Loads scheduling for demand response in energy communities

Eric Bourreau, Bernard Fortz, Amaury Pachurka, Michael Poss, Mariam Sangaré

Loads scheduling for demand response in energy communities

Eric Bourreau[1], Bernard Fortz[2,3], Amaury Pachurka[4], Michael Poss[1], and Mariam Sangaré[1]

[1]LIRMM, University of Montpellier, Montpellier, France, ✉ {eric.bourreau,michael.poss,msangare}@lirmm.fr
[2]Départemet d'informatique, Université libre de Bruxelles, Belgique, ✉ bernard.fortz@ulb.be
[3]INOCS, INRIA Lille-Nord Europe, France
[4]Smart Lou Quila, Gard, France, ✉ smartlouquila@gmail.com

Abstract

This paper focuses on optimizing collective self-consumption in an energy community composed of households and premises (stadium) by scheduling loads of electrical appliances owned by the members. The corresponding community remains connected to the public grid, and each member can produce and/or store photovoltaic energy. Furthermore, they can exchange this energy with the public grid or other energy community members. The proposed strategy aims at implementing a Demand Side Management program by taking advantage of the controllable loads' characteristics. A MILP formulation of the problem allows, on the one hand, to give the optimal planning of the operation of the electrical devices. On the other hand, it provides the optimal solutions for managing the storage units and the energy exchanges between community members and the public grid to minimize the energy flows from the public grid to the community over the time horizon. However, this MILP does not allow us to efficiently solve the large instances of the problem. Thus, we develop a column generation-based heuristic to find solutions for large problem instances. Our numerical experiments based on real data collected in the south of France, show that joining an energy community saves money on energy bills and reduces the total energy drawn from the primary grid by at least 15%.

keywords: Energy communities, Loads scheduling, Column generation heuristic

1 Introduction

The increasing environmental concerns and the European Union's commitment to sustainability have led to the approval of the Clean Energy Package by the European Commission in 2019 [5]. This package provides a regulatory framework for Renewable Energy Communities and Citizens Energy Communities. Instead of operating as a single Renewable self-consumer and dealing with the disadvantages of this status (loss of surplus energy, for example), economic agents evolving in the same locality can form an energy community and share advantages and disadvantages. Since the approval of the Clean Energy package, projects in member countries like France, Spain and Portugal are trying to implement the directives of the package. Therefore, we are assisting the multiplication of efforts by legislators, researchers, practitioners, and citizens to implement projects aimed at the generation, management, and consumption of locally produced electricity to support the energy transition.

1.1 Renewable Energy Communities

An energy community can be defined as a grouping of two or more physical or legal persons who combine their efforts to make the best use of the energy they produce locally. Here, we are interested in RECs. RECs involve groups of citizens, social entrepreneurs, public authorities, and community organizations participating directly in the energy transition by jointly investing in, producing, consuming, selling, and distributing renewable energy. Some benefits of RECs are: more efficient storage and sharing of green energy. Indeed, when a green energy producer operates alone, i.e., produces, consumes, and injects energy surplus into the primary grid, the energy surplus after charging the batteries is necessarily injected into the primary energy grid. Therefore, this energy may not be consumed locally. Being in the community

increases the storage capacities because instead of systematically injecting in the network, one can prefer to inject in the batteries of the community. In addition, community members can exchange energy at more attractive rates than those offered by the public grid. The authors of [16] have highlighted the advantages of sharing the outputs of the collective system between consumers. Specifically, the maximization of load matching through the aggregation of heterogeneous demand profiles induces a smooth load curve that can imply an increase in the self-consumption rate of about 20% [14]. Similarly, [17] shows that the aggregation of individual loads leads to a global load curve more adapted to the solar energy production profile and thus increases the collective self-consumption. The members of an energy community, unlike classical consumers, are therefore more involved in decision-making, especially in energy sharing.

We are witnessing a continuous increase in global energy demand. It is estimated that residential consumption alone accounts for 40% of global energy consumption [6]. Energy communities are an alternative way to reduce the carbon footprint of members. The expansion of energy communities would reduce the consumption of fossil fuels and increase the use of green energy. However, this requires the construction of smart buildings in which almost all devices are remotely manageable. It also involves the conduct of studies leading to the development of advanced solutions to support the paradigm shift from the traditional energy systems paradigm of "generation-follows-load" to the new paradigm of "load-follows-generation", which have led to the development of Demand Side management methodologies [4].

1.2 Demand side management

Demand-side management (DSM) is the planning, implementation, and monitoring (by a power system operator) of activities to induce a change in consumer behavior to modify its consumption profile. It is a function that allows the involvement of consumers in the management of smart grids through informed decision-making regarding their energy consumption, which helps the energy providers reduce the peak load demand and reshape the load profile [13]. Appeared in the 70s in response to rising energy costs, the concept has been approached in different ways. The first approach is based on an agreement between the consumer and the network operator. Through this agreement, the consumer authorizes the network operator to disconnect a selection of devices when the cumulative consumption reaches a threshold. This approach is called Direct Load Control (DLC), which may be effective in some cases but not in cases where the network operator manages several consumers for obvious reasons. Hence, the second approach is where the operator, through various mechanisms, encourages consumers to consume or not during specific periods (Decentralized Control, DC). Among these mechanisms are the energy prices that variate according to different parameters: the general state of the market, the global load level, and the global level of energy production. The consumer is thus periodically confronted with arbitration between consuming or making savings. Contrary to the DLC, DC allows the consumer to manage loads through the circulation of information. However, although advantageous for both parties, this second approach may not be very effective if it is not accompanied by the consumer's efficient scheduling of energy generation, storage, and consumption. Therefore, developing planning algorithms integrated into a management system would allow autonomous and optimal management of all production, consumption, and storage activities.

This paper focuses on community members' simultaneous planning, consumption, and storage activities. The community remains connected to the public grid, and the members may or may not own a green energy production system and energy storage system. Moreover, each member performs a set of controllable loads (i.e., with flexible and programmable execution) and non-controllable loads (i.e., with fixed power profile) [4] during a given time horizon. Finally, community members who generate photovoltaic energy can exchange their surplus with others and/or feed it into the public grid.

The main objective for the community is to maximize the auto-consumption of renewable energies, or in other words, to minimize the exchanges with the public grid. Therefore, we focus on scheduling members' loads by taking advantage of the specific characteristics of these loads.

1.3 Related works

Loads scheduling to optimize energy efficiency is a well-assimilated practice in the industrial context. It is the case, for example, of [9] where the authors study a multi-objective flow shop minimizing the tardiness and the total energy cost. Likewise, the literature on load scheduling in residential, microgrid, and energy communities is abundant. Typically, the approaches for solving the problems can be categorized into

two types: offline and online. In the online planning approach, the decision at period h is revealed at the beginning of that period. That approach, which requires increased consumer participation, may be ineffective because of the fatigue phenomenon described by [18]. In the offline approach, the optimization is performed once before the planning horizon starts. The members execute their tasks during horizon according to the returned scheme. Our work belongs to the class of offline approaches.

Energy communities can take several forms. There members may consist of prosumers, consumers, a mix of prosumers and consumers [10], whether these members have individual storage units or a central storage unit [22], only prosumers not having batteries [11], etc. As shown in Figure 1, the community considered here is composed of a mix of prosumers and consumers, and each of them may or may not own a storage unit. In addition, the works related to energy communities seek answers to various questions and the technical solving tools that are used strongly depend on these questions. These tools include linear programming, mixed-integer programming, bi-level programming, game theory.

When game theory or bi-level models are involved, it is often under the form of Stackelberg games. The latter consist of a leader and a follower, where the leader (level 1) is an entity similar to the primary grid that sets the energy buying/selling prices to optimize the gain (positive or negative) or incite the followers to prioritize green energy consumption. Then, the followers (level 2) make their decisions to optimize their earnings and ensure their comfort regarding the leader's decisions.

Among the works that adopt a bi-level approach, we have [11], where is presented a model for energy management in a community of prosumers and trading among peers. There is no individual storage system; instead, a centralized storage entity that can buy surplus generation exists. The energy surplus of individuals can be sold/purchased to a neighbor, the centralized storage entity, or the main grid. Similarly, [23] presents an energy system composed of energy suppliers and small microgrids. Microgrids can have batteries but require an energy supplier in case of production or demand surplus. The paper considers a supplier named Genco that interacts with the microgrids via power purchase/sale contracts. Genco aims to offer the best contracts to avoid losing customers to competitors. The microgrids seek to minimize the total cost of interactions with Genco. First, a deterministic model is formulated, then a stochastic model considering the uncertainty of intermittent energy sources. In [1], the authors consider an environment composed of an energy-sharing system involving a set of consumers and a system operator. The system operator's goal is to maximize its payoff. Consumers have loads that can be delayed but executed within specific time windows, with certain penalties. So, consumers' objective is to minimize their objective function composed of energy bills and inconvenience, which conflicts with the system operator's goal. In addition to the bi-level model, they propose a single-level reformulation and two heuristics that provide good solutions by relying on the structure of the problem.

Regarding game theory, [22] proposes a peer-to-peer trading scheme for a community of prosumers and consumers. Consumers draw their needs from the main network. The primary grid sets buying and selling prices so that prosumers have the least incentive to trade with it. The followers respond by forming coalitions of neighbours who will trade their surplus with those in need of energy. The paper studies the properties of the Stackelberg game and show in particular that prosumers have no interest in going it alone, that prosumers will only form two coalitions. They finally show that there is a stable Stackelberg equilibrium.

Turning to linear programming approaches, [10] seeks the percentage of prosumers to include in a community that maximizes the financial gain. To answer such a question, the authors introduce a cost-optimization model for peer-to-peer energy-sharing communities, making continuous decisions while using a forecast for the input data. Based on real-world data, they deduce that we can achieve cost-saving and increased collective self-consumption for small communities of 2-5 members. Similarly, [21] proposes a data-driven flexibility optimizer model for day-ahead energy profile scheduling. First, this optimizer estimates the green energy production and energy consumption of heat pumps and cooling appliances, using a prediction model based on the fully connected neural network architecture. Then, an optimization problem is formulated to minimize flexibility procurement to reduce peak demand and increase green energy consumption. Finally, day-ahead scheduling is performed according to the optimization model results. Also, [7] proposes load planning in hospital, sensitive to electrical failures. The network comprises hospital beds, PV generation, and repurposed EV storage system. The presented model is tested under different scenarios; the results show that electricity bill reduction of 9.4% and energy reduction drawn from the main grid are achievable.

1.4 Contributions

In this work, we seek to optimize collective self-consumption by scheduling the controllable electric devices' loads of community members, energy exchanges between peers, and battery storage. Unlike many models from the literature, where one seeks to optimize profit, this work aims at optimizing the energy drawn from the public grid. In other words, we wish to minimize the consumption of non-green energy.

The problem is modelled as a Mixed-Integer Linear programming Model (MILP), which, in addition to returning optimal schedules, gives a plan of energy exchanges and a management scheme for electricity storage units installed in the community during the considered planning horizon. Our starting is the model proposed in [15], which we significantly enrich by considering more realistic modelling of the temperature evolution in the rooms and water heaters, following Newton's cooling law. As the resulting MILP is hardly solvable to optimality, even for small instances, we developed a column generation-based heuristic that convexifies the set of feasible schedules for the heating tasks.

Experiments reported in this paper show that the amount of energy drawn from the network is reduced by at least 15% when individuals operate as a community compared to the cumulative energy drawn from the primary grid when they act individually. They also show that the MILP method is more efficient for solving small problem instances, while the heuristic provides high-quality solutions for large problem instances.

The remainder of the paper is organized as follows: after describing the problem under study in Section 2, Section 3 presents the mixed-integer programming model to acquire the exact solutions to the problem. Section 4 presents the heuristic based on column generation, returning the efficient solution of large problem instances. Section 6 presents the experimental results. The paper concludes with a conclusion and perspectives for future works in Section 8.

2 Problem description

We study the following planning problem in this paper: Consider a set of N agents (residences, businesses, schools, stadiums, etc.) forming an energy community. Each community member may or may not have power generation and energy storage assets. During a time horizon H sliced into periods of equal length,

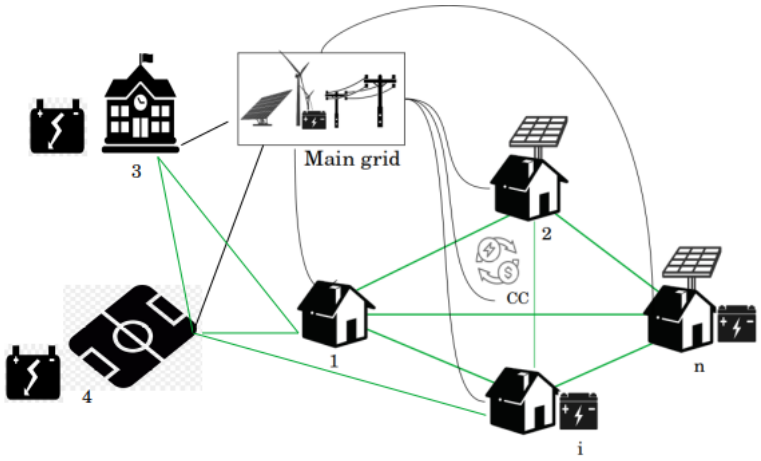


Figure 1: Community architecture (CC represents the Community Coordinator).

each member wishes to perform a set of tasks with electrical devices. Each task belongs to one of the following categories based on its characteristics.

2.1 Loads categorization

According to [4], controllable loads have flexible and programmable operations. The controllable loads are grouped into two categories: type A, B loads. The reorganization of controllable loads over time allows increasing the energy efficiency of the community. We use the same loads nomenclature as [15].

Type A loads are related to tasks whose execution allows regulating the temperature of certain environments to ensure human comfort. They include heating, ventilation, air conditioning, and water heaters. The execution of these tasks may be interrupted for certain periods before being resumed while ensuring that the targeted temperature objectives are met. In practice, an individual regulates the temperature of a room to reach a comfort zone and maintains this comfort zone until a specific time of the day. For example, an individual may want the temperature of a space to be between 22 °C and 24 °C from 6 pm to 11 pm. The solution proposed in this paper for scheduling type A loads determines the starting time and periodic consumption levels for the tasks during the planning horizon.

Type B loads must be executed within some given time windows and with fixed periodic electrical consumption levels. They are generated using electric appliances such as washing machines, dryers, and electric vehicles. For each corresponding task, the user indicates the time windows during which the task could be executed and also gives the periodic consumption levels in each time window. The model returns the best schedule to operate the tasks while respecting the community's operating constraints.

Type C loads have an uncontrollable starting time and periodic consumption levels. In other words, these loads must be executed without delay after the request and necessarily with the required electrical consumption levels. Examples of type C tasks include lighting, cooking, television, fridge, internet box, and appliances on standby. Therefore, each member has to estimate the periodic accumulation of the consumption of the corresponding tasks on the planning horizon.

Here are some examples of type B tasks in a planning horizon of 24h, sliced into periods of 1 hour. Three tasks are required: charging an electric vehicle (green), using a washing machine (gray), and a dryer (blue). We have two schedules for each task, as presented in Figure 2. The first plan is for the washing in three periods: in the first period, the washing machine will use 2kWh and then 1 kWh in the following periods. The second plan foresees to do the washing in two periods, with 2 kWh per period. The reasoning is the same for the two other tasks.

Tasks	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
Washing										2	1	1												
					2	2																		
Dryer																					2	2		
													2	2										
EV	2	2	2	2																				
											2	2	2	1	1									

Figure 2: Example of type B tasks.

According to this categorization, the problem is determining two types of decisions. First, we must decide when to start the type A tasks and which power levels to use periodically to reach the temperature objective. Second, we must choose the best schedules provided by the members to execute the type B tasks. These decisions must satisfy the individual and global constraints of the community. Recall that tasks of type C are uncontrollable, so no further decision is required for these.

2.2 Energy network model

Figure 1 presents the considered community members' asset ownership characteristics. The members can exchange green energy (Peer-to-Peer electricity transactions) at fixed rates known by all members. Indeed, members owning green energy generation tools can sell their surplus to the other community members or the green energy supplier. Furthermore, a green energy generator owning a storage unit can store the surplus and consume or sell it later. However, members holding only storage units cannot sell energy. Instead, they buy energy for their usage. Each individual is directly linked to the public grid and can subtract/inject energy from/into it when needed. Let C_{ih} be the amount of energy drawn from the network by member $i \in N$ at time period $h \in H$. The objective is to minimize the sum of the powers drawn from the primary grid by all the community members over the planning horizon H.

Section 3 presents the problem's formulation as mixed-integer linear program.

3 Loads scheduling model

This section introduces a mixed-integer linear program to get the exact solutions of the loads scheduling problem. The data used in the model are:

Sets

N	set of members in the community
B_i	set of batteries of member $i \in N$
J_{iA}	set of tasks of type A of member $i \in N$
J_{iB}	set of tasks of type B of member $i \in N$
H	set of the periods in the planning horizon
K_{ij}^{A}	set of rooms of member i where task $j \in J_{iA}$ can be performed
S_{ij}	set of schedules given by $i \in N$ for the execution of task $j \in J_{iB}$
$P_{ijk}^{\mathtt{A}}$	set of the power levels available to perform task $j \in J_{iA}$ in room $k \in K_{ij}^{A}$ of $i \in N$

Params

Δ_h	length of period h (*hour*)
π_i	subscribed power level of member $i \in N$
Γ_{ib}	capacity of battery $b \in B_i$ of member $i \in N$
ξ_{bi}	initial amount of electricity in battery $b \in B_i$ of member $i \in N$
η_{bi}	automatic discharge rate of battery $b \in B_i$ of member $i \in N$
ϕ_{bi}	maximum number of cycles for the battery $b \in B_i$ of member $i \in N$
β	maximum spending degradation threshold allowed (%)
Ω_i	equal to 1 if member i is allowed to exchange electricity, 0 otherwise
d_{bih}, c_{bih}	discharge and charging efficiencies of battery $b \in B_i$ of member $i \in N$ at period h (%)
ν_{ijk}	equal 1 if task $j \in J_{iA}$ is executed by member i in room $k \in K_{ij}^{A}$ 0, otherwise
$\theta_j(p)$	function modeling the temperature variation when performing task $j \in J_{iA}$ according to power p
$[t_{ijk}^{\mathtt{low}}, t_{ijk}^{\mathtt{up}}]$	comfort temperature targeted by member i when performing task $j \in J_{iA}$ in room k (°C)
$[h_{ijk}^{\mathtt{low}}, h_{ijk}^{\mathtt{up}}]$	time where member$i$'s confort zone must must be reached when performing task $j \in J_{iA}$
$\bar{T}_{ijk}^{\mathtt{room}}$	initial temperature of room $k \in K_{ij}^{A}$ where member i wants to perform task $j \in J_{iA}$
$T_h^{\mathtt{out}}$	outside temperature at time period $h \in H$ (°C)
$T_h^{\mathtt{room}}$	room temperature at time period $h \in H$ (°C)
$P_{ih}^{\mathtt{Gen}}$	energy production of member i at period h (kW)
$P_{ijhs}^{\mathtt{B}}$	power consumption of task $j \in J_{iB}$ of member i at period h in the schedule $s \in S_{ij}$ (kW)
$P_{bih}^{\mathtt{in}}, P_{bih}^{\mathtt{out}}$	charging and discharging power of battery $b \in B_i$ of member $i \in N$ at time period h (kW)
$p_{ih}^{\mathtt{C}}$	cumulative power consumption of type C tasks at time period $h \in H$
G_i	gain of member $i \in N$ when operating outside a community(€)
$v_h^{\mathtt{MG}}$	unit purchase price of electricity from the primary grid during period $h \in H$ (€/kWh)
$\tilde{v}_h^{\mathtt{MG}}$	unit sale price of electricity to the primary grid during period $h \in H$ (€/kWh)
$v_h^{\mathtt{Com}}$	unit purchase price of electricity in the community during period $h \in H$ (€/kWh)
$\tilde{v}_h^{\mathtt{Com}}$	unit sale price of electricity in the community during period $h \in H$ (€/kWh)
$v_h^{\mathtt{GES}}$	hnit purchase price of electricity to *the green energy supplier* at period $h \in H$ (€/kWh)
$\tilde{v}_h^{\mathtt{GES}}$	unit sale price of electricity to *the green energy supplier* at time period $h \in H$ (€/kWh)

Note that the prices $\tilde{v}_h^{\mathtt{GES}}$ and $v_h^{\mathtt{GES}}$ apply when individuals are in the community. In addition, $\tilde{v}_h^{\mathtt{MG}}$ and $v_h^{\mathtt{MG}}$ are the selling and buying prices when the members are outside the community. Prices with index C are intra-member transaction prices.

Type A tasks are heating tasks, performed to regulate the temperature. Without loss of generality, we consider two types of heating tasks: houserooms and water heaters. We model the variation of temperature for a task according to the power level as follows. Let p_h be the power level at which the device operates at time period h.

- For a house room, based on Newton's law of cooling [2], the temperature variation function θ_1 can be written as
$$\theta_1(p_h) = \theta_1(p_{h-1}) + \frac{\Delta}{C_r}(p_h - U(\theta_1(p_{h-1}) - T_h^{\texttt{out}})),$$
where C_r is heat capacity (J/K), parameter U designates the heat loss coefficient of a room in (W/K), and Δ is the heating time in second. $T_h^{\texttt{out}}$ is the external temperature at period $h \in H$.
- For a water heater, using the method from [19], the temperature variation function θ_2 is given by
$$\theta_2(p_h) = \frac{\Delta_h}{Mc_p}\left(-\frac{S}{R}\left[\theta_2(p_{h-1}) - T_h^{\texttt{room}}\right] - 1000 M_w c_p[\theta_2(p_{h-1}) - T_{in}] + \upsilon p_h\right) + \theta_2(p_{h-1}),$$
where c_p is the isobaric specific heat capacity of water (kcal/kg.°C), υ is the efficiency of the electricity-to-heat transformation, M is the weight of the water (kg). The data S designates the exchange surface of the water container with the external area, R is the thermal resistance of the tank insulation in m^2.°C/W and M_w is the average hot water demand rate during the time interval, which we assume to be equal to zero since we don't have that information. Parameter T_{in} is the supply domestic cold water temperature, and recall that $T^{\texttt{room}}$ is the temperature of the ambient environment. Finally, Δ_h heating time in seconds.

Given these data and heating model, our problem can be formulated with the following variables:

$x^A_{ijkhp} \in \{0,1\}$	is equal to 1 if and only if task $j \in J_{iA}$ is in progress in period $h \in H$ in room $k \in K^A_{ij}$ for member $i \in N$,
$x^s_{ij} \in \{0,1\}$	is equal to 1 if and only if schedule $s \in S_{ij}$ is chosen for the execution of task $j \in J_{iB}$ of community's member i,
$z_{bih} \in \{0,1\}$	is equal to 1 if and only if battery $b \in B_i$ of member $i \in N$ is charging in time period $h \in H$,
$w_{bih} \in \{0,1\}$	is equal to 1 if and only if the operation of battery $b \in B$ of $i \in N$ changes from discharging or inactive to charging in period $h \in H$,
$T_{ijkh} \geq 0$	represents the temperature in room k of member $i \in N$ reached by performing task $j \in J_{iA}$ in period $h \in H$,
$q_{bih} \in \mathbb{R}$	is the amount of energy injected into/out of battery $b \in B_i$ of $i \in N$ in time period $h \in H$, with $q_{bih} \leq 0$ if b discharges, $q_{bih} \geq 0$ if b charges, and $q_{bih} = 0$ if b is inactive,
$E_{bih} \geq 0$	is the amount of electricity available in battery $b \in B_i$ of member $i \in N$ at the end of period h,
$f_{ieh} \geq 0$	is the amount of energy produced by member i and sent to member e in period $h \in H$,
$I_{ih} \geq 0$	is the amount of photovoltaic energy injected in the public grid by member $i \in N$ in period $h \in H$,
$C_{ih} \geq 0$	is the amount of energy withdrawn from the grid by member $i \in N$ in period $h \in H$,
$\tilde{G}_i \in \mathbb{R}$	is the gain of member $i \in N$ when operating in the community.

The load scheduling problem is formulated as :

$$\min \quad \sum_{i \in N} \sum_{h \in H} C_{ih} \tag{1a}$$

$$\text{s.t.} \quad \sum_{j \in J_{iA}} \sum_{k \in K^A_{ij}} \sum_{p \in P^{\texttt{A}}_{ijk}} p \cdot x^A_{ijkhp} + \sum_{j \in J_{iB}} \sum_{s \in S_{ij}} P^{\texttt{B}}_{ijhs} x^s_{ij} + p^{\texttt{C}}_{ih} + \sum_{b \in B_i} \frac{q_{bih}}{\Delta_h}$$

$$= P_{ih}^{\texttt{Gen}} + \sum_{\substack{i' \in N \\ i' \neq i}} (f_{i'ih} - f_{ii'h}) + C_{ih} - I_{ih} \qquad i \in N, h \in H \qquad (1b)$$

$$\sum_{p \in P_{ijk}^{\texttt{A}}} x_{ijkhp}^{A} \leq 1 \qquad i \in N, j \in J_{iA}, k \in K_{ij}^{A}, h \in H \qquad (1c)$$

$$\sum_{s \in S_{ij}} x_{ij}^{s} = 1 \qquad j \in J_{iB}, i \in N \qquad (1d)$$

$$\sum_{i' \in N, i' \neq i} f_{ii'h} + I_{ih} \leq P_{ih}^{\texttt{Gen}} + \sum_{b \in B_i} \frac{E_{bih}}{\Delta_h} \Omega_i \qquad i \in N, h \in H \qquad (1e)$$

$$\tilde{G}_i = \sum_{h \in H} \left[\sum_{\substack{i' \in N \\ i' \neq i}} (v_h^{\texttt{Com}} f_{ii'h} - \tilde{v}_h^{\texttt{Com}} f_{i'ih}) + v_h^{\texttt{GES}} I_{ih} - \tilde{v}_h^{\texttt{GES}} C_{ih} \right] \Delta_h \qquad i \in N \qquad (1f)$$

$$\frac{\tilde{G}_i - G_i}{|G_i|} \geq \beta \qquad i \in N \qquad (1g)$$

$$\sum_{i' \in N, i' \neq i} f_{i'ih} + C_{ih} \leq \pi_i \qquad i \in N, h \in H \qquad (1h)$$

$$\sum_{i' \in N, i' \neq i} f_{ii'h} + I_{ih} \leq \pi_i \qquad i \in N, h \in H \qquad (1i)$$

$$T_{ijkh} = \theta_j \left(\sum_{p \in P_{ijk}^{\texttt{A}}} p \cdot x_{ijkhp}^{A} \right) \qquad i \in N, j \in J_{iA}, k \in K_{ij}^{A}, h \in H \qquad (1j)$$

$$t_{ijk}^{\texttt{low}} \leq T_{ijkh} \leq t_{ijk}^{\texttt{up}} \qquad i \in N, j \in J_{iA}, k \in K_{ij}^{A}, h \in [h_{ijk}^{\texttt{low}}, h_{ijk}^{\texttt{up}}] : \nu_{ijk} = 1 \qquad (1k)$$

$$T_{ijk0} = \bar{T}_{ijk}^{\texttt{room}} \qquad i \in N, j \in J_{iA}, k \in K_{ij}^{A} \qquad (1l)$$

$$\Delta_h d_{bih} P_{bih}^{\texttt{out}} (z_{bih} - 1) \leq q_{bih} \leq \Delta_h c_{bih} P_{bih}^{\texttt{in}} z_{bih} \qquad i \in N, b \in B_i, h \in H \qquad (1m)$$

$$z_{bih} - z_{bi(h-1)} \leq w_{bi(h-1)} \qquad i \in N, b \in B_i, h \in 2, \ldots, |H| \qquad (1n)$$

$$\frac{E_{bih} - q_{bi(h-1)}}{E_{bi(h-1)}} = \eta_{bi} \qquad i \in N, b \in B_i, h \in 1, \ldots, |H| \qquad (1o)$$

$$\sum_{h \in H} w_{bih} \leq \phi_{bi} \qquad i \in N, b \in B_i \qquad (1p)$$

$$E_{bih} \leq \Gamma_{bi} \qquad i \in N, b \in B_i, h \in H \qquad (1q)$$

$$E_{bi0} = \xi_{bi} \qquad i \in N, b \in B_i \qquad (1r)$$

$$E_{bih} = \xi_{bi} \qquad i \in N, b \in B_i \qquad (1s)$$

$$x_{ijkhp}^{A} \in \{0, 1\} \qquad i \in N, j \in J_{iA}, h \in H, p \in P_{ijk}^{\texttt{A}}, k \in K_{ij}^{A} \qquad (1t)$$

$$x_{ij}^{s} \in \{0, 1\} \qquad i \in N, j \in J_{iB}, s \in S_{ij} \qquad (1u)$$

$$z_{bih}, w_{bih} \in \{0, 1\} \qquad i \in N, b \in B_i, h \in H \qquad (1v)$$

$$E_{bih} \geq 0 \qquad i \in N, b \in B, h \in H \qquad (1w)$$

$$C_{ih} \geq 0 \qquad i \in N, h \in H \qquad (1x)$$

$$I_{ih} \geq 0 \qquad i \in N, h \in H \qquad (1y)$$

$$f_{ii'h} \geq 0 \qquad i, i' \in N, h \in H. \qquad (1z)$$

The MILP's is to minimize the cumulative energy collected by community members over the planning horizon, expressed in the objective function (1a). Constraints (1b) enforce that for each member $i \in N$, at any time period h, the sum of the energies received, and the energy produced is equal to the energy consumed plus the energy injected into the grids (public and community). On the left-hand side of the equality, the two sums represent respectively the consumption of type A and B tasks, $p_{ih}^{\texttt{C}}$ is the total consumption of type C tasks in period h. The third sum represents the extractions into/out of batteries.

The right-hand side represents the energy exchanges between the community members. Constraints (1c) impose that at most one power level is chosen in period $h \in H$ to regulate the temperature of room $k \in K_{ij}^A$ of member $i \in N$. Constraints (1d) guarantee that only one schedule is selected to perform task $j \in J_{iB}$ of member i during planning horizon H. Community members can only feed photovoltaic energy into the public grid. Therefore, a member who does not produce green energy should only receive the energy needed for instant consumption or for charging the batteries. Hence, Constraints (1e) enforce that an individual who does not produce energy does not send energy. In addition, membership in the energy community should not degrade an individual's situation beyond a certain threshold. Constraints (1g) enforce that a member's gain can deteriorate by at most $\beta\%$. Constraints (1h) and (1i) are related to the subscribed power levels of members. Indeed, a member can only inject or withdraw a predefined amount of energy per period.

The next constraints are related to type A tasks for temperature regulation. Constraints (1j) define the temperature for each room and each period. Constraints (1k) enforce that the temperature of a room must be in a specific interval during a given time interval. Finally, Constraints (1l) set the initial temperature of the rooms.

We then turn to battery usage. Constraints (1m) ensures that q_{bih} lies between the minimum and maximum power levels allowed for discharging/charging battery b of member i at period $h \in H$. The number of cycles allowed is limited to limit the degradation of a battery. Constraints (1n) set variables w_{bih} to track the changes in z_{bih}. Constraints (1p) enforce the desired maximum number of cycles. Constraints (1q) enforce capacity constraints on batteries, and Constraints (1r) and (1s) imply that the initial and final states of batteries are the same.

Finally, Constraints (1t) to (1z) define the domain of the variables.

Remark. *Note that nothing prevents an individual i from selling to the community member i' energy drawn from the main grid. However, if such a solution is returned by the model, one can readily obtain a solution that is not less efficient by letting i' draw that energy directly in the main grid instead of i.*

4 Column generation-based heuristic

The proposed MILP can hardly be solved optimally for large problem instances, often not even finding feasible solutions. To overcome this, we propose a column generation-based heuristic, in which the columns are the schedules for executing type A tasks. That heuristic allows us to generate only the schedules with minimum costs (through a pricing problem) that are likely to improve the restricted master problem (RMP) instead of explicitly generating them. The approach we propose is a heuristic because new columns are generated only at the root node of the branch-and-bound tree.

4.1 Dantzig-Wolfe reformulation

The column generation-based algorithm relies on a Dantzig-Wolfe reformulation [8] where the scheduling constraints related to the type A loads are put into a pricing problem, in this case, the temperature constraints (1j), (1k) and (1l). All other constraints of the load scheduling MILP are placed into the restricted master problem. It includes the same variables of the loads scheduling MILP except for those associated with the type A tasks. Indeed, instead of choosing a power level at each period, we will have to determine which schedule to select among those returned by the pricing problem over the iterations.

Let X_{ijk} denotes the set of feasible schedules for the requested task $j \in J_{iA}$ in room $k \in K_{ij}^A$ of member $i \in N$ i.e. such that the temperature constraint is satisfied:

$$X_{ijk} = \left\{ x^A \in \{0,1\}^H : T_{ijk0} = \bar{T}_{ijk}^{\texttt{room}},\ T_{ijkh} = \theta_j \left(\sum_{p \in P_{ijk}^{\texttt{A}}} p \cdot x_{ijkhp}^A \right) \text{ and } t_{ijk}^{\texttt{low}} \leq T_{ijkh} \leq t_{ijk}^{\texttt{up}}, \forall h \in [h_{ijk}^{\texttt{low}}, h_{ijk}^{\texttt{up}}] \right\}$$

$\forall i \in N, j \in J_{iA}, k \in K_{ij}^A$. Let us introduce a binary variable $\sigma_{ijk}^{\chi^A}$ that is equal to 1 if schedule $\chi^A \in X_{ijk}$ has been selected to perform task j in the k^{th}-room of member i. Performing a Dantzig-Wolfe reformulation of the constraints corresponding to X_{ijk}, namely (1j)-(1l), we obtain the following reformulation:

$$\min \sum_{i\in N}\sum_{h\in H} C_{ih} \tag{2a}$$

$$\text{s.t.} \sum_{\substack{j\in J_{iA}\\ k\in K_{ij}^A}} \sum_{p\in P_{ijk}^{\mathtt{A}}} p\cdot\left(\sum_{\chi^A\in X_{ijk}} \chi^A_{ijkhp}\sigma^{\chi^A}_{ijk}\right) + \sum_{\substack{j\in J_{iB}\\ s\in S_{ij}}} P^{\mathtt{B}}_{ijhs}x^s_{ij} + p^{\mathtt{C}}_{ih} + \sum_{b\in B_i}\frac{q_{bih}}{\Delta_h} = P^{\mathtt{Gen}}_{ih} + \sum_{\substack{i'\in N\\ i'\neq i}}(f_{i'ih} - f_{ii'h})$$

$$+C_{ih} - I_{ih} \qquad \forall i\in N, h\in H \tag{2b}$$

$$\sum_{\chi^A\in X_{ijk}} \sigma^{\chi^A}_{ijk} = 1 \qquad \forall i\in N, j\in J_{iA}, k\in K^A_{ij} : \nu_{ijk} = 1 \tag{2c}$$

(1d) to (1i), (1m) to (1z),

where we denote respectively the dual values of constraints (2b) and (2c) by $\{\alpha_{ih}\}_{i\in N,h\in H}$ and $\{\tau_{ijk}\}_{i\in N,j\in J_{iA},k\in K^A_{ij}}$. The Restricted Master Problem (RMP) is obtained from the above reformulation by considering only subsets of elements in X_{ijk}, which we denote $\tilde{X}_{ijk}$.

The pricing problem aims at determining the feasible schedules for the required type A tasks in the planning horizon. In other words, it aims at determining schedules that satisfy the temperature constraints of these tasks. Mathematically, the pricing problem searches for the cheapest vector in X_{ijk} based on the reduced costs obtain from RMP.

$$\min\left\{\sum_{h\in H} -\alpha_{ih}p\cdot\chi^A_{hp} - \tau_{ijk} \text{ s.t. } \chi^A\in X_{ijk}\right\} \quad \forall i\in N, j\in J_{iA}, k\in K^A_{ij}. \tag{3}$$

To optimize over X_{ijk}, it is convenient to re-introduce variables x^A and T from the original formulation, leading to the following pricing problem for each $i\in N, j\in J_{iA}, k\in K^A_{ij}$:

$$\min \sum_{h\in H}\sum_{p\in P^{\mathtt{A}}_{ijk}} -\alpha_{ih}p\cdot x^A_{ijkhp} - \tau_{ijk} \tag{4a}$$

$$\text{s.t. } T_{ijkh} = \theta_j\left(\sum_{p\in P^{\mathtt{A}}_{ijk}} p\cdot x^A_{ijkhp}\right) \qquad \forall h\in H \tag{4b}$$

$$t^{\mathtt{low}}_{ijk} \le T_{ijkh} \le t^{\mathtt{up}}_{ijk} \qquad \forall h\in[h^{\mathtt{low}}_{ijk}, h^{\mathtt{up}}_{ijk}] : \nu_{ijk} = 1 \tag{4c}$$

$$T_{ijk0} = \bar{T}^{\mathtt{room}}_{ijk} \tag{4d}$$

$$\sum_{p\in P^{\mathtt{A}}_{ijk}} x^A_{ijkhp} \le 1 \qquad \forall h\in H \tag{4e}$$

$$x^A_{ijkhp}\in\{0,1\} \qquad \forall h\in H, p\in P^{\mathtt{A}}_{ijk}. \tag{4f}$$

It can possibly improve the restricted master problem if $\tilde{z}_{ijk} < 0$. In this case, the corresponding column is added to the restricted master problem.

4.2 The algorithm

This section describes the column generation-based heuristic detailed in Algorithm 1, which starts by determining an initial solution. We determine initial feasible schedules by solving (1) with $f_{ii'h}$ for each i, $i'\in N$ and $h\in H$, that correspond to the case where there are no internal links between members. We add these schedules to $\tilde{X}$ and solve RMP to recover the dual values α and τ. That concludes the initial phase.

The following instructions are repeated until a stopping criterion is satisfied: solve the pricing problem, which returns the feasible schedules of all required tasks. For each feasible schedule, if the corresponding reduced cost is negative, add that planning to the relaxed RMP and solve it to get the dual values.

We consider two stopping criteria. First, if the pricing problem returns only schedules with positive or null reduced costs. Or if a fixed maximum number `maxIter` of iteration is reached. When a stopping criterion is satisfied, the last step consists of solving the RMP with the integrality constraints so as to get the final solution of the heuristic.

Due to the number of members and type A tasks, instead of solving the pricing problem for each task $j \in J_{iA}$, in each room $k \in K_{ij}^A$ of each member $i \in N$, we may solve a unique pricing problem. We will see later which is the most efficient way to proceed. After solving the pricing problem at once, we get a feasible schedule for each required task. Only those with negative reduced cost will be kept. The pricing problem is then:

$$\min \sum_{\substack{i\in N\\ j\in J_{iA}}} \sum_{\substack{k\in K_{ij}^A\\ h\in H}} \sum_{p\in P_{ijk}^{\mathtt{A}}} -\alpha_{ih} p \cdot x_{ijkhp}^A - \tau_{ijk} \tag{5a}$$

$$\text{s.t. } T_{ijkh} = \theta_j \left(\sum_{p\in P_{ijk}^{\mathtt{A}}} p \cdot x_{ijkhp}^A \right) \qquad \forall i \in N, j \in J_{iA}, k \in K_{ij}^A, h \in H \tag{5b}$$

$$T_{ijkh} \in [t_{ijk}^{\mathtt{low}}, t_{ijk}^{\mathtt{up}}] \qquad \forall i \in N, j \in J_{iA}, k \in K_{ij}^A, h \in [h^{\mathtt{low}}, h^{\mathtt{up}}]_{ijk} : \nu_{ijk} = 1 \tag{5c}$$

$$T_{ijk0} = \bar{T}_{ijk}^{\mathtt{room}} \qquad \forall i \in N, j \in J_{iA}, k \in K_{ij}^A \tag{5d}$$

$$\sum_{p\in P_{ijk}^{\mathtt{A}}} x_{ijkhp}^A \leq 1 \qquad \forall i \in N, j \in J_{iA}, k \in K_{ij}^A, h \in H \tag{5e}$$

$$x_{ijkhp}^A \in \{0,1\} \qquad \forall i \in N, j \in J_{iA}, k \in K_{ij}^A, h \in H, p \in P_{ijk}^{\mathtt{A}}. \tag{5f}$$

The algorithm is schematized in Algorithm 1, in which c, and PP denote respectively the matrix of reduced costs and the pricing problem.

Algorithm 1: Column generation-based heuristic's algorithm.

input : `maxIter`
output : best solution found

Function `Main( maxIter)`:
- solve F^{MILP} without internal links to get X^{init}, a tuple of feasible schedules, add X^{init} to $\tilde{X}$; $iter = 0$;
- **do**
  - $iter++$;
  - solve (RMP) to get the dual values α and τ;
  - solve (PP) with these dual values to get tuple X^{iter} and matrix c;
  - **for** $i \in N$, $j \in J_{iA}$, $k \in K_{ij}^A$ **do**
    - **if** $\nu_{ijk} > 0$ **then**
      - **if** $c_{ijk} < 0$ **then**
        - add X_{ijk}^{iter} to $\tilde{X}_{ijk}$;
      - **end**
    - **end**
  - **end**
- **while** $iter \leq$ `maxIter` **or** $(c \geq 0) == 0$;
- solve (RMP) with integrality constraints to get *Solution*;

return *Solution*.

4.3 Complexity of the pricing problem

Let us now study the complexity of PP.

Proposition 1. *PP is polynomially solvable if* $|H| = 1$.

Proof. Suppose that $m \geq 1$ tasks are required by a single member to heat houserooms, $j = 1$. Thus, to alleviate notations, we drop indices i, j and h in the rest of the proof. The problem comes to choose one power in the set $P_k^{\mathtt{A}}$ of available powers for each houseroom k. By assuming that the required temperature variation is feasible, an answer of the decision problem is obtained by performing at most $\sum_{k=1}^{k=m} 3|P_k^{\mathtt{A}}|$ operations. □

We consider next the special case with a unique member and a unique type of tasks to be executed, so we drop indexes i and j. A unique task is considered then, we also drop index k. We further assume that $t^{\mathtt{low}} = t^{\mathtt{up}} = \tilde{\theta}$, $h^{\mathtt{low}} = h^{\mathtt{up}} = |H|$ and $\bar{T}^{\mathtt{room}} = 0$. We also consider that $(\Delta/Cr) = 1$ for the room, so the temperature variation function θ turns to:

$$T_h = T_{(h-1)} + p_h - T_h^{loss},$$

where p_h is the power used to perform the task in the room at period h and T_h^{loss} is the heat lost by the room at period h. Then,

$$T_{|H|} = \sum_{h \in H} (p_h - T_h^{loss}),$$

which must be equal to $\tilde{\theta}$. With these simplifications, PP becomes:

$$\tilde{z} = \min \sum_{h \in H} \sum_{p \in P^{\mathtt{A}}} -\alpha_h p \cdot x_{hp}^A - \tau \tag{6a}$$

$$\text{s.t.} \sum_{h \in H} \Big(\sum_{p \in P^{\mathtt{A}}} p \cdot x_{hp}^A - T_h^{loss} \Big) \geq \tilde{\theta} \tag{6b}$$

$$\sum_{p \in P^{\mathtt{A}}} x_{hp}^A \leq 1 \qquad \forall h \in H \tag{6c}$$

$$x_{hp}^A \in \{0,1\} \qquad \forall h \in H, p \in P^{\mathtt{A}}. \tag{6d}$$

Proposition 2. *If* $|H| > 1$*, PP is NP-hard even if a single task has to be planned.*

Proof. We will show that the well-known Multiple-Choice Subset Sum Problem (MCSSP) (known to be NP-hard [12]) is polynomially reducible to the pricing problem. Recall that, given a set of m classes $N_1, \ldots, N_m$, each class containing weights $w_{i1}, \ldots, w_{in_i}$, the MCSSP aims at selecting at most one weight from each class such that the total weight sum is maximized without exceeding the capacity c. Introducing binary variables x_{ij} denoting which weights are taken, MCSSP can be cast as

$$W = \max \sum_{i=1}^{m} \sum_{j \in N_i} w_{ij} x_{ij} \tag{7a}$$

$$\text{s.t.} \sum_{i=1}^{m} \sum_{j \in N_i} w_{ij} x_{ij} \leq c \tag{7b}$$

$$\sum_{j \in N_i} x_{ij} \leq 1 \qquad \forall i = 1, \ldots, m \tag{7c}$$

$$x_{ij} \in \{0,1\} \qquad \forall i = 1, \ldots, m, j \in N_i. \tag{7d}$$

The maximization form of MCSSP may be transformed into minimization form by finding for each class N_i: $\bar{w}_i = \min_{j \in N_i} w_{ij}$, by setting $\tilde{w}_{ij} = w_{ij} - \bar{w}_i \ \forall j \in N_i$, and $\tilde{c} = c - \sum_{i=1}^{m} \bar{w}_i$. The minimization problem is defined in $\tilde{w}$ and $\tilde{c}$.

Let us now consider an instance of PP with $\tilde{c} = \tilde{\theta}$, $m = |H|$, $N_i = P^{\mathtt{A}}$ $\forall i = 1, \ldots, m$, $\tau = 0$, $\alpha_h = -1$ $\forall h \in H$, and $T_h^{loss} = 0$ $\forall h \in H$. Finally, $\tilde{w}_{ij} = \tilde{w}_{hp}$ where $\tilde{w}_{hp}$ denotes the power selected in N_h. PP formulated in (6) becomes:

$$\tilde{z} = \min \sum_{h=1}^{m} \sum_{p \in N_h} \tilde{w}_{hp} x_{hp}^A \tag{8a}$$

$$\text{s.t.} \sum_{h=1}^{m} \sum_{p \in N_h} \tilde{w}_{hp} x_{hp}^A \geq \tilde{c} \tag{8b}$$

$$\sum_{p \in N_h} x_{hp}^A \leq 1 \qquad \forall t = 1, \ldots, m \tag{8c}$$

$$x_{hp}^A \in \{0, 1\} \qquad \forall t = 1, \ldots, m, p \in N_h. \tag{8d}$$

We see that the above instance of PP corresponds to the considered instance of MCSSP, written in the minimization form, proving the reduction and thus, the hardness of PP. □

5 Extension to multiple days

This section is devoted to solving the problem of planning loads over several successive days.

Note that this study is only relevant if accurate forecasts of energy production and consumption are available. Indeed, the temperature variation functions are continuous, and the temperature preferences of individuals are predictable. Furthermore, predictions can be made for the other two load categories. Thus, let D be the set of successive days over which we wish to solve the planning problem. Solving the multiple-day load scheduling problem can be approached in two ways. The first approach consists in adding an index d indexing day $d \in D$ to the decision variables of the previously described MILP, while ensuring the continuity of the room temperature and the state of charge of the batteries. The result is a unique MILP to be solved over the horizon of several days.

The second method is a heuristic approach, where we solve the initial MILP by modifying the objective function for each day of time horizon $|D|$. Indeed, in the model described in Section 3, there is a trade-off between injecting into the public grid or charging the batteries each time there is a surplus of global energy production of the community. However, since we do not have a view on the events of days $d + 1, \ldots, |D|$, it would be better to prioritize battery charging in such a case. We do this prioritization using $\sum_{i \in n} \sum_{h \in H} (C_{ih} + I_{ih})$ as the objective function for each day $d \in D$.

Algorithm 2 describes how that heuristic works. Each day d one solves the MILP when there is no exchange between the members to acquire data G_i $\forall i \in N$ because we must ensure that constraint (1g) is satisfied each day. Then, the initial model is solved for day d, with the initial temperature and state of charge of the batteries equal to their final state on day $d - 1$. Furthermore, for the water heaters, since the water of day $d - 1$ is consumed, and we do not have a hot water consumption function, we assume that the water temperature is reset at period, $h_\rho = 10$ of each day.

In that algorithm, G, E, T_H represent, respectively, the vector of earnings, the vector of final state of charge of the batteries and the final temperature vectors of day d when there is no exchange between the members. The model when there is no exchange between the members corresponds to the case where $f_{ieh} = 0$ $\forall i, e \in N, h \in H$. Furthermore, $\tilde{E}$ and $\tilde{T}_H$ represent respectively the state of charge vector of the batteries and the final temperature vector of day d when the individuals exchange energy.

Algorithm 2: Heuristic for solving the scheduling problem on several days.

for *d in D* **do**
- include data for day d;
- solve the model when there are no exchanges between members to get G, E, T_H;
- solve the model when the members exchange their energy surplus to get vectors $\tilde{E}$, $\tilde{T}_{Hd}$;
- update the data for next day by adding G, E, T_H, $\tilde{E}$ and $\tilde{T}_{Hd}$;
- return the best solution found for day d

end

In the following section we report the results we obtain with the two approaches to solving the problem written in this section as well as a comparison between these approaches.

6 Experimental results

We present in this section the experimental results of the solution approaches presented previously. Namely, we compare:

- F^{MILP}: The monolithic formulation in Section 3
- F^{MILP}_{index}: Algorithm 2 for several consecutive days
- CG^{PPo}: Algorithm 1 using `PPo`
- CG^{PPs}: Algorithm 1 using `PPs`

To assess the impact of joining a collective self-consumption community, we compare two scenarios: community with and without internal links between members.

- The community scenario without internal links corresponds to the case when no link exists between the members except the main grid. In this case, the photovoltaic energy producers inject their surplus directly into the main grid.
- The community scenario with internal links corresponds to the case where the individuals operate in a community. They can exchange their surplus with other members of the community. They then have to make arbitrage between injecting into the community or injecting into the grid.

This section is divided into two subsections. In the first subsection, we present the instance used to assess the solving approaches. The instance is based on realistic data from Smart Lou Quila (figure 3 present the members' locations in the Cailar). The second subsection presents the numerical results obtained on these instances. Notice that the latter subsection also contains scalability experiments that assess the MILP and the column generation heuristics on larger instances obtained by multiplying the available data several times.

Figure 3: Community's members' locations.

6.1 Instance

6.1.1 Storage and generation assets

The instance is generated over one day by steps of 30 minutes. The community is composed of seven members, each with equipment whose characteristics are presented in Table 1. Members 1 and 2 possess both energy production assets and storage units. The third member has only a battery. The three

members, 4, 5, and 6, have only the energy generation asset. The last member does not have any equipment. Finally, the subscribed power per member in kVa are respectively; 6, 36, 6, 9, 9, 6, and 9. In the following, to ensure model consistency, we fix the parameter $\Omega_i \in N$ such that $\Omega_i = 1$ if and only if the member i has a power generation system.

Member 1		Member 2	
Photovoltaic (PV)	Yes	PV	Yes
PV Capacity	3.2kWp	PV Capacity	6.12kWp
Battery	Yes	Battery	Yes
Number	1 unit	Number	1 unit
Capacity	9.8kWh	Capacity	9.8kWh
Initial state of charge	4.5kWh	Initial state of charge	4.5kWh
Efficiency	97.5%	Efficiency	97.5%
Power	3.7kW	Power	5kW
Periodic discharge rate	1%	Periodic discharge rate	1%
Member 3		Member 4	
PV	No	PV	Yes
PV Capacity	0kWp	PV Capacity	3.2kWp
Battery	yes	Battery	No
Number	1 unit	Number	0 unit
Capacity	9.8kWh	Capacity	0kWh
Initial state of charge	4.5kWh	Initial state of charge	0kWh
Efficiency	97.5%	Efficiency	0%
Power	3.7kW	Power	0kW
Periodic discharge rate	1%	Periodic discharge rate	0%
Member 5		Member 6	
PV	Yes	PV	Yes
PV Capacity	3.2kWp	PV Capacity	3.2kWp
Battery	No	Battery	No
Member 7			
PV		No	
Battery		No	

Table 1: Production and storage assets description in the community.

Figure 4 presents the periodic real productions of the members of Smart Lou Quila [3], composed of seven members: 6 residences and a municipal stadium located in the south of France. It also presents the community's total energy production. The data are taken from date January the 08 and 09, 2022.

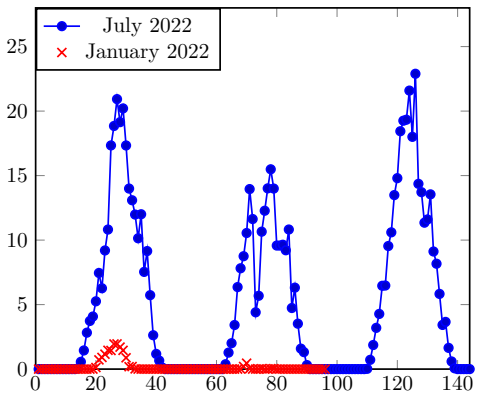


Figure 4: Community's total energy production in kW on two time horizons.

6.1.2 The loads

We present in this section realistic data for the loads, built up together with our partner Smart Lou Quila and inquiries realized among the members of the demonstrator. We present next the requested tasks by the members during the planning horizon according to the three described loads categories. The planning horizon is consists of a day sliced into 48 equal length periods. As mentioned previously, we consider two type A tasks: room and water heating. Each member has at most three rooms. The following array ν indicates the requested type A tasks by the members. Member 1 (in bold) wants to regulate the temperature of three rooms and one water heater. Member 2 did not request any type A task during the horizon. Member 3 has three tasks, two-room heating and one water heating and so on for the next members. Thus, ν_{ijk} is equal to 1 if task $j \in J_{iA}$ is executed by member i in room k, 0 otherwise. In addition, $j = 1$ means that the corresponding task is room heating, and $j = 2$ means that the requested task is water heating.

$$\begin{array}{l}\nu_{ij1}=[\mathbf{1\ 1}\ ;\ 0\ 0\ ;\ 1\ 1\ ;\ 1\ 1\ ;\ 1\ 1\ ;\ 0\ 1\ ;\ 1\ 1\ ]\\ \nu_{ij2}=[\mathbf{1\ 0}\ ;\ 0\ 0\ ;\ 0\ 0\ ;\ 0\ 0\ ;\ 1\ 0\ ;\ 1\ 0\ ;\ 1\ 0\ ]\\ \nu_{ij3}=[\mathbf{1\ 0}\ ;\ 0\ 0\ ;\ 1\ 0\ ;\ 0\ 0\ ;\ 0\ 0\ ;\ 1\ 0\ ;\ 1\ 0\ ]\end{array}$$

$\forall i \in N, j \in J_{iA}$. The following matrices $t^{\texttt{low}}$ and $t^{\texttt{up}}$ represent the desired temperature of the members in their corresponding rooms during intervals $[h^{\texttt{low}}_{ki}, hu_{ki}]$ knowing that the initial temperature in the rooms is depicted in $\bar{T}^{\texttt{room}}$.

$$t^{\texttt{low}} = \begin{pmatrix} 19 & 0 & 16 & 17 & 20 & 22 & 19 \\ 22 & 0 & 19 & 22 & 23 & 22 & 19 \\ 16 & 0 & 19 & 17 & 20 & 22 & 19 \end{pmatrix} \quad t^{\texttt{up}} = \begin{pmatrix} 22 & 24 & 19 & 20 & 21 & 24 & 21 \\ 24 & 24 & 24 & 23 & 24 & 24 & 21 \\ 20 & 21 & 24 & 20 & 22 & 24 & 21 \end{pmatrix}$$

$$h^{\texttt{low}} = \begin{pmatrix} 20 & 1 & 20 & 20 & 34 & 18 & 20 \\ 20 & 1 & 20 & 20 & 34 & 18 & 20 \\ 20 & 1 & 20 & 20 & 34 & 18 & 20 \end{pmatrix} \quad h^{\texttt{up}} = \begin{pmatrix} 30 & 48 & 40 & 40 & 47 & 34 & 42 \\ 30 & 48 & 40 & 40 & 47 & 34 & 42 \\ 30 & 48 & 40 & 40 & 47 & 34 & 42 \end{pmatrix}$$

$$\bar{T}^{\texttt{room}} = \begin{pmatrix} 12 & 14 & 16 & 12 & 14 & 12 & 15 \\ 15 & 14 & 12 & 11 & 14 & 12 & 8 \\ 12 & 14 & 10 & 12 & 14 & 12 & 10 \end{pmatrix}$$

We now present the physical characteristics of the rooms in Table 2, while Table 3 presents these for the water heaters. In Table 2, column Surface designates the room's surface of rooms, C_r the heat capacity, and U the heat loss coefficient. We calculate C_r and U according to the information given by the members.

Member	room	Surface	C_r	U	Member	room	Surface	C_r	U
Member 1	room 1	$9m^2$	297	12	Member 3	room 1	$18m^2$	594	24
	room 2	$15m^2$	495	20		room 2	$9m^2$	297	12
	room 3	$18m^2$	594	24		room 3	$9m^2$	297	12
Member 4	room 1	$12m^2$	396	16	Member 5	room 1	$25m^2$	825	33.3
	room 2	$20m^2$	660	26.6		room 2	$10m^2$	330	13.3
	room 3	$12m^2$	396	16		room 3	$12m^2$	396	16
Member 6	room 1	$18.5m^2$	610.5	24.6	Member 7	room 1	$15m^2$	495	20
	room 2	$9m^2$	297	12		room 2	$10m^2$	330	13.3
	room 3	$10m^2$	330	13.3		room 3	$14m^2$	462	18.6

Table 2: Rooms physical characteristics.

We collect the external temperature data for the considered time horizons, July 23 to 25 and January 08 to 09, 2022 on [20], which reports the weather data of the closest station to the community. The temperature data $T^{\texttt{out}}_h \ \forall\, h \in H$ is presented in Figure 5.

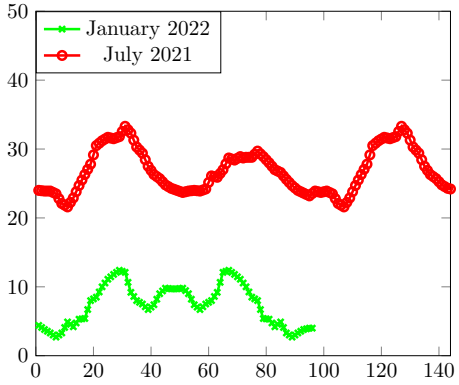


Figure 5: External temperatures for January 08 to 09, 2022, and July 23 to 25, 2021.

Each member has a single water heater, and the following table presents their physical characteristics. The parameter $c_p = 1$ is the specific heat capacity of water. The column M shows the weight of the water (kg), column S designates the exchange surface of the water container with the external area, K is the exchange coefficient (kcal/hm^2°C). Finally, recall that $T^{\texttt{room}}$ is the temperature of the ambient environment and the efficiency of the electricity-to-heat transformation $\upsilon = Mc_p/(Mc_pr + \Delta SKr)$.

Member	S	M	K	r
1	15	75	1	1.2
2	7	350	1	1.2
3	2	100	1	1.2
4	3.75	200	1	1.2
5	2.4	150	1	1.2
6	2	100	1	1.2
7	2.6	150	1	1.2

Table 3: Water heaters characteristics.

At period $h^{\texttt{water}} = 36$, the water's temperature in the heater of each member must be between 55 and 60 °C for initial temperatures represented in $\bar{T}^{\texttt{water}} = [9; 8; 8; 9; 5; 8; 9]$, and an ambient temperature $T^{\texttt{room}} = 17$ °C for each member. For these instances, we don't have type B tasks without lost of generality. As periodic cumulative consumption of type C tasks, we take 10% of the real energy consumption of the considered days for each member. Finally, the energy buying and selling prices per kWh are:

$$\upsilon_h^{\texttt{MG}} = 0.1685,\ \tilde{\upsilon}_h^{\texttt{MG}} = 0.1$$
$$\upsilon_h^{\texttt{Com}} = 0.1400,\ \tilde{\upsilon}_h^{\texttt{Com}} = 0.12$$
$$\upsilon_h^{\texttt{GES}} = 0.1685,\ \tilde{\upsilon}_h^{\texttt{GES}} = 0.065.$$

and the threshold of economic degradation which must not be exceeded is $\beta = 15$.

6.1.3 Instance 7 ex

Here we present a seven members instance called: **7 ex**, which we use to illustrate our remarks. Instance 7 ex characteristics in terms of assets possession, is described in Table 1. The planning horizon is 24 hours sliced into periods of 30 minutes. The periodic total production and the external temperature for instance 7 ex are presented by Figure 6.

Each member has at most three houserooms of 9 m^2 and three water heater of 100 litres. The array ν present the required type A task by members.

$$\nu_{ij1}=[1\ 1\ ;\ 0\ 0\ ;\ 1\ 1\ ;\ 1\ 1\ ;\ 1\ 1\ ;\ 0\ 1\ ;\ 1\ 1\ ]$$
$$\nu_{ij2}=[1\ 0\ ;\ 0\ 0\ ;\ 0\ 0\ ;\ 0\ 0\ ;\ 1\ 0\ ;\ 1\ 0\ ;\ 1\ 0\ ]$$
$$\nu_{ij3}=[1\ 0\ ;\ 0\ 0\ ;\ 1\ 0\ ;\ 0\ 0\ ;\ 0\ 0\ ;\ 1\ 0\ ;\ 1\ 0\ ]$$

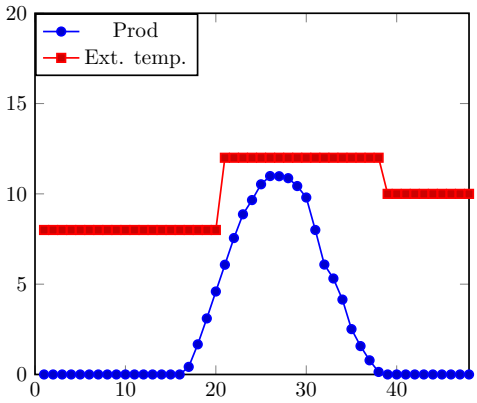


Figure 6: Values for illustrative instance.

$\forall i \in N, j \in J_{iA}$. The temperature preferences values for houserooms are: $h^{\texttt{low}}$=[22 12 22 22 22 22 19] $h^{\texttt{up}}$=[24 24 24 24 24 24 24] $t^{\texttt{low}}$=[10 16 20 20 34 18 20] $t^{\texttt{up}}$=[30 46 40 40 47 34 42]

Next section presents the results of the solution approaches on the described instance and the instance obtained by duplicating the real instance of seven community members.

6.2 Experimental results

This section reports the computational tests conducted to evaluate the two solution approaches presented in the previous sections. These tests have been carried out on a processor Intel Xeon E312 (Sandy Bridge) CPU2.29GHz, the MILP's solving time is t_l = 5600s, the column generation's pricing problem has a maximum time `time_limit` =200s, and the maximum number of column generation iterations is `maxIter` =10 for each instance. Finally, the RMP with integrality constraints has a time limit of `time_limit` =3600s. Notice that we consider two ways of solving the pricing problem: solving the pricing problem for all tasks at once or solving one pricing problem per task at each iteration. Let us denote these approaches as `PPo` and `PPs`, respectively. We limit the cumulative time of the CG iterations in both cases to `time_limit` =2000s which include initialization time of 200s.

The experiment shows that CG^{PPo} quickly finds a good solution. However, the method does not converge because, from one iteration to the other, the objective value of the RMP improves very little, as can be seen in Figure 7 which presents the evolution of the said value for instance 7 ex. For CG^{PPs}, which consists, at each iteration of solving the pricing problem for each task, the solution is improved quickly, and we can even find an optimal solution if the total number of tasks to be performed is small. The disadvantage is that solving a pricing problem for each task can be time-consuming. It is, therefore, necessary to make a trade-off between time and quality of the solution. If there is a need to save time, CG^{PPo} may be more appropriate, while CG^{PPs} may be interesting for small instances for which one may want to wait longer to get a better solution.

We conducted a scalability experiment, by constructing larger instances and duplicating the members characteristics up to 32 times, leading to $|N| = 224$ members in the community. For these larger instances, the MILP returns no integer solution after the time limit. On the other hand, the heuristic based on the column generation algorithm can return good solutions, even for the largest problem instances. The results of the solving approaches are reported in Table 4 where $|N|$ denotes the number of members in the community. *obj* and obj_{CG} represent, respectively, the sum of the electricity extracted from the primary grid during the planning horizon returned by the MILP and the column generation-based heuristic. *Bb* is the best bound obtained when executing F^{MILP} on the instances. Columns *Gap* are the gap between the objective and the best bound. Output "***" means that no feasible solution has been found after the time limit.

Figure 8 presents in detail all the results obtained by the discussed methods on instance 7 ex. Figure 8a corresponds to the first scenario: there are no energy exchanges between the community members. So, when a prosumer has surplus energy at a period h, it is injected into that battery or into the primary grid. Individuals who do not have production assets draw all their consumption from the public grid. When there are no internal links between members, the total amount collected from the primary grid

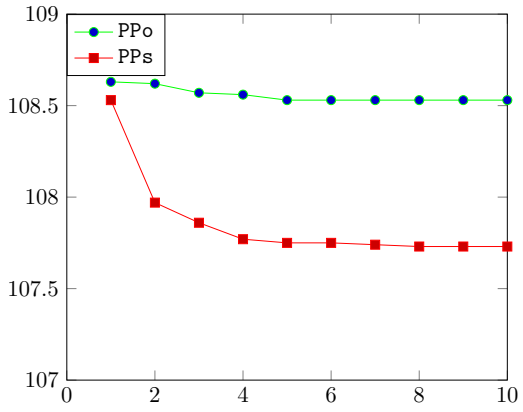


Figure 7: Objective value variation for the pricing problem solve ways.

	F^{MILP}'s solutions				CG^{PPo}'s solutions			CG^{PPs}'s solutions		
$\|N\|$	*obj* kWh	*Gap* %	*Bb* kWh	CPU (s)	obj_{CG} kWh	CPU (s)	*Gap* %	obj_{CG} kWh	CPU (s) (s)	*Gap* %
7 ex	107.74	0.56	107.18	t_l	108.53	1078.52	1.33	107.73	300.01	0.55
7	233.74	1.26	230.79	t_l	234.24	2010.31	1.54	234.24	577.73	1.54
28	938.96	1.19	927.81	t_l	940.98	2023.66	1.41	938.96	2394.09	1.19
56	1877.92	1.20	1855.44	t_l	1881.94	2036.56	1.41	1877.94	2381.98	1.20
112	***	***	3598.52	t_l	**3675.90**	**2188.07**	**2.11**	**3663.03**	**2419.62**	**1.76**
224	***	***	7198.23	t_l	**7360.32**	**2362.23**	**2.20**	**7321.15**	**2464.65**	**1.68**

Table 4: Comparison between the solutions of the resolution approaches.

equals 132.11 kWh. When they form a community in the second scenario (Figures 8b to 8d), in this example, there is almost no injection into the primary grid, the photovoltaic energy produced by the community members is consumed locally. As a result, the community collects 107.75, corresponding to a decrease of almost 18.5% in non-green power. The members make savings because the purchase price in the community is more attractive than the purchase price in the primary grid.

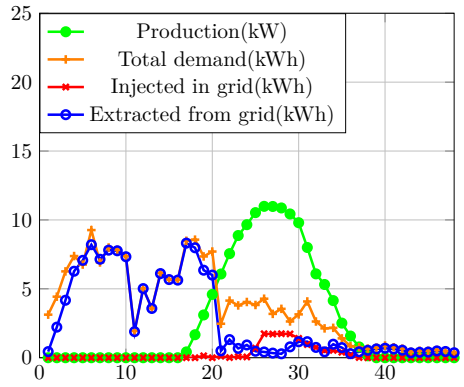
6.3 Several consecutive days

In what follows we present the results of the models when the number of time periods increases. We compare the results of the different solution approaches: F^{MILP} with the index of the day, the heuristic presented in Algorithm 2: F^{MILP}_{index} and the column generation-based heuristics CG^{PPo} and CG^{PPs}. We compare the models for three instances for which the number of time period $|H|$ is in $\{48,\ 96,\ 144\}$ and data collected from January 8 to 9, 2022, and from July 23 to 25, 2021. Table 5 presents these results where `time_limit` $= t_l = 5600$s for each approach.

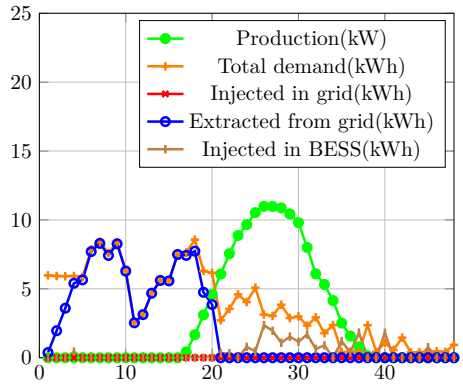
For January 8 to 9,2022										
	F^{MILP}'s solutions with day index				F^{MILP}_{index}'s solutions		CG^{PPo}'s solutions		CG^{PPs}'s solutions	
$\|H\|$	*obj* kWh	*Gap* %	*Bb* kWh	CPU (s)	obj_H kWh	CPU (s)	obj_{CG} kWh	CPU (s)	obj_{CG} kWh	CPU (s)
48	233.74	1.26	230.79	t_l	233.74	$0.5t_l$	234.74	2010.31	234.24	577.73
96	431.05	0.99	426.78	t_l	432.04	t_l	431.97	2014.65	431.46	2279.72
For July 23 to 25, 2021										
	F^{MILP}'s solutions with day index				F^{MILP}_{index}'s solutions		CG^{PPo}'s solutions		CG^{PPs}'s solutions	
$\|H\|$	*obj* kWh	*Gap* %	*Bb* kWh	CPU (s)	obj_H kWh	CPU (s)	obj_{CG} kWh	CPU (s)	obj_{CG} kWh	CPU (s)
48	109.17	0.31	108.83	t_l	109.17	$0.5t_l$	114.83	2824.74	109.17	1924.12
96	521.14	0.17	520.26	t_l	521.04	$0.5t_l$	523.37	2001.81	521.27	1639.83
144	614.60	0.32	612.63	t_l	613.54	$0.5t_l$	621.73	2056.48	614.20	2358.28

Table 5: Comparison between the solutions of the resolution approaches.

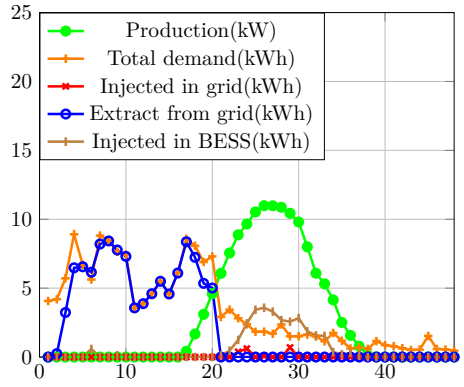
Summer instances sometimes are easier to solve than winter ones. Indeed, the external temperatures



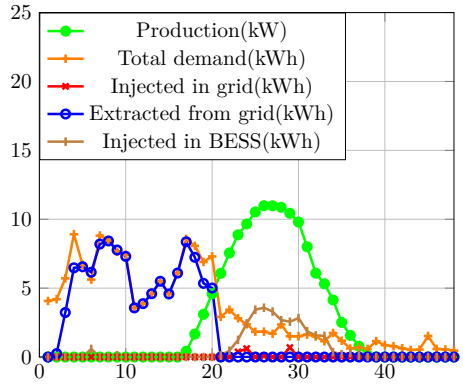
(a) Community without internal links.



(b) Community with internal links: F^{MILP}.



(c) Community with internal links: CG^{PPo}.



(d) Community with internal links: CG^{PPs}.

Figure 8: Solutions for instance 7 ex.

are not too far from the target temperatures when type A tasks are performed, except for heat waves. Moreover, knowing that the problem's difficulty comes from planning, the problem is easier in this case because no planning is required. Besides, CG^{PPs} remains more efficient concerning solution quality if one is willing to allow more solution time, while the quality of CG^{PPo}'s solutions degrades as the time horizon increases. On the other hand, Algorithm 2 can be used for small communities but not for large problem instances for the previously discussed reasons.

7 Numerical improvements

7.1 Special Ordered Set variables

We discuss here an enhancement that reduces the solving times of the previous model. Notice that at each period and for each task we can choose at most one power level to perform type A tasks. We can thus replace inequality constraint (1c) by an equality constraint by adding a dummy power level $p_0 = 0$ in each set $P^{\mathtt{A}}_{ijk}$. Then, we sort the powers in ascending order and introduce a binary variable $\hat{x}_{ijkh(pos(p))}$ for $p \in P^{\mathtt{A}}_{ijk} \cup \{0\}$ that is related to variables x^A_{ijkhp} through the relations:

$$x^A_{ijkhp} = \hat{x}_{ijkh(pos(p))} - \hat{x}_{ijkh(pos(p)+1)} \qquad \forall i \in N, j \in J_{iA}, k \in K^A_{ij}, h \in H, p \in P^{\mathtt{A}}_{ijk}. \tag{9}$$

Thus, constraint (1c) is replaced by constraints (10) and (11),

$$\hat{x}_{ijkh(pos(p))} \geq \hat{x}_{ijkh(pos(p)+1)} \qquad \forall i \in N, j \in J_{iA}, k \in K^A_{ij}, h \in H, p \in P^{\mathtt{A}}_{ijk} \tag{10}$$

$$\hat{x}_{ijkh0} = 1 \qquad \forall i \in N, j \in J_{iA}, k \in K^A_{ij}, k \in T \tag{11}$$

Finally, constraint (1j) becomes constraint (12) where $v(p)$ is the power level directly superior to p (that is, $pos(v(p)) = pos(p) + 1$)

$$T_{ijkh} = \theta_j \left(\sum_{p \in P^{\mathtt{A}}_{ijk}} (p - v(p)) \hat{x}_{ijkh(pos(p))} \right) \qquad \forall i \in N, j \in J_{iA}, k \in K^A_{ij}, h \in H. \tag{12}$$

Table 6 reports the results obtained with the different solving approaches after that replacement.

	F^{MILP}'s solutions				CG^{PPo}'s solutions			CG^{PPs}'s solutions		
$\|N\|$	*obj* kWh	*BB* kWh	Gap %	CPU (s)	obj_{CG} kWh	CPU (s)	*Gap* %	obj_{CG} kWh	CPU (s)	*Gap*%
7 ex	107.73	107.25	0.45	t_l	108.80	226.32	1.53	107.73	339.85	0.45
7	233.74	231.79	0.83	t_l	233.74	241.82	0.83	233.74	2242.34	0.83
28	938.96	931.28	0.82	t_l	938.96	243.18	0.82	939.47	2221.61	0.88
56	1877.92	1861.71	0.82	t_l	1877.92	312.17	0.82	1878.95	2358.13	0.92
112	3659.84	3624.90	0.95	t_l	3659.84	412.75	0.95	3659.90	2244.12	0.95
224	***	7219.16	***	t_l	7380.25	2464.28	2.18	7334.55	2425.97	1.57

Table 6: Comparison between the solutions of the resolution approaches.

By comparing Table 6 with the previous Table 4, we notice an improvement in the solving time and the solution's quality of heuristic `PPo`, in opposition to `PPs`. We also see that F^{MILP}'s Best bound are slightly better than before.

7.2 Heuristic enhancement

An improving track for the column generation-based heuristics is to return the first integer solution returned by the pricing problems. Since an improving solution for a task must have a negative reduced cost, we add constraints $A \leq \epsilon$, where A represents the reduced costs. Then, we set up CPLEX to return the first integer solution. Table 7 contains the results of that improving track, where `maxIter` $= 10$, and $\epsilon = 10^{-2}$.

Before replacing x^A by $\hat{x}$						
	CG^{PPo}'s solutions			CG^{PPs}'s solutions		
$\|N\|$	obj_{CG} kWh	CPU (s)	*Gap* %	obj_{CG} kWh	CPU (s)	*Gap* %
7	233.74	430.93	0.83	233.74	413.14	0.83
28	938.96	447.52	0.82	938.96	475.88	0.82
56	1877.84	491.71	0.82	1877.82	482.83	0.82
112	3674.84	698.4	1.36	3659.92	624.29	0.95
224	7319.67	789.88	1.37	7353.02	1093.38	1.82
After replacing x^A by $\hat{x}$						
	CG^{PPo}'s solutions			CG^{PPs}'s solutions		
$\|N\|$	obj_{CG} kWh	CPU (s)	*Gap* %	obj_{CG} kWh	CPU (s)	*Gap* %
7	233.74	426.09	0.83	233.74	419.22	0.83
28	938.96	436.09	0.83	938.96	474.24	0.82
56	1877.92	456.77	0.82	1877.92	478.07	0.82
112	3659.84	606.99	0.95	3659.92	404.53	0.82
224	7320.08	1239.01	1.38	7319.87	1031.89	1.37

Table 7: Returning the first integer solution found by PP.

Comparing to Tables 4 and 6 we note that the resolution times are generally improved as shown in Table 7. Specifically, CG^{PPo} is more efficient regarding time before and after replacing x^A by $\hat{x}$. However, we notice a difference for CG^{PPo} according to whether we consider x^A or $\hat{x}$, indeed, the replacement of x^A by $\hat{x}$ increases the CPU. Also CG^{PPs} is faster with this track compared to Tables 4 and 6, however, `PPs` does not find better solutions at relatively small iteration numbers. This track does not improve the quality of the solutions of either CG^{PPo} or CG^{PPs}. Moreover, the quality of the solutions degrades more when x^A is replaced by $\hat{x}$ for both CG^{PPo} and CG^{PPs}.

To summarize, we assumed that introducing SOS could induce an improvement of the problem resolution. After the experiments, we notice on the one hand, that the resolution times are significantly improved for CG^{PPo} and CG^{PPs}. On the other hand, by returning the first integer solution for the heuristic we notice an improvement of the resolution time before and after replacing x^A by $\hat{x}$ compared to Table 4. However, the quality of CG^{PPs}'s solution is deteriorated when x^A is replaced.

8 Conclusions and perspectives

In this paper, we address the optimization of collective self-consumption in an energy community. We assumed that the scheduling of the operation of electrical appliances coupled with a smooth simulation of the operation of electric devices owned by the community members would allow modifying the global consumption curve while satisfying the global constraints of the community and the individual constraints of the energy community members. We then developed a mixed-integer linear programming model to obtain the optimal schedules for the resulting planning problem. Then, faced with the inability of the MILP to solve large instances of the problem, we implemented a heuristic based on the column generation algorithm to overcome this problem. The tests demonstrate that the assumed strategy of scheduling loads can bring significant advantages (economical, sustainable and social). This validates the Smart Lou Quila demonstrator on realistic instances and ensure that bigger community will still collect benefice of this technology.

We considered a community of $|N|$ individuals, and we encounter difficulties when $|N|$ is large, which is predictable. One solution is to consider m sub-communities. However, optimizing separately in each community does not lead to optimized management of the large community. It would be interesting to explore this avenue. On the other hand, our current work assumes that everything takes place in a certain environment, which is not the case in practice. We could explore the track of inline planning or consider contingencies.

References

[1] Sezin Afşar, Luce Brotcorne, Patrice Marcotte, and Gilles Savard. Revenue optimization in energy networks involving self-scheduled demand and a smart grid. *Computers & Operations Research*, 134:105366, 2021.

[2] Anonymous. Scala graduum caloris. calorum descriptiones & signa. *Philosophical Transactions*, 22:824–829, 1701.

[3] Beoga/Le Cailar. External temperature. https://beoga.fr/en/smart-lou-quila/.

[4] Raffaele Carli, Mariagrazia Dotoli, Jan Jantzen, Michael Kristensen, and Sarah Ben Othman. Energy scheduling of a smart microgrid with shared photovoltaic panels and storage: The case of the ballen marina in samsø. *Energy*, 198, 2020.

[5] European Commission. Clean energy for all europeans package. https://ec.europa.eu/energy/topics/energy-strategy/clean-energy-all-europeans_en, (accessed: 30.11.2021).

[6] Giuseppe Tommaso Costanzo, Guchuan Zhu, Miguel F. Anjos, and Gilles Savard. A system architecture for autonomous demand side load management in smart buildings. *IEEE Transactions on Smart Grid*, 3(4):2157–2165, 2012.

[7] Guven Denizhan, Kayalica M. Ozgur, and Kayakutlu Gulgun. Critical power demand scheduling for hospitals using repurposed ev batteries. *Technology and Economics of Smart Grids and Sustainable Energy*, 6, 2021.

[8] Jacques Desrosiers, Franpis Sournis, and Martin Desrochers. Routing with time windows by column generation. *Networks*, 14:545–565, 1984.

[9] Junwen Ding, Sven Schulz, Liji Shen, Udo Buscher, and Zhipeng Lü. Energy aware scheduling in flexible flow shops with hybrid particle swarm optimization. *Computers & Operations Research*, 125:105088, 2021.

[10] Romaric Duvignau, Verena Heinisch, Lisa Göransson, Vincenzo Gulisano, and Marina Papatriantafilou. Benefits of small-size communities for continuous cost-optimization in peer-to-peer energy sharing. *Applied Energy*, 301:117402, 2021.

[11] Edstan Fernandez, M.J. Hossain, Khizir Mahmud, Mohammad Sohrab Hasan Nizami, and Muhammad Kashif. A bi-level optimization-based community energy management system for optimal energy sharing and trading among peers. *Journal of Cleaner Production*, 279:123254, 2021.

[12] Hans Kellerer, Ulrich Pferschy, and David Pisinger. *Introduction to NP-Completeness of Knapsack Problems*, pages 483–493. Springer Berlin Heidelberg, 2004.

[13] Thillainathan Logenthiran, Dipti Srinivasan, and Tan Zong Shun. Demand side management in smart grid using heuristic optimization. *IEEE Transactions on Smart Grid*, 3(3):1244–1252, 2012.

[14] G. Pontes Luz, M.C. Brito, J.M.C. Sousa, and S.M. Vieira. Coordinating shiftable loads for collective photovoltaic self-consumption: A multi-agent approach. *Energy*, 229:120573, 2021.

[15] Anjos F. Miguel, Luce Brotcorne, Martine Labbé, and Maria Restrepo Ruiz. Load Scheduling for Residential Demand Response on Smart Grids. Optimization Online eprints (6384).

[16] Ricardo Moura and Miguel Centeno Brito. Prosumer aggregation policies, country experience and business models. *Energy Policy*, 132:820–830, 2019.

[17] Vera Reis, Rita H. Almeida, José A. Silva, and Miguel C. Brito. Demand aggregation for photovoltaic self-consumption. *Energy Reports*, 5:54–61, 2019.

[18] Miadreza Shafie-Khah and Pierluigi Siano. A stochastic home energy management system considering satisfaction cost and response fatigue. *IEEE Transactions on Industrial Informatics*, 14(2):629–638, 2018.

[19] Gulai Shen, Zachary E. Lee, Ali Amadeh, and K. Max Zhang. A data-driven electric water heater scheduling and control system. *Energy and Buildings*, 242:110924, 2021.

[20] Météo station Nîme/Courbessac. External temperature. https://prevision-meteo.ch/climat/journalier/nimes-courbessac/2021-07 (accessed: 11.04.2022).

[21] Anuradha Tomar, D.S. Shafiullah, P.H. Nguyen, and Marcel Eijgelaar. An integrated flexibility optimizer for economic gains of local energy communities — a case study for a university campus. *Sustainable Energy, Grids and Networks*, 27:100518, 2021.

[22] Wayes Tushar, Tapan Kumar Saha, Chau Yuen, Thomas Morstyn, Nahid-Al-Masood, H. Vincent Poor, and Richard Bean. Grid influenced peer-to-peer energy trading. *IEEE Transactions on Smart Grid*, 11(2):1407–1418, 2020.

[23] Wim van Ackooij, Jérôme De Boeck, Boris Detienne, Stefania Pan, and Michael Poss. Optimizing power generation in the presence of micro-grids. *Eur. J. Oper. Res.*, 271(2):450–461, 2018.