feasible solutions, is affected by the problem's size. In comparison, CG was able to solve all 1120 instances to proven optimality. Furthermore, computing times for both methods are displayed in Figure 1. For all values of T, CG exhibits a more scalable behaviour than MILP,

Table II
MILP STATISTICS FOR $T = 24$, $\xi^{\text{DEP}} = 0.66$

R	Variables (M) Bin.	Cont.	Solved (%)	Time (s) Root	Total	B&B nodes	Root % gap
1024	0.13	0.19	100	9.6	40.1	1.5	0.00
1536	0.19	0.29	100	20.6	132.9	11.5	0.00
2048	0.26	0.39	100	30.2	291.0	108.1	0.00
3072	0.39	0.57	100	65.4	649.8	394.4	0.00
4096	0.52	0.77	100	147.3	1263.7	562.8	0.00
6144	0.78	1.15	70	314.8	2907.1	550.1	0.00
8192	1.04	1.54	60	534.6	2937.4	664.3	0.00

Table III
CG STATISTICS FOR $T = 24$, $\xi^{\text{DEP}} = 0.66$

R	Col. (k)	Iter.	Solved (%)	Time (s) Master	Pricing	Total	Root % gap
1024	5.3	45.7	100	0.6	1.7	3.8	0.00
1536	8.0	46.0	100	1.1	1.7	5.1	0.00
2048	10.6	46.1	100	1.8	1.8	6.8	0.00
3072	16.2	46.6	100	3.8	1.9	10.6	0.00
4096	21.6	47.2	100	6.4	2.0	15.0	0.00
6144	31.7	46.3	100	13.5	2.0	25.2	0.00
8192	43.4	47.6	100	25.1	2.1	40.6	0.00

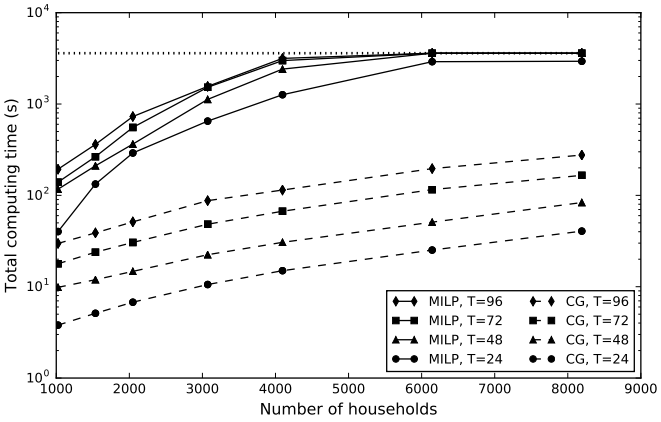

Figure 1. Average computing times for CG and MILP formulations, with $\xi^{\text{dep}} = 0.66$. The horizontal dotted line depicts the one-hour time limit for MILP.

and achieves speed-ups of to two orders of magnitude. These results confirm that, even if it may be tractable for small-size systems, a centralized approach fails to handle large numbers of resources.

To further analyse the scalability of CG, Figure 2 shows the number of CG iterations to reach convergence, corresponding to the number of times the RMP is solved. Here, the number of iterations is more relevant than raw computation times, since the latter depends on machine specifications and on the solver's performance. On one hand, an increase in the length of the time-horizon T leads, as expected, to an increase in the

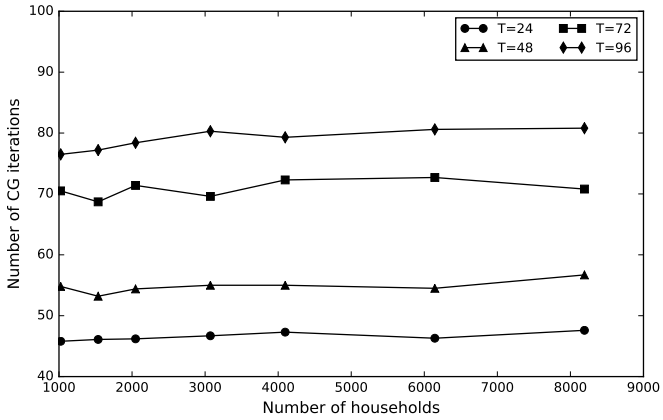

Figure 2. Average number of CG iterations, including the recovery heuristic. $\xi^{\text{dep}} = 0.66$.

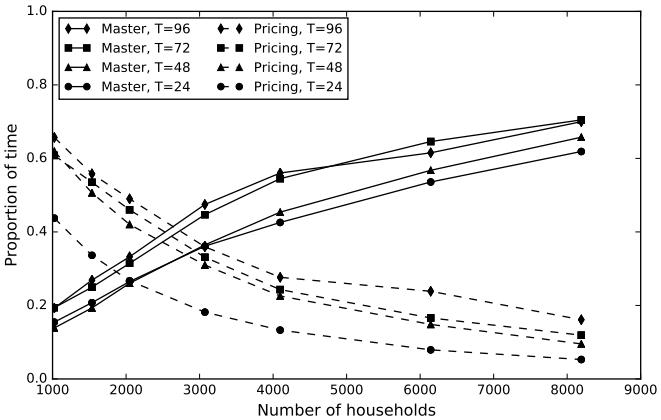

Figure 3. Time spent solving the master problem and pricing sub-problems, as a fraction of total computing time. $\xi^{\text{dep}} = 0.66$.

number of iterations. On the other hand, as was hinted at in Table III, the number of iterations appears to be independent of the number of households. This remarkable behaviour is due to similarities between sub-problems. This explanation is corroborated by the fact that larger values of ξ^{dep}, which only affects the distribution of households, and thus of sub-problems, resulted in more iterations. Overall, CG is sensitive to the distribution of resources, rather than their number.

Finally, the proportions of time spent solving the master problem and sub-problems, respectively, are displayed in Figure 3. Computation overheads can be significant, but they are highly dependent on the implementation. Therefore, were factored them out of the plots. Clearly, as the number of resources increases, most of the time is spent solving the RMP. Indeed, since sub-problems are solved in parallel, the number of resources has little influence on the duration of the pricing step. This is consistent with computing times reported in Table III. Conversely, the size of the RMP is given by the number of columns generated. Therefore, as R increases, so does the size of the RMP and the associated computational cost. Consequently, solving the RMP is a computation bottleneck for CG, and currently constitutes the main limitation to the algorithm's scalability.

D. Further discussion

As shown in Tables II and III, root gaps for CG and MILP were often found to be lower than 10^{-4}, meaning both formulations have essentially zero integrality gap. This observation carries over to the rest of the dataset: for MILP, root gaps were always less than 0.02%, and smaller than 0.01% in 843 instances out of 1120. For CG, the largest recorded root gap was 0.008%. Moreover, root gaps tended to be smaller as the number of resources increased.

We now assess whether low integrality gaps reflect intrinsic properties of the problem at hand, or arise from our numerical data. More specifically, we increased the batteries' minimum power ratios $P^{\text{ch, min}}/P^{\text{ch, max}}$ (resp. $P^{\text{dis, min}}/P^{\text{dis, max}}$) from 0, as in our initial tests, to 90%. This is done by setting the value of $P^{\text{ch, min}}$ (resp. $P^{\text{dis, min}}$) accordingly. Figure 4 displays the resulting evolution of computing time (left axis) and integrality gap (right axis). Results are reported for $R = 1024$, $T = 24$ and $\xi^{\text{dep}} = 0.66$, and similar behaviour were observed for other settings. Figure 4 shows that the MILP integrality gaps increases considerably, from 0.005% to over 2%. This resulted in an increase in the number of branching nodes, which we do not report for lack of space, and in computing time. On the other hand, although the CG gap increased, it remained below 0.01%. Moreover, the increase in computing time for CG is caused only by longer solving times for sub-problems. The number of CG iterations did not increase, nor did solving time for the RMP. Overall, CG appears to be more robust than MILP. This robustness is explained by the fact that changes in the resources' operation only affect sub-problems for CG while, for MILP, the entire problem structure may be affected.

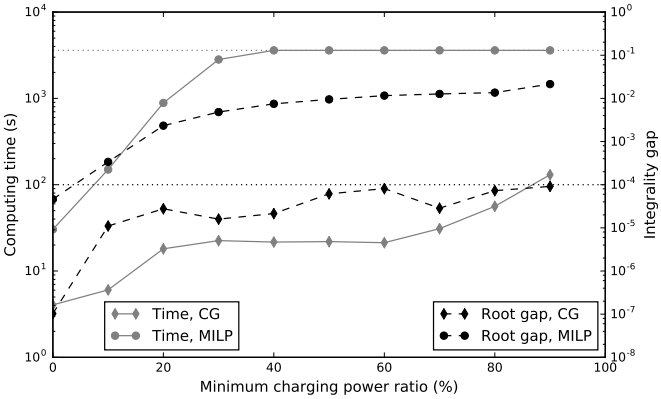

Figure 4. Average computing time (gray, left-hand axis scale) and root gap (black, right-hand axis scale) with respect to minimum charging power ratio. The dotted horizontal lines respectively denote the MILP time limit (in gray) and the 10^{-4} threshold for gaps (in black). Results obtained with $R = 1024$, $T = 24$ and $\xi^{\text{dep}} = 0.66$.

V. Conclusion

We have considered the problem of coordinating the operation of multiple households, for demand response applications in smart-grids. In particular, we focused on the challenges raised by discrete decisions in households' operation, such as on-off constraints. The resulting problem can be formulated as

a centralized MILP, however this approach is intractable for large systems.

In this work, we proposed an exact, distributed column-generation algorithm, for optimally coordinating households in a decentralized fashion. We showed that this framework is not restricted to households, but allows to integrate any type of resource whose operation can be written within a MILP. We demonstrated the practical efficiency and scalability of the proposed approach through extensive computational experiments, and provided mathematical insight on the algorithm's behaviour.

ACKNOWLEDGEMENTS

This work was supported by the NSERC Energy Storage Technologies Network (NEST Net).

REFERENCES

[1] M. H. Albadi and E. F. El-Saadany, "A summary of demand response in electricity markets," *Electric Power Systems Research*, vol. 78, no. 11, pp. 1989–1996, 2008.

[2] L. Gkatzikis, I. Koutsopoulos, and T. Salonidis, "The role of aggregators in smart grid demand response markets," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 7, pp. 1247–1257, 2013.

[3] P. Siano, "Demand response and smart grids—a survey," *Renewable and Sustainable Energy Reviews*, vol. 30, no. Supplement C, pp. 461 – 478, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1364032113007211

[4] R. Deng, Z. Yang, M. Y. Chow, and J. Chen, "A survey on demand response in smart grids: Mathematical models and approaches," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 3, pp. 570–582, June 2015.

[5] J. S. Vardakas, N. Zorba, and C. V. Verikoukis, "A survey on demand response programs in smart grids: Pricing methods and optimization algorithms," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 1, pp. 152–178, 2015.

[6] Z. Zhao, L. Wu, and G. Song, "Convergence of volatile power markets with price-based demand response," *IEEE Transactions on Power Systems*, vol. 29, no. 5, pp. 2107–2118, Sept 2014.

[7] J. Li, Z. Wu, S. Zhou, H. Fu, and X.-P. Zhang, "Aggregator service for pv and battery energy storage systems of residential building," *CSEE Journal of Power and Energy Systems*, vol. 1, no. 4, pp. 3–11, 2015.

[8] M. Parvania, M. Fotuhi-Firuzabad, and M. Shahidehpour, "Optimal demand response aggregation in wholesale electricity markets," *IEEE Transactions on Smart Grid*, vol. 4, no. 4, pp. 1957–1965, 2013.

[9] ——, "Iso's optimal strategies for scheduling the hourly demand response in day-ahead markets," *IEEE Transactions on Power Systems*, vol. 29, no. 6, pp. 2636–2645, 2014.

[10] Z. Zhu, J. Tang, S. Lambotharan, W. H. Chin, and Z. Fan, "An integer linear programming based optimization for home demand-side management in smart grid," in *2012 IEEE PES Innovative Smart Grid Technologies (ISGT)*, Jan 2012, pp. 1–5.

[11] P. Chavali, P. Yang, and A. Nehorai, "A distributed algorithm of appliance scheduling for home energy management system," *IEEE Transactions on Smart Grid*, vol. 5, no. 1, pp. 282–290, Jan 2014.

[12] T. Logenthiran, D. Srinivasan, and T. Z. Shun, "Demand side management in smart grid using heuristic optimization," *IEEE Transactions on Smart Grid*, vol. 3, no. 3, pp. 1244–1252, Sept 2012.

[13] K. Paridari, A. Parisio, H. Sandberg, and K. H. Johansson, "Demand response for aggregated residential consumers with energy storage sharing," in *2015 54th IEEE Conference on Decision and Control (CDC)*, Dec 2015, pp. 2024–2030.

[14] M. A. A. Pedrasa, T. D. Spooner, and I. F. MacGill, "Coordinated scheduling of residential distributed energy resources to optimize smart home energy services," *IEEE Transactions on Smart Grid*, vol. 1, no. 2, pp. 134–143, Sept 2010.

[15] M. Kraning, E. Chu, J. Lavaei, S. Boyd *et al.*, "Dynamic network energy management via proximal message passing," *Foundations and Trends® in Optimization*, vol. 1, no. 2, pp. 73–126, 2014.

[16] J. Rivera, P. Wolfrum, S. Hirche, C. Goebel, and H. A. Jacobsen, "Alternating direction method of multipliers for decentralized electric vehicle charging control," in *52nd IEEE Conference on Decision and Control*, Dec 2013, pp. 6960–6965.

[17] N. Gatsis and G. B. Giannakis, "Residential demand response with interruptible tasks: Duality and algorithms," in *2011 50th IEEE Conference on Decision and Control and European Control Conference*, Dec 2011, pp. 1–6.

[18] ——, "Decomposition algorithms for market clearing with large-scale demand response," *IEEE Transactions on Smart Grid*, vol. 4, no. 4, pp. 1976–1987, Dec 2013.

[19] S. J. Kim and G. B. Giannakis, "Scalable and robust demand response with mixed-integer constraints," *IEEE Transactions on Smart Grid*, vol. 4, no. 4, pp. 2089–2099, Dec 2013.

[20] S. Mhanna, A. C. Chapman, and G. Verbic, "A fast distributed algorithm for large-scale demand response aggregation," *IEEE Transactions on Smart Grid*, vol. 7, no. 4, pp. 2094–2107, July 2016.

[21] P. M. Namara and S. McLoone, "Hierarchical demand response using dantzig-wolfe decomposition," in *IEEE PES ISGT Europe 2013*, Oct 2013, pp. 1–5.

[22] H. A. Toersche, A. Molderink, J. L. Hurink, and G. J. M. Smit, "Column generation based planning in smart grids using triana," in *IEEE PES ISGT Europe 2013*, Oct 2013, pp. 1–5.

[23] H. Harb, J.-N. Paprott, P. Matthes, T. Schütz, R. Streblow, and D. Müller, "Decentralized scheduling strategy of heating systems for balancing the residual load," *Building and Environment*, vol. 86, pp. 132 – 140, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0360132314004260

[24] M. Beaudin and H. Zareipour, "Home energy management systems: A review of modelling and complexity," *Renewable and Sustainable Energy Reviews*, vol. 45, pp. 318 – 335, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1364032115000568

[25] G. T. Costanzo, G. Zhu, M. F. Anjos, and G. Savard, "A system architecture for autonomous demand side load management in smart buildings," *IEEE Transactions on Smart Grid*, vol. 3, no. 4, pp. 2157–2165, Dec 2012.

[26] G. Desaulniers, J. Desrosiers, and M. M. Solomon, *Column generation*, 1st ed., ser. GERAD 25th anniversary. Springer Science & Business Media, 2006, vol. 5.

[27] IESO, "Power data." [Online]. Available: http://www.ieso.ca/power-data

[28] "Historical data," toronto City station, ID 6158355. [Online]. Available: http://climate.weather.gc.ca/historical_data/search_historic_data_e.html

[29] Tesla, "Powerwall specifications." [Online]. Available: https://www.tesla.com/support/powerwall

Miguel F. Anjos is Professor of Mathematics and Industrial Engineering of Polytechnique Montreal, where he holds the NSERC-Hydro-Quebec-Schneider Electric Senior Industrial Research Chair on Optimization for the Smart Grid, and the Inria International Chair on Power Peak Minimization for the Smart Grid.

His research interests are in mathematical optimization, and particularly its application in power systems and smart grids. His allocades include a Canada Research Chair, the Méritas Teaching Award, a Humboldt Research Fellowship, the title of EUROPT Fellow, and the Queen Elizabeth II Diamond Jubilee Medal. He is a fellow of the Canadian Academy of Engineering.

Andrea Lodi has been full professor of Operations Research at DEI, University of Bologna between 2007 and 2015. Since 2015 he is Canada Excellence Research Chair in "Data Science for Real-time Decision Making" at Polytechnique Montréal.

His main research interests are in Mixed-Integer Linear and Nonlinear Programming and Data Science and his work has received several recognitions including the IBM and Google faculty awards. He has been network coordinator and principal investigator of two large EU projects/networks, and, since 2006, consultant of the IBM CPLEX research and development team. He is the co-principal investigator of the project "Data Serving Canadians: Deep Learning and Optimization for the Knowledge Revolution" and scientific co-director of IVADO, the Montréal Institute for Data Valorization.

Mathieu Tanneau received the M.Sc from École polytechnique, France, in 2017, and is currently pursuing the Ph.D. degree at Polytechnique Montréal, Canada.

His research interests are in distributed methods for large scale mixed-integer optimization, as well as the integration of machine learning techniques within optimization frameworks.