Towards Sustainable Energy-Efficient Communities Based on a Scheduling Algorithm

Carlos Cruz * , **Esther Palomar** , **Ignacio Bravo** and **Alfredo Gardel**

Department of Electronics, University of Alcala, Alcala de Henares, 28871 Madrid, Spain;
esther.palomar@uah.es (E.P.); ignacio.bravo@uah.es (I.B.); alfredo.gardel@uah.es (A.G.)
* Correspondence: carlos.cruzt@uah.es; Tel.: +34-91-8856-589; Fax: +34-91-885-6591

Abstract: The Internet of Things (IoT) and Demand Response (DR) combined have transformed the way Information and Communication Technologies (ICT) contribute to saving energy and reducing costs, while also giving consumers more control over their energy footprint. Unlike current price and incentive based DR strategies, we propose a DR model that promotes consumers reaching coordinated behaviour towards more sustainable (and green) communities. A cooperative DR system is designed not only to bolster energy efficiency management at both home and district levels, but also to integrate the renewable energy resource information into the community's energy management. Initially conceived in a centralised way, a data collector called the "aggregator" will handle the operation scheduling requirements given the consumers' time preferences and the available electricity supply from renewables. Evaluation on the algorithm implementation shows feasible computational cost (CC) in different scenarios of households, communities and consumer behaviour. Number of appliances and timeframe flexibility have the greatest impact on the reallocation cost. A discussion on the communication, security and hardware platforms is included prior to future pilot deployment.

Keywords: cooperative smart community; scheduling algorithm; consumer preferences; renewables

1. Introduction

There exists a global aim to conceive novel sustainable services and energy infrastructures to balance supply and demand. Over the last decade, many sustainable development initiatives across the globe have been promoting regulatory campaigns, such as pricing or optional/mandatory thermal retrofit policies, looking at the engagement of cost-effective social behaviour and/or a social pro-environmental morality [1]. To this regard, the Internet of Things (IoT) and Demand Response (DR) combined have transformed the way Information and Communication Technologies (ICT) contribute to saving energy and reducing costs, while also giving consumers more control over their energy footprint [2,3]. Connected devices (e.g, household items, machines, vehicles or gadgets) can automatically influence each other in order to increase the overall potential for energy efficiency and the range of management systems' involvement.

DR programmes, designed to stimulate changes in consumers' electric usage patterns, thus appear to bolster not only energy efficiency, but also renewable energy resource management initiatives. Current DR strategies are based on providing end-users with individualised tailored advice about their particular habits with incentive payments for load reductions when needed to ensure reliability [4]. For instance, as control and communication technologies become more widely accessible, electricity prices and information are delivered more effectively to consumers. This allows consumers to identify and more easily target discretionary loads that can be curtailed or shifted. On one hand, we can find new challenges to the analysis of these loads and the extraction of consumer/community patterns that produce more automatic and user-friendly DR systems as

well as driven by congestion management instead of being price-based. On the other hand, this automation should be enabled by on-site energy controls fed by near-real-time pricing information without significant customer effort or intervention. Furthermore, the real exploitation of renewable sources for energy supply presents multiple challenges not only to utilities, grid and system operators, but also to the consumer that knows very little about its availability or potential from microproduction [5] and energy harvesting processes [6,7]. For instance, according to the Eurostat survey (https://ec.europa.eu/eurostat) (Figure 1), only 19% of the final energy consumption in residential sector comes from renewable resources.
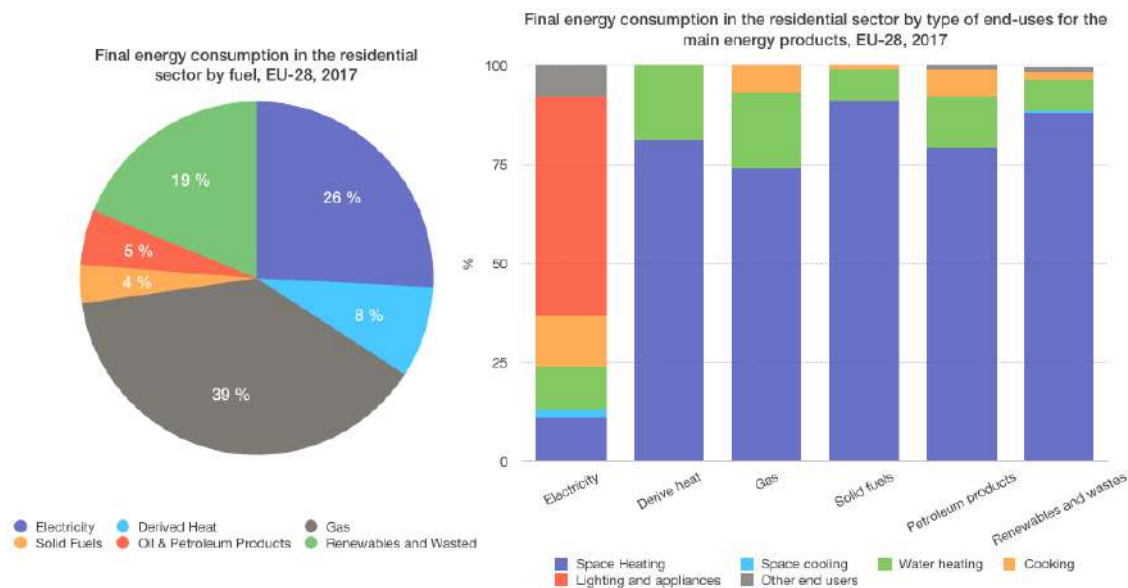

Figure 1. Final energy consumption in the European residential sector from Eurostat survey 2017.

Our proposal intends to bridge the aforementioned gap between utilities and consumers by leveraging consumer cooperation towards a joint daily schedule of their household appliances operation using supply generated from renewable energy sources. In this work, we assume the existence of an Utility entity (a set of energy providers or substations) generating, accumulating, storing and ultimately serving electricity to the consumers. This role, the Utility, is therefore in charge of allocating the available supply from the different energy sources at disposal of the community; it is not, however, dealing with the final destination of the supply (whether to power low-energy electronics or bigger appliances). As an application scenario, imagine a smart community of electricity consumers who, empowered by a better access to their consumption controls and appliance interconnection, are provided with sufficient incentives to coordinate and adjust their energy demands for a certain purpose. These consumption controls are coordinated by a Home Energy Management System, which enables energy management at homes. By doing this, consumers are able to visualise the energy data and make optimum use of energy by controlling their electrical appliances. They autonomously adapt their energy consumption by means of sharing nearly real-time electricity demand information. An aggregator device, capable of shifting the consumers' use of the resource, will be able to make the overall consumption pursue common goals such as being sustainable, ecofriendly or cheaper. On the other hand, utilities, allowed to perform real-time billing, profiling and fault detection, are also creating incentives for users consuming renewable sources (e.g., guaranteeing the lowest price if the load demand does not exceed a certain threshold). They produce, store, distribute and serve the supply to the consumers who will now benefit from additional information about the supply availability. It has to be observed that neither utilities nor consumers are considering microgenerated energy in the current model. Hence, it is responsibility of the aggregator the computation and rescheduling

of the total daily load of the community to avoid overloading the utility supply from renewables. This scheduling represents an optimisation problem whose main factors are the 24 h-vector of the next day's supply from renewable sources and the duration and activation time preferences of every consumer's appliance.

In this work, we present a cooperative DR system designed to promote behavioural changes in small or large communities with common interests. The involved entities will reach binding agreements and coordinated behaviour through the aggregator, a device that collects 24-h vectors with the consumers' demands and the expected supply from renewables. It also centralises the supply allocation algorithm that optimises the distribution of the available green supply between the consumers taking into consideration their time preferences of appliance activation. Experimentation on the algorithm implementation is conducted using estimated values and benchmarks. We include the analysis of the power consumption in watts for most commonly used appliances taking average measures informed by manufacturers; for each appliance type, we show the efficiency label (according to EU normative), the estimated cost while in operation mode and standby mode as well as the average consumption in 24 h time. In addition, we evaluate the algorithm over different strategies of player order selection as well as over the application of four heuristics that optimise the objective search. Evaluation results throw feasible computational cost in all these different programming configurations as well as considering a series of scenarios for household and community settings, and consumer behaviours. Finally, from the empirical results we can discuss on the hardware and networking requirements for an efficient pilot deployment.

The paper is organised as follows. We discuss the related work in Section 2. Section 3 states the system model and design decisions. We describe the simulation of the implemented scheduling algorithm and estimate the performance cost in Section 4. Technical considerations in terms of communication, network protocols, security and hardware platforms are drawn in Section 5. Finally, Section 6 concludes and establishes future research directions.

2. Related Work

The starting point of our research can be found in the works by the authors of [8,9], where an adaptive model for DR is envisioned over the deployment of smart meter networks. Special focus is taken on the software design in order to facilitate the integration and scalability of the community system future development. An example of a DR aggregator model is designed in the work by the authors of [10] to facilitate renewable energy integration, where end consumers play a key role.

One of the major challenges in the energy efficiency context is the way to involve end-users in energy markets. This fact can be exemplified in the works by the authors of [11,12], where systems are designed to facilitate DR for residential prosumers. For instance, the work by the authors of [12] shows a system based on an aggregator of residential prosumers that participate in the day-ahead energy market to minimise operation costs by controlling appliances. The performance of an optimisation-based residential energy management scheme is presented in the work by the authors of [13]. This work applies a constrained swarm intelligence model to minimise the total cost of household electricity consumption. As it has been stated by the authors of [14], models based on DR, smart technologies and intelligent controllers can lead to a considerable energy consumption reduction.

The vast majority of the related work addresses energy-efficient solutions and optimisation algorithms from a single consumer/home viewpoint. The appliance scheduling optimisation solution in the work by the authors of [15] considers time ranges and consumer preferences along with different types of appliance consumption profiles. Their solution is based on the Mixed Integer Linear Programming (MILP) technique under "Gurobi" solver, which is addressed to minimise both the total energy cost and the peak load of all the home appliances used per day. This model is unscalable though. Similarly, household load scheduling is also approached in the work by the authors of [16] by the MILP optimisation model. The MILP model and a heuristic algorithm accounting for a typical household user are simulated taking into account overall costs, climatic comfort level and timeliness.

MILP is also the technique applied to load shifting by the authors of [17] to optimise the interaction between an aggregator and smart consumers' operation. The specific DR program incentives and the consumers' needs are the main parameters that a Smart Home (SH) controller considers to reshape the consumers' demand profile through shifting the operation of flexible loads. Focused on users' individual preferences, the work by the authors of [18] sets priorities and preferred time intervals for load scheduling, along with making efforts to optimise the consumption curves of household, commercial and industrial consumers.

A remarkable interaction between the utility and its consumers is modelled through a two-step centralised game in [19], where consumers reduce the peak-to-average power ratio by optimising their energy schedules. The utility supplier pulls consumers in a round-robin (RR) fashion and provides them with energy price parameter and current consumption summary vector. Each user, then, optimises its own schedule and reports it back to the supplier, which, in turn, updates its energy price parameters before pulling the next consumers. Also centralised but considering renewable energy technologies to improve energy efficiency and reduce costs through optimisation algorithms, approaches in the work by the authors of [20] focus on the context of microgrids and storage at residential and commercial building environments. In addition, heuristics based on genetics algorithms [21] and neural networks [22] work on the scheduling of the consumer consumption to save the peak formation. Their simulation results show that the proposed algorithms reduce the peak-to-average ratio and help users minimise their energy expenses without compromising comfort.

Applying a distributed and an autonomous Demand Side Management (DSM) within a neighbourhood, the consumers' schedulers in the work by the authors of [23] are assumed to be built inside smart meters and connected to the power grid and a local area network. In order to reduce the total energy cost, these schedulers interact automatically by running a distributed algorithm to find the optimal energy consumption schedule. Subscribers also receive incentives to use the schedulers via a novel pricing model derived from a game-theoretic analysis. The authors of [24] formulate a power allocation game, where multiple companies, leaders and their consumers are the followers to reach a unique pure-strategy Nash equilibrium via a distributed algorithm. Authors find that the multi-period scheme, compared with the single-period one, provide more incentives for energy consumers to participate in DR. For a comprehensive description of the many algorithms that can be used to solve the resource allocation problem, see the work by the authors of [25].

In summary, several optimisation Pareto-efficient approaches to the load and/or consumption adaptive scheduling have been the focus of much attention in demand side management, SHs, wireless sensor networks, broadband networks, and smart grids [26,27].

3. System Model

Our proposal embraces the use of renewable resources aiming three main actors: *Consumer*, *Aggregator* and *Utility*. Figure 2 illustrates the main roles and processes within the adopted cooperative DR framework.

The first actor, *Consumer*, provides the home energy usage to be managed and automatically controlled by Home Energy Management System (HEMS) that performs three main functions: (1) schedule demand, (2) appliance control and (3) information provider. It selects the daily scheduling preferences, managing a profile for collaboration in a DR system and viewing its account and consume information. The consumer can manage them from a portable device (i.e., an app installed on a mobile phone or tablet) that is connected to a communication network for preferences scheduling. A community will comprise a set of consumers sharing electricity supplier or substation. HEMS pulls scheduling information and generates processed data to the *Aggregator*. HEMS is also responsible for collecting information from the *Aggregator* and controlling a variety of home appliances.
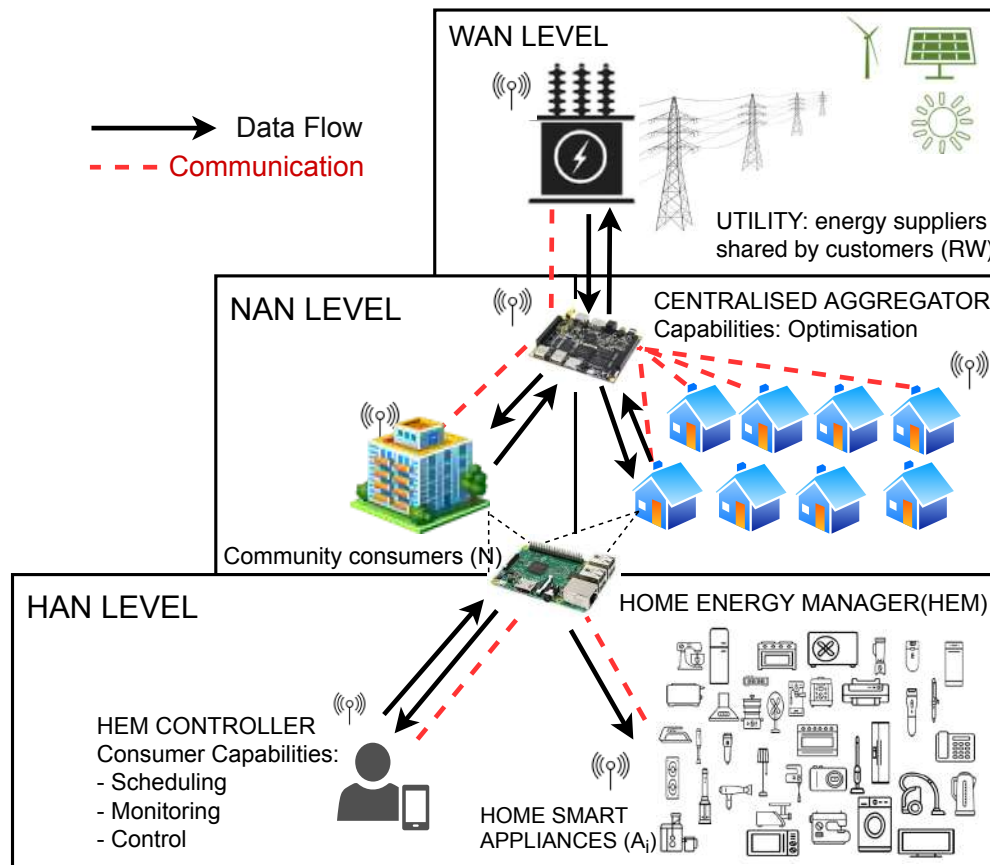
Figure 2. Smart cooperative system divided into Home Area Network (HAN), Neighbour Area Network (NAN) and Wide Area Network (WAN).

In our proposal, *Consumers* adapt their energy consumption cooperatively on a centralised way, that is, sharing their demand schedule with a data collector, which facilitates the integration of energy consumption information into a common view. This integration is performed over the so-called *Aggregator*, the second actor, which implements an optimised resource allocation algorithm as a response to supply conditions, in particular, targeting renewable sources. The *Aggregator* is defined as the optimal system providing energy management services in order to efficiently manage demand in SH [28]. HEMS acts as a central node and receives the demand scheduled information from the *Aggregator*. Then it loads the power consumption preferences to each appliance and establishes communication for managing the appliances. The *Aggregator* allows the local distribution of the energy provided, according to the availability of renewable resources. This energy management system will be connected to the *Utility*, the third actor, which is a set of energy suppliers shared by customers. We presume utilities implement a distributed generation that allows to gather energy from mainly renewable sources addressed to give lower environmental impacts and improve supply security.

3.1. Consumer System Design

Let $\mathcal{N}$ denote an ordered set of *Consumers* that are willing to cooperate in the pursuit of global community targets (i.e., becoming greener) by sending their data to the *Aggregator*. Each *consumer* $i \in \mathcal{N}$ has a set of household appliances labeled as $\mathcal{A}_i$. Fixed energy load is identified by factors such as the consumers' habits, their behaviours and their use of appliances, as well as a variable load resulting from the use of such appliances and other equipments. Formulae and benchmarks can be used to

estimate appliance and home electronic energy use in kilowatt hours (kWh) as well as household local records.

Bearing in mind a discrete time slot system, and without loss of generality, we assume that time granularity is one hour per day. Regarding the appliances, each *Consumer* is supposed to preallocate a certain amount of fixed demand and variable consumption planned for the next 24 h [29]. For each appliance, $a_{ij} \in \mathcal{A}_i$, we assume both daily fixed and variable energy consumption scheduling vectors at each time slot $t \in \{0, \dots, 23\}$ to control its non-shiftable and shiftable consumption respectively.

We define $f\mathcal{D}^t_{i,a_{ij}}$ and $v\mathcal{D}^t_{i,a_{ij}}$ by denoting the corresponding one-hour fixed and variable energy consumptions respectively. Variable energy demand is characterised by its flexibility, as the *Consumer* preference for an appliance starting in a specific period of time is taken into consideration. For each appliance, there is an execution window (i.e., a closed interval) that selects a minimal starting time, and a maximal ending time labelled by t^i_{beg} and t^i_{end}. t^i_{sched} is defined as the working time of appliance "i" and matches the range of operation start time $t^i_{sched} \in [t^i_{beg}, t^i_{end}]$. $\mathcal{L}$ is defined as the duration of the planned operation of appliance a_{ij} in the next day. Load needs to be switched on for a time between two predefined moments: $\forall_{ij} \in \mathcal{A}_i, t^i_{sched} \geq t^i_{beg}$. In this line, load also needs to be switched off: $\forall_{ij} \in \mathcal{A}_i, t^i_{sched} \leq t^i_{end}$. In other words, *Consumer i* will set the following data for its appliance $a_{ij} \in \mathcal{A}_i$ (see Table 1).

Table 1. Appliance configuration.

Appliance Configuration				
Consumption (kWh)	Fixed consumption (kWh)	Duration (hours)	Time ON	Time OFF
$v\mathcal{D}^t_i$	$f\mathcal{D}^t_i$	$\mathcal{L}$	t_{beg}	t_{end}

- Fixed consumption (kWh) when appliance a_{ij} is in standby status
- Consumption (kWh) when a_{ij} is on
- Duration (hours/minutes) of the planned operation of appliance a_{ij} in the next day
- Point in time (hour, e.g., 8am) of preferred start of appliance a_{ij} activation
- Point in time (hour, e.g., 12pm) of preferred end of appliance a_{ij} operation

A centralised home controller provides access to all the appliances and devices at home via wireless networks; it will receive and apply the 24-hour reallocated vector from the *Aggregator* to systematically activate/deactivate every appliance without human intervention.

Moreover, we have developed an energy consumption scheduling app based on the Adobe XD template [30], which provides the consumer with an interface to control, monitor, visualise and program the functioning of appliances. More specifically, it allows the configuration and setting of the aforementioned data for each appliance a_{ij}. Figure 3 depicts a usage sequence to explain how the application works. The app allows users to check the resources used in the previous 24 h as well as to select the appliance in relation to the dwelling zones such as the kitchen or the bathroom, among others. At this stage, consumer will be able to indicate the time range and the duration of activation for each appliance. The last window summarises the introduced demand information. It also provides an estimated power cost in operation and standby for each appliance (according to benchmark analysis in Table 2). Consumers have to give consent by sending these data to the home controller. Finally, a vector is sent to the *Aggregator* with the data structure shown in Table 1.

Figure 3. Consumer scheduling app.

Data aggregation is defined as a centralised system with aggregation tasks that communicates with the *Utility* and the *Consumers* as shown in Figure 4. An algorithm is originally designed to optimise the allocation of the expected electricity supply from renewables among the community's *Consumers* related to their chosen preferences.
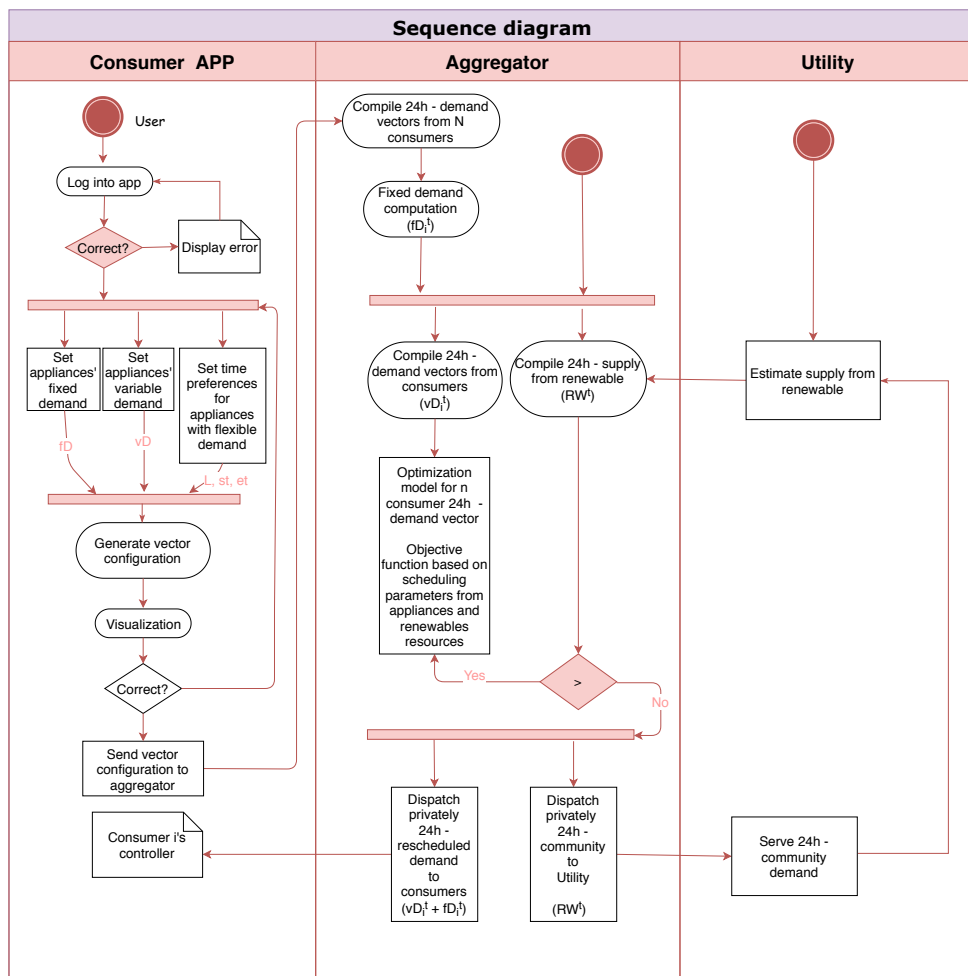

Figure 4. Sequence stating main processes and message exchange among the system's players.

3.2. Aggregator System Design

As an example of a renewable source efficient use, the *Utility* provides essential information on reliable renewable source and fossil energy programmed for the next 24 h by replacing carbon-intensive energy. The energy supply generated from a set of renewable sources at a time slot $t \in \{0, \ldots, 23\}$ is denoted by $\mathcal{RW}^t$. $f\mathcal{U}^t$ represents the energy supply at time t generated from a set of fossil sources. The *Utility* centralises the distribution of the energy, the notification to the *Aggregator*, and the billing process. The renewable supply vector $\mathcal{RW}$ is essential for the *Aggregator* in the optimisation of a fair allocation of such supply between the *Consumers'* fixed (non-shiftable) and variable (shiftable) energy demands.

The daily fixed demand for consumer $i \in \mathcal{N}$ is denoted by $f\mathcal{D}_i = \sum_{t=0}^{23} \sum_{a_{ij} \in \mathcal{A}_i} f\mathcal{D}_{i,a_{ij}}^t$ as the aggregated load of non-shiftable local consumption of the appliances and frequent behaviours. The *Aggregator* can then easily compute the daily fixed demand for the whole community of consumers at a time t as $f\mathcal{D}^t = \sum_i^N f\mathcal{D}_i^t$. On a daily basis, the *Aggregator* verifies that the total energy consumed by all appliances in the system fulfils the daily utility service provided by the *Utility*. It is critical that the community does not reach the worst case such as $\sum_i^N \sum_{t=0}^{23} f\mathcal{D}_i^t \gg \sum_{t=0}^{23} \mathcal{RW}^t$. On the contrary, aggregation of the variable energy is more complex given the consumers' time preferences. The *Aggregator* will execute a fair-share rescheduling of the community's requested variable demand per hour $v\mathcal{D}_i^t$ aiming at $\forall t \in \{0, \ldots, 23\}, \sum_i^N (f\mathcal{D}_i^t + v\mathcal{D}_i^t) \leq \mathcal{RW}^t$. We will show refinements of the proposed scheduler algorithm looking at the max–min fairness, Pareto-efficiency, envy-freeness, and truthfulness while serving Consumers' preferences. Perhaps the simplest way to give each Consumer equal chance against all other is to recursively apply a "round-robin" strategy in the allocation of each Consumer's needs. Fair random assignment is one of the refinement methods to be compared. A global centralised optimisation problem is faced here, where only a "Nash bargaining solution" is possible such as $\forall i \in \{1 \ldots \mathcal{N}\}, \mu_i^t = f\mathcal{D}_i^t + min\{\mathcal{DFC}(v\mathcal{D}_i^t)\} \leq \mathcal{RW}^t$.

Therefore, solutions to the optimisation problem should satisfy t_{sched}^i and $\mathcal{L}$ while avoiding overconsumption at $\mathcal{RW}^t$. The formulation is explained in Algorithm 1 (Demand Calculation Function, $\mathcal{DCF}$) and it will be shaped as its minimum, i.e., $min \, \mathcal{DFC}(\cdot)$ upon request of Algorithm 2. In $\mathcal{DFC}$ function, a search for the optimum time slot for every appliance activation takes place given its activation time, its preference interval and the available supply in kW from the renewable utility. In particular, taking into account $\mathcal{A}_i$, t_{sched}^i, t_{beg}^i and t_{end}^i variables, the optimisation will determine how appropriate an adjustment is by minimising the total overconsumption (in hours) of the community appliances against the available renewable supply at a certain time slot.

Finally, upon reaching the optimisation objective, the *Aggregator* will notify the community that an agreement has been reached and privately release the reallocated demand vector $\overrightarrow{\mu}_i \forall i \in \mathcal{N}$ to each Consumer.

Algorithm 1 Demand Calculation Function ($\mathcal{DCF}$)

1: RW^t(renewable vector) $= \{\sigma_1, \ldots, \sigma_{24}\}$
2: $\mathcal{N} = size(\mathcal{A}_i)$
3: Defining variables t^i_{beg}, t^i_{end}
4: **for** ihour time to the total number of hours **do**
5:     **if** ihour doesn't belongs to interval $[t^i_{beg}, t^i_{end}]$ **then**
6:        $f\mathcal{D}^t_i$ computation
7:     **end if**
8: **end for**
9: **for** iappliance 1 to size of appliances configuration ($\mathcal{A}_i$) **do**
10:     $\mathcal{RW}^t = \mathcal{RW}^t - -\mathcal{A}_i(f\mathcal{D}^t_i)$
11: **end for**
12: $\mathcal{A}_i(v\mathcal{D}^t_i) = \mathcal{A}_i(v\mathcal{D}^t_i) - -\mathcal{A}_i(f\mathcal{D}^t_i)$
13: $\mathcal{A}_i(\mathcal{D}^t_i(\mathcal{D}^t_i < 0)) = 0$
14: $\mathcal{A}_i(f\mathcal{D}^t_i) = \mathcal{A}_i(f\mathcal{D}^t_i) - -\mathcal{A}_i(f\mathcal{D}^t_i)$
    Objective Function $\mathcal{F}(\mathcal{A}_i, \mathcal{RW}^t, t^i_{sched})$
Require: $\mathcal{A}_i$ configuration: $v\mathcal{D}^t_i, f\mathcal{D}^t_i, \mathcal{L}_i, t^i_{beg}, t^i_{end}$
Ensure: $t^i_{beg} < t^i_{end}$
15: $\mathcal{HC}$ initialisation (consume Hourly Energy)
16: **for** iappliance 1 to size of appliance configuration **do**
17:     Set t^i_{beg}
18:     Set t^i_{end} based on $\mathcal{L}_i$ and t^i_{beg}
19:     **for** ihour time to the total number of hours **do**
20:        **if** ihour belongs to interval $[t^i_{beg}, t^i_{end}]$ **then**
21:           $\mathcal{HC}(ihour) \leftarrow \mathcal{HC}(ihour) + \mathcal{A}_i(v\mathcal{D}^t_i)$
22:        **else**
23:           $\mathcal{HC}(ihour) \leftarrow \mathcal{HC}(ihour) + \mathcal{A}_i(f\mathcal{D}^t_i)$
24:        **end if**
25:     **end for**
26: **end for**
27: $\mathcal{RW}^t s = \mathcal{RW}^t - -\mathcal{HC}^t$
28: $\mathcal{RW}^t s(\mathcal{RW}^t s < 0) = 0$
29: $Demanded\_\mathcal{RW}^t \leftarrow min(\mathcal{RW}^t, \mathcal{HC}^t)$
30: $\mathcal{R}1 = sum(\mathcal{RW}^t s); \mathcal{R}2 = max(\mathcal{HC}^t)$
31: $Result = sum(\mathcal{R}1 + \mathcal{R}2)$
32: **return** $Result, Demanded\_\mathcal{RW}^t, \mathcal{HC}^t(t^i_{sched})$

Algorithm 2 RR strategy

1: Generate parameters for consumer allocation
2: Define global variable $\mathcal{RW}$
3: **while** (user $< \mathcal{N}$) and (min($\mathcal{RW}$) >= 0) **do**
4:     **if** Optimisation needs **then**
5:        Load consumer preferences. $\mathcal{A}_i$ size from preference array
6:        Call Optimisation Function under variables preferences: $\mathcal{RW}, \mathcal{A}_i, \mathcal{N}$
7:        Number of user ++
8:        **if** ($\mathcal{RW}$ equals to 0) **then**
9:           Break
10:        **end if**
11:     **end if**(No consumer to optimise)
12: **end while**

3.3. Proposed Algorithm: A Fair Division Game

The *Aggregator* can apply different approaches to optimisation search within the aggregated load vector $v\mathcal{D}$. It contains all consumer appliances' scheduling that could be shifted within their preferred activation time frame. This scheduling problem at the *Aggregator* can be seen as a division game given a set of players (either the consumers in $\mathcal{N}$ or all its appliances) and a set of assets (the supply from renewables in $\mathcal{RW}$). In our algorithm we have opted for a turn-based sequential game played by Appliance instead of Consumer for optimisation purposes.

The scheduling problem has to produce a fair division of $\mathcal{RW}$, i.e., a set of rules that, when properly used by the players, guarantees at the end of the game each player will have received a fair share of the assets. In our view, a fair share means that consumers' preferences on the appliance activation are considered by the *Aggregator* with equity and privacy. As in turn-based sequential games, defining the order under which players start within a turn could be approached in terms of (A) Round-Robin (RR) start: the first player selection policy is RR; (B) random RR; (C) ranking: the first player being the same every time; and (D) randomness: the first player is randomly selected (likewise the sequence order), as follows.

(A)　The RR principle, known from other fields such as network scheduling and processor queuing, is based on a process/game/technique, where each task/person/device takes an equal share of something in turn. The RR scheduling can allocate the available electricity from renewables both simple and fairly among the *Consumers/ Appliances*, because (1) the consumers' number is known and fixed and (2) the reallocation process is centralised by the *Aggregator* which, starting on its own, will satisfy the demand of the *Consumers/Appliances* in a periodically repeated order. We include pseudocode of our algorithm's main function in the round robin strategy, being the rest pseudocodes similar with exception of the player turn selection on Algorithm 2-line 3. RR results in max–min fairness if the *Consumers/Appliances'* demands are equally sized; otherwise, fair queuing that establishes a fair share size would be desirable.

(B)　A random RR scheduling: A similar process as in A), though the election of first *Consumer* is random.

(C)　A picking-sequence has several merits as a fair division protocol [31]. Assuming that each agent has a (private) ranking over the set of objects, the allocator must find a policy (i.e., a sequence of agents that maximises the expected value of some social welfare function). Moreover, picking sequences are a natural way of allocating (indivisible) items to agents in a decentralised manner: at each stage, a designated agent chooses an item among those that remain available. The goal of the method is to identify the fairest sequence.

(D)　A random process could, or could not, introduce efficiency (no other "random" assignment dominates) in the aforementioned methods while keeping them Pareto-efficient, envy-free and giving good approximation to the social welfare. Efficiency in terms of computational time is also at stake.

4. System Validation

In this section, we measure the Computational Cost (CC) of the implemented scheduling algorithm evaluating the suitability of a number of four heuristics applied to the optimisation search and on a series of different case scenarios of consumer communities. In particular, the evaluation of the heuristics and their behaviour on our algorithm under the same input parameters will assist in the selection of the hardware platform for an efficient deployment.

4.1. Optimisation Algorithms Used

We adopt heuristic techniques to perform a partial random search of optimal solutions to our objective, i.e., either when the reallocation demand is met or when the number of predefined iterations is reached. We have identified and implemented the following four optimisation methods to guide the

search of a workable solution, i.e., the nearest local minimum standard strategy in our Algorithm 1 as follows.

(i) Simulated Annealing (SA) [32] finds a local minimum solution for our Algorithm 1 ($\mathcal{DCF}$) starting at an initial operation time t^i_{sched}. As explained in Algorithm 3, SA starts generating trial point based on current estimates and evaluates the function by accepting a new value generated after $\mathcal{T}$ parameter is set. The solution must consider the $[t^i_{beg}, t^i_{end}]$ time constraints. t^i_{sched} can randomly generate and filter by $\mathcal{L}$. In case of better $\mathcal{D}$, the original one $\mathcal{D}'$, $t^i_{sched'}$ could be accepted as better solution if $\mathcal{D}'$ is worst than $\mathcal{D}$. After the internal counter reaches its threshold, $\mathcal{T}$ is cooled down and re-select the best solution again with the reset counter.

Algorithm 3 Optimisation based on SA algorithm

1: Let $\mathcal{T} > 0$ as initial parameter
2: Let $\mathcal{N}(\mathcal{T})$ as maximum number of iterations
3: **while** stop criterion has not been met **do**

4:   Randomly generate a fasible solution t_{sched}
5:   Evaluate t^i_{sched}, $\mathcal{D} = \text{f}(t^i_{sched})$
6:   n = 1
7:   **while** while n $<= \mathcal{N}(\mathcal{T})$ **do**

8:     Generate solution $t^i_{sched'}$ based on t^i_{sched}
9:     Evaluation of $t^i_{sched'}$; $\mathcal{D}' = \text{f}(t^i_{sched'})$; $\delta = \text{f}(t^i_{sched'}) - \text{f}(t^i_{sched})$
10:    **if** $\text{f}(t'_{osi}) < \text{f}(t^i_{sched})$ **then**

11:      $t^i_{sched} = t^i_{sched'}$
12:    **else**

13:      **if** $\delta >= 0$ and u$< \exp((\text{f}(t^i_{sched'}) - \text{f}(t^i_{sched}))/\mathcal{T})$ **then**

14:        $t^i_{sched} = t^i_{sched'}$
15:      **end if**
16:    **end if**
17:    n = n+1
18:   **end while**
19:   $\mathcal{T}$ reduction and update t^i_{sched} at each reduction
20: **end while**

(ii) Genetic algorithm (GA) [33] is identified as a method mainly used to solve optimisation problems based on a natural selection process similar to biological evolution. GA finds an optimal operative time from our Algorithm 1 ($\mathcal{DCF}$) for the $\mathcal{A}_i$ variables. As explained in Algorithm 4, GA can find a solution beginning with random population of points. GA repeatedly modifies a population of individual solutions. At each step, GA produces a next generation population based on a randomly selection of individuals from the current population. After that, the population turns into an optimal solution. The evaluation number is increased when the method finishes by calculating one generation $\mathcal{P}$. Each generation is a feasible solution for the appliance scheduling (t^i_{sched} per appliance). In the evaluation stage, the best solution $t^i_{sched'}$, which has the lowest demand, is inserted to the best solution set. Mutation and crossover operators are selected to generate the next evaluation from the current generation. The mutation operator randomly shifts the scheduled start times of some appliances in order to generate newly solutions that may have a better result in demand efficiency. They are screened with the constraints to filter out the infeasible ones. The crossover driver swaps scheduled t^i_{sched} under feasible solutions.

Algorithm 4 Optimisation based on GA algorithm

 1: Generate Solutions. Build a set of PopSize $\mathcal{P}$ solution
 2: Reformulation of solutions. Selection of a local search method to each solution in $\mathcal{P}$
 3: **while** number of evaluations $<$ MaxEval **do**
 4:　　t^i_{sched} introduction to P. Evaluation of solution in $\mathcal{P}$ and update
 5:　　Probability of survival based on the quality of the solution
 6:　　$\mathcal{P}$ solution is partially selected to apply the mutation and crossover operation
 7:　　Number of evaluation ++
 8:　　Constraint validate $\mathcal{P}$ for each t^i_{sched}. Discard solutions which are disqualified
 9: **end while**

(iii) Pattern Search (PS) [34] polls the values around the current point and determines the direction that will minimise our Algorithm 1 ($\mathcal{DCF}$) starting at an initial operation time t^i_{sched}. For each possible direction, an all linear combination of the current position is created, and each pattern is multiplied by the size of the mesh to obtain a new one. As presented in Algorithm 5, PS investigates nearest neighbourhood of a possible solution always in the range of lower and upper bounds $[t^i_{beg}, t^i_{end}]$ for each appliance. This solution seeks to find a better one. A failure improvement generation by neighbours ($\mathcal{L}$ and $\mathcal{D}$) would reduce the search step (Δ). Search finishes when the step gets sufficiently short, ensuring the convergence to a local minimal overconsumption.

Algorithm 5 Optimisation based on PS algorithm

 1: Initialise predefine default search step Δ_0; t^i_{sched} and $\Delta=\Delta_0$
 2: **while** Termination condition not reached **do**
 3:　　init current solution $\mathcal{D}= (t^i_{sched}+\mathcal{L}*\Delta)$
 4:　　Evaluate nearest neighbours in $\mathcal{D}$
 5:　　**if** betters in $\mathcal{D}$ **then**
 6:　　　　Update the current solution to the best neighbour in $\mathcal{D}$; $\Delta=\Delta_0$
 7:　　**else**
 8:　　　　Search step reduction $\Delta=\Delta_0/2$
 9:　　**end if**
10: **end while**

(iv) Particle Swarm Optimisation (PSO) [35] is a stochastic search method and simulates the social behaviour of particles used to find parameters that minimise a given objective. The optimisation determines the minimum value and the best location evaluating our Algorithm 1 ($\mathcal{DCF}$) through iterations.

Algorithm 6 illustrates this search procedure, which is initialised with the generation of particles assigning initial velocities and positions. The operative appliances time is defined as a set of lower and upper bounds $[\overrightarrow{t_{beg}}, \overrightarrow{t_{end}}]$, where the solution is found in operation time range $\overrightarrow{t_{beg}} = (t^{i1}_{beg}, ..., t^{ij}_{beg})$, and $\overrightarrow{t_{et}} = (t^{i1}_{end}, ..., t^{ij}_{end})$. The vectors $\overrightarrow{x} = (x^{i1}, ..., x^{ij})$ and $\overrightarrow{v} = (v^{i1}, ..., v^{ij})$ are the current position and velocity, respectively. Each individual adjusts its position according to a linear combination of its inertia ω, the best location of individual particle $\overrightarrow{p} = (p^{i1}, ..., p^{ij})$ and the best location of particle swarm $\overrightarrow{g} = (g^{i1}, ..., g^{ij})$. The confidence degree is determined by the random operators ϕ_p and ϕ_g in the range [0,1] together with the confidence coefficients c_p and c_g. They are responsible for moving in the direction of the best position of a particle and the global best position. The new displacements are no more than one way of trying to imitate other individuals. It then iteratively updates the solution positions (the new location is the old one plus the velocity, modified to keep particles within $[\overrightarrow{t_{beg}}, \overrightarrow{t_{end}}]$, velocities and neighbours). The solution, above $\mathcal{A}_i$, tries to find the optimal ones. After several iterations, particles converge to the best solution.

Algorithm 6 Optimisation based on PSO algorithm

1: Initialise population of particles with random values positions in the search space $\vec{x} \sim U[\overrightarrow{t_{beg}}, \overrightarrow{t_{end}}]$
2: Set each particle best known position to its initial position $\vec{p} \leftarrow \vec{x}$
3: Initialise each particle velocity to random values $\vec{v} \sim U[-\vec{d}, \vec{d}]$ where $\vec{d} = \overrightarrow{beg} - -\overrightarrow{end}$
4: Initialise the best known position $\vec{g}$ to the $\vec{x}$ where $f(\vec{x})$ is lowest
5: **while** Termination condition not reached **do**

6:　　**for** Each particle i **do**

7:　　　　**if** $i > 1$ **then**

8:　　　　　　Choose two random numbers ϕ_p, ϕ_g
9:　　　　　　Adapt velocity $\vec{v} \leftarrow \omega \vec{v} + c_p \phi_p (\vec{p} - \vec{x}) + c_g \phi_g (\vec{g} - \vec{x})$
10:　　　　　　Bound $\vec{v}$ for all dimensions i ($\vec{v}$, -$\vec{d}$, $\vec{d}$)
11:　　　　　　Update the position of the particle $\vec{x} \leftarrow \vec{x} + \vec{v}$
12:　　　　　　Bound population x_i for all dimensions i ($\vec{x}, \overrightarrow{t_{beg}}, \overrightarrow{t_{end}}$)
13:　　　　**end if**
14:　　　　**if** $f(\vec{x}) < f(\vec{p})$ **then**

15:　　　　　　update the particle's best known position $\vec{p} \leftarrow \vec{x}$
16:　　　　**end if**
17:　　　　**if** $f(\vec{x}) < f(\vec{g})$ **then**

18:　　　　　　update the particle's best known position $\vec{g} \leftarrow \vec{x}$
19:　　　　**end if**
20:　　**end for**
21:　　$\vec{g}$ holds the best found position in search space
22: **end while**

4.2. Performance Analysis

Simulation runs on a computer with the following specifications: CPU: 2.3 GHz Intel Core i5; Memory: 8 GB 2133 MHz LPDDR3 and MATLAB R2018b [36]. We evaluate the computational cost of the proposed Algorithm 1 on a series of experiments that represent a variety of possible scenarios of community sizes, consumption patterns or consumer behaviour as depicted in Table 3. Experimentation will help us to identify the most influential factor/s in the computation of the community scheduling.

We have conducted an analysis on the most common appliances' real consumption estimation from manufacturers and data sources from the authors of [37], the U.S. Department of Energy (http://www.energy.gov/), the National Grid report (http://www.nationalgrid.com), the authors of [38] and the reports (https://standby.lbl.gov/docs) as well as the manufacturer data to set the scenarios. Our benchmark is depicted in Table 2. Scenarios were envisioned from the design of a residential building as in Figure 5. In particular, we have generated eight scenarios as illustrated in Table 4 and conducted hundreds of experiments for the different factor values to obtain results on a boxplot shape. On the one hand, we can denote as altruistic or flexible a consumer whose time preferences range is big (e.g., from 0h to 23 h); such types of communities are represented by Cases 1, 3, 5 and 7. On the other hand, Cases 2, 4, 6 and 8 illustrate communities on a more selfish setting. Duration is set equally in both situations.

Table 2. Common household appliance energy use.

ID	Appliance	Model	Watts (W)	Efficiency Ranges European Union A, A^+, A^{++}, A^{+++}	Estimated Average Power in 24 h (kWh)	Estimated Standby Power in 24 h (kWh)	Estimated Operative Time in 24 h (h)
AP1	Water Heater	Wesen ECO30	2000	A++	10–14.73	0.010	1–15
AP2	Clothes Dryer	Balay 3SB285B	4350	A+	1–2.22	0.015	1–10
AP3	Clothes Washer	Eutrotech 1106	1800	A++	1–2.67	0.015	0.5–10
AP4	Iron	Rowenta DX1411	2100	A+	0.1–3	0.002	1–3
AP5	Air conditioner	Fujitsu STG34KMTA	9400	-	3.9–24.3	0.015	0.3–15
AP6	Room air conditioner	Rinnai RPC26WA	2600	-	8–24.3	0.015	3–18
AP7	Heater	DeLonghi HSX3324FTS	2400		1–7	0.08	0.1–10
AP8	Fan heater	Dyson AM09	2000	-	1–6.7	0.015	0.1–10
AP9	Dehumidifier	DeLonghi DEX	210	A++	4–24.3	0.005	1.1–9
AP10	Electric blanket	Medisana HDW	120	-	1–3	0.08	1.2–9
AP11	Ceiling Fans	Westinghouse Bendan	80	A	0.5–9	0.01	0.5–5
AP12	Attic Fans	Remigton	500	-	4.73–6	0.01	0.1–18
AP13	Tower Fan	Sunbeam FA7250	40	-	1.4–3	0.03	0.1–18
AP14	Hoover	BGLS4TURBO	750	-	3–6	0.02	0.3–18
AP15	Boiler	Greenstar Ri	9000	A	8–22	0.05	0.1–3
AP16	Coffee maker	DeLonghi ECOV	1100	A+	9–12	0.05	0.1–3
AP17	Refrigerator	Bosch KDN46VI20	500	A+	8.77–10	0.05	4.77–24
AP18	Dishwasher	Bosch SMS88TI36E	1500	A+++	0.5–1.5	0.015	0.3–4
AP19	Food processor	Becken BFP-400	110	A+	0.5–2	0.015	0.1–5
AP20	Freezer	Bosch GSN36BI3P	350	A++	6–8	0.009	0.1–24
AP21	Microwave	Balay 3CG5172N0	1700	A+	0.9–3	0.01	0.1–4
AP22	Oven	Bosch VBD5780S0	5000	A+	10.96–12	0.01	0.1–8
AP23	Toaster	Russell Hobbs 21973	1100	A+	0.2–1	0.01	0.1–1
AP32	Lighting	Osram	100	-	0.7–3	0.01	0.1–24
AP25	Vaporizer	Philips GC362/80	400	A+++	0.3–2	0.07	0.1–8
AP26	Printer	HP Officejet 3833	100	-	0.8–1	0.05	0.1–4
AP27	Computer	Samsung ls24a450	350	A++	0.7–15.3	0.05	0.1–24
AP26	TV	Panasonic TX43E302B	54	A++	0.1–100	0.05	0.1–24
AP29	Kettle	Philips HD4644/00	3000	A+	6–19	0.01	0.1–1
AP30	Security Alarm	Vbestlife	20	-	0.6-1	0.02	0.1–24
AP31	Auto Cook	MUC88B68ES	1200	A++	1–3	0.09	0.1–3
AP32	Air Cleaner	Balay 3BC598GN	150	A+	1.1–6	0.01	0.1–6
AP33	Vacuum Cleaner	Hoover TH31HO01	1000	A++	0.9–3	0.06	0.2–4
AP34	Electric Fryer	DeLonghi F26237	1800	-	13–16	0.05	0.2–3
AP35	LedTV	LG 49LJ515V	250	A++	1.9–5	0.05	0.2–24
AP36	Electric Store	Dura Heat EUH4000	4000	-	2.4–4	0.05	0.3–23
AP37	Speaker	Logitech Z120	180	A+++	0.3–4	0.01	0.2–20
AP38	Hair Dryer	Rowenta CV3812F0	2100	A+++	0.99–4	0.01	0.2–6
AP39	Smart Camera	Yi Home	4	-	0.99–2	0.01	0.2–24
AP40	Monitor Sensor	iHome	5	-	0.99–10	0.01	0.1–24

Table 3. List of factors for the different case scenarios.

Factor	Type	Value
Community Size	High, Low	30, 5
N. of Appliances	High, Low	1200, 40
Distribution of Appliances	Same, Different	S, D
Fixed Demand	High, Low	Not influenced by optimisation
Variable Demand	High, Low	Up to 18 kWh[6], Up to 9 kWh[6]
Consumer Flexibility	High, Low	24 h, $\mathcal{A}$ duration: $\mathcal{L}$
Vector of $\mathcal{RW}$	Even, Uneven	10 kWh, [10 kWh–20 kWh] 50% SD

Table 4. Possible load-shape situations.

	Community Size $\mathcal{N}$	N. of Appliances $\mathcal{A}$	Distribution of Appliances	Fixed Demand $f\mathcal{D}$ (kWh)	Variable Demand $v\mathcal{D}$ (kWh)	Consumer Flexibility $\mathcal{CF}$	RW Vector per Hour (kWh)
Case 1	From 5 to 30	From 40 to 1200	S	Up to 0.43	Up to 9	24 h	10
Case 2	From 5 to 30	From 40 to 1200	S	Up to 0.43	Up to 9	$\mathcal{L}$	10
Case 3	From 5 to 30	From 40 to 1200	D	Up to 0.43	Up to 9	24 h	10
Case 4	From 5 to 30	From 40 to 1200	D	Up to 0.43	Up to 9	$\mathcal{L}$	10
Case 5	From 5 to 30	From 40 to 1200	S	Up to 0.43	Up to 18	24 h	10
Case 6	From 5 to 30	From 40 to 1200	S	Up to 0.43	Up to 18	$\mathcal{L}$	10
Case 7	From 5 to 30	From 40 to 1200	D	Up to 0.43	Up to 18	24 h	10
Case 8	From 5 to 30	From 40 to 1200	D	Up to 0.43	Up to 18	$\mathcal{L}$	10

We will compare our method's performance with the four different heuristics mentioned in Section 4.1, i.e., SA, PSO, GA and PS, and evaluate the efficiency of the different strategies presented in Section 3.3 on the search for the optimisation objective.

Figures 6–9 depict the scheduling cost for the different case scenarios of consumers using strategy C picking-sequence. These scenarios represent extreme conditions either considering high number of appliances and/or an uneven distribution of them, and also the flexibility of the consumers' time preferences. Communities with selfish settings or fixed consumption display the best results over all different optimisation procedures (cases 2, 4, 6 and 8).
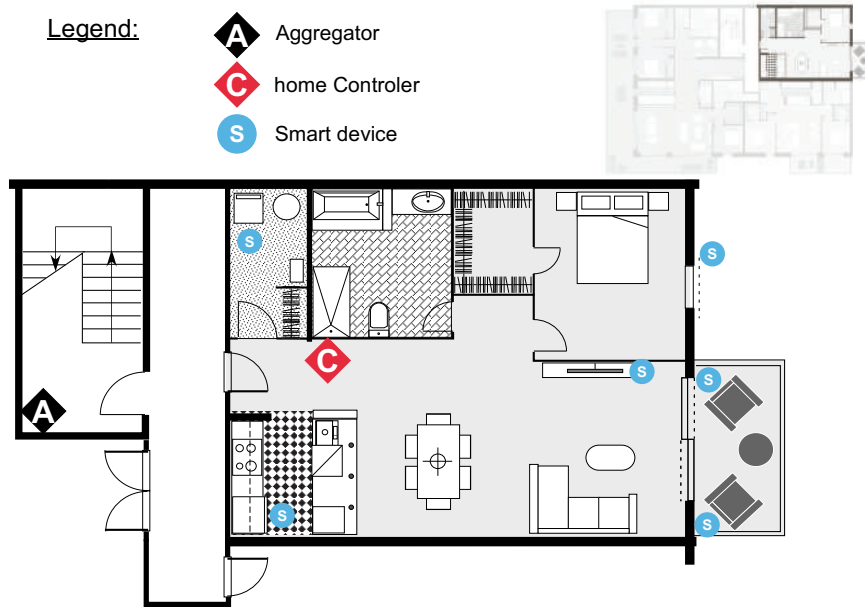
Figure 5. Distribution of the appliances, consumers and aggregator in the community.
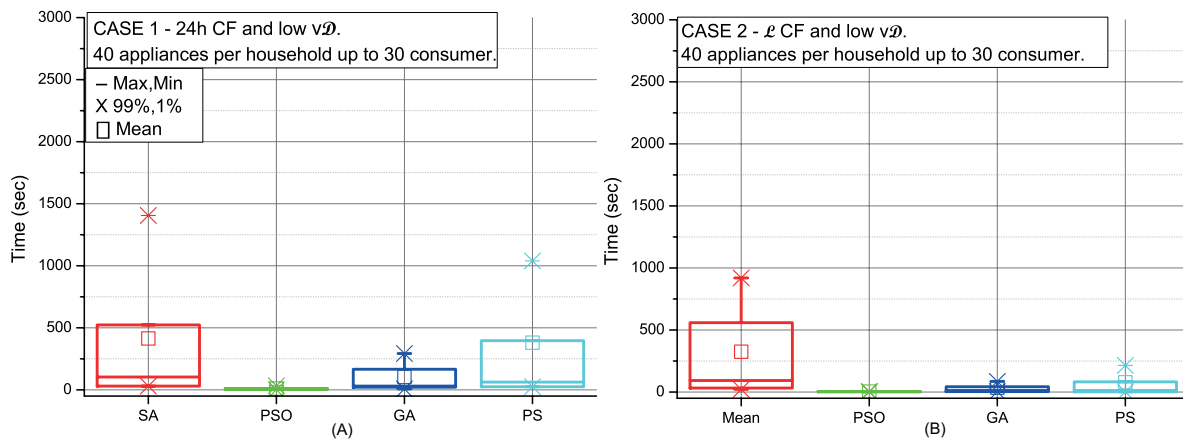
Figure 6. Comparison of the SA, PSO, GA and PS methods for low variable and high fixed consumption in 24 h CF (**A**) and $\mathcal{L}$ CF (**B**).
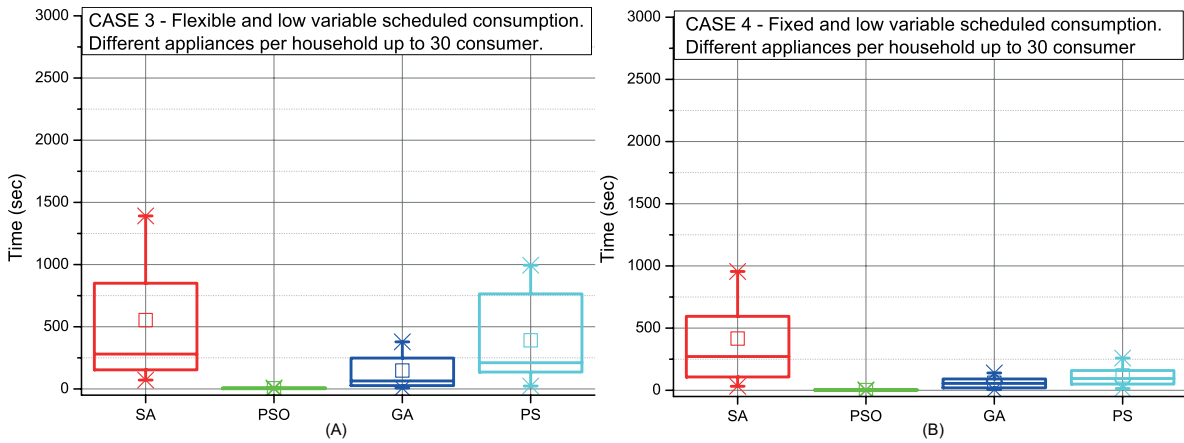
Figure 7. Comparison of the SA, PSO, GA and PS methods for low variable and high fixed consumption in 24 h CF (**A**) and $\mathcal{L}$ CF (**B**).
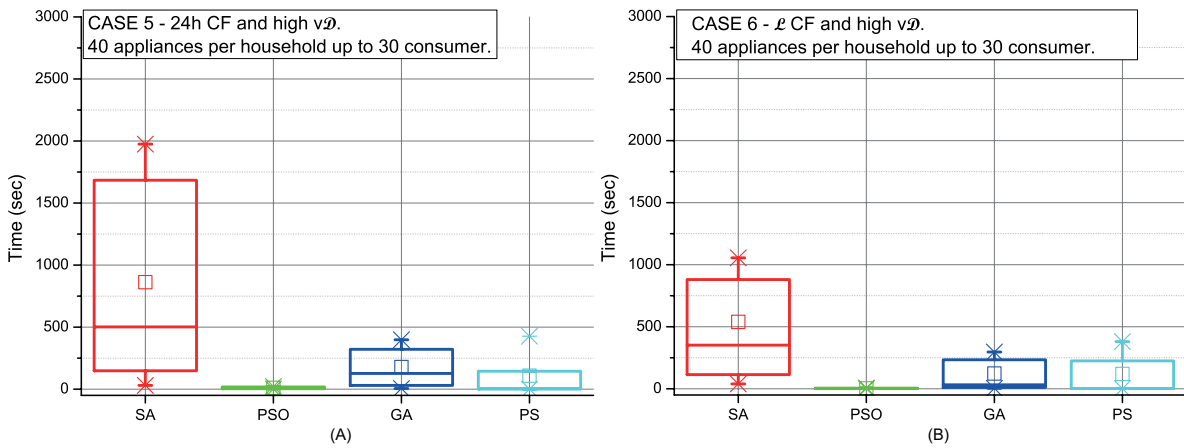

Figure 8. Comparison of the SA, PSO, GA and PS methods for both high variable and fixed consumption in 24 h CF (**A**) and $\mathcal{L}$ CF (**B**).
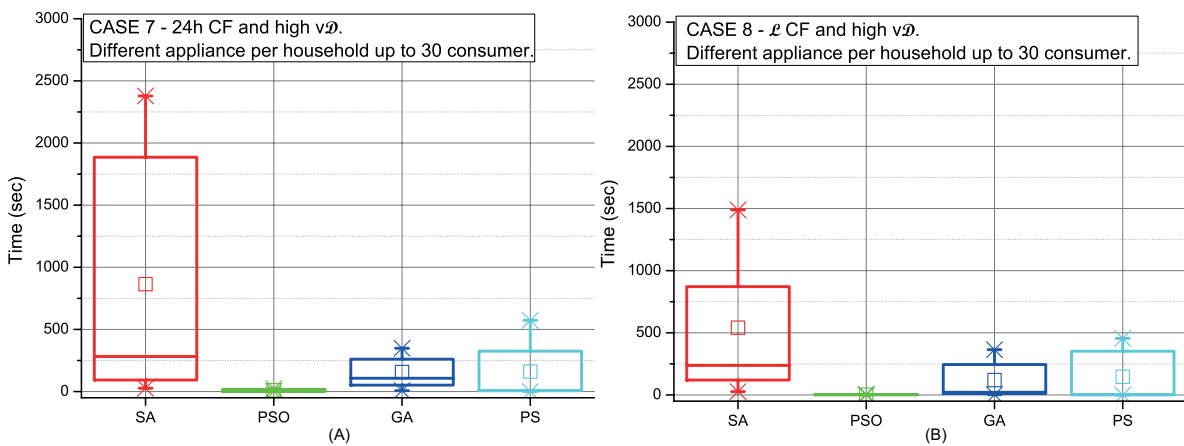

Figure 9. Comparison of the SA, PSO, GA and PS methods for both high variable and fixed consumption in 24 h CF (**A**) and $\mathcal{L}$ CF (**B**).

The CC is higher when consumers have an uneven number of appliances. This effect can be observed in Figure 7 and in comparison with Figure 6. The same occurs in scenarios with high variable demand (see Figure 9 and Figure 8). In addition, a high variable demand (Figures 8 and 9) penalises the CC when compared with low settings (Figures 6 and 7). We find the worst case situation on

altruistic communities with high variable demand when applying strategy C under SA optimisation. As Figure 9 (red colour) shows, it takes 30 minutes. Both PSO and GA work with sets of solutions that interact between themselves. Both perform better due to the number of solutions managed at the same time. We can also conclude that strategy C under all possible scenarios can be solved within the next 24 h, being PSO the most computationally efficient for scheduling (28sec).

Additional simulation measures the performance of communities of 20 appliances per consumer in Case 1. Figure 10 (left) compares all the algorithms and shows that PSO achieves a global optimum solution quickly. GA obtains similar outcomes. Applying SA, Figure 10 (right) shows that the cost needed for the scheduling increases linearly with the number of appliances.
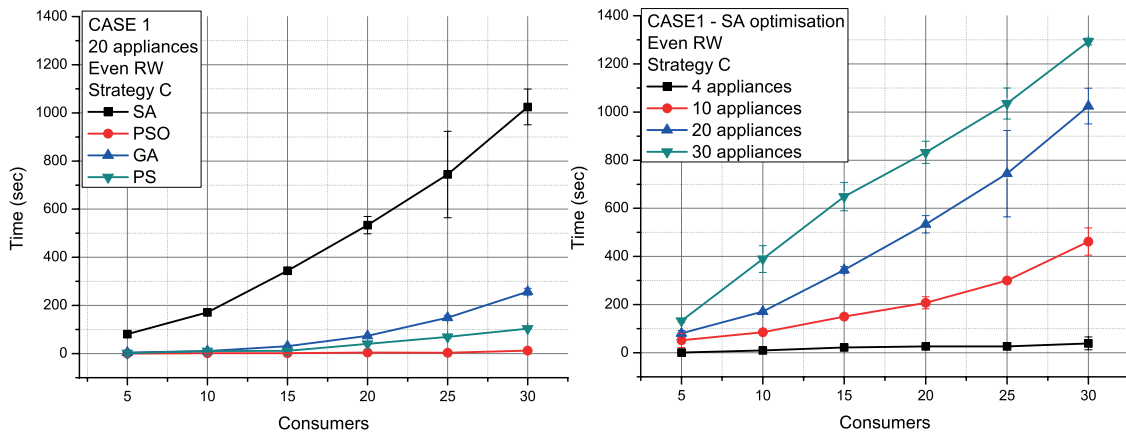

Figure 10. Different approaches (**left**) and appliances number results (**right**).

So far, experiments have been mainly focused on strategy C. Figure 11 depicts the optimised cost obtained after applying all strategies, and taking into account the different factors (see Table 4). These factors are differentiated by branches and data are expressed as a percentage of the required CC.
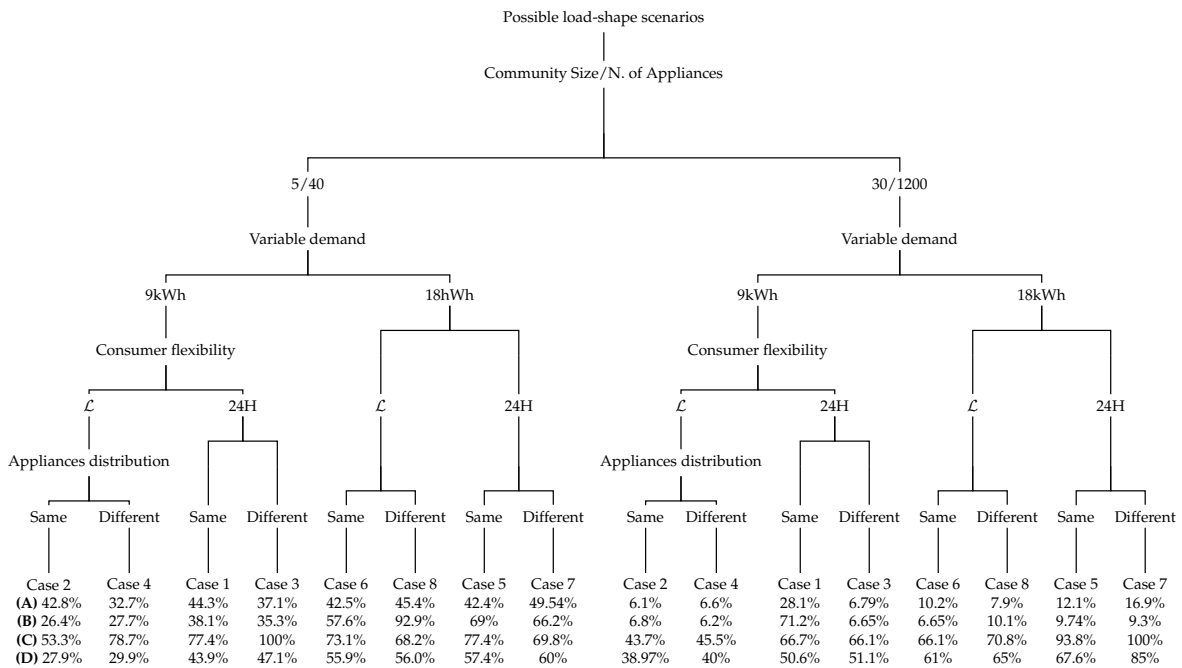

	Case 2	Case 4	Case 1	Case 3	Case 6	Case 8	Case 5	Case 7	Case 2	Case 4	Case 1	Case 3	Case 6	Case 8	Case 5	Case 7
	Same	Different	Same	Different	Same	Different	Same	Different	Same	Different	Same	Different	Same	Different	Same	Different
(A)	42.8%	32.7%	44.3%	37.1%	42.5%	45.4%	42.4%	49.54%	6.1%	6.6%	28.1%	6.79%	10.2%	7.9%	12.1%	16.9%
(B)	26.4%	27.7%	38.1%	35.3%	57.6%	92.9%	69%	66.2%	6.8%	6.2%	71.2%	6.65%	6.65%	10.1%	9.74%	9.3%
(C)	53.3%	78.7%	77.4%	100%	73.1%	68.2%	77.4%	69.8%	43.7%	45.5%	66.7%	66.1%	66.1%	70.8%	93.8%	100%
(D)	27.9%	29.9%	43.9%	47.1%	55.9%	56.0%	57.4%	60%	38.97%	40%	50.6%	51.1%	61%	65%	67.6%	85%

Figure 11. CC results (in %) after applying different strategies (Section 3.3) performed under SA: Round-Robin (**A**), Randomly Round-Robin (**B**), "having the first consumer always the same" (**C**) and randomly (**D**).

Strategy C performs badly, with higher CC in all circumstances. This is mainly because the *Aggregator* needs more resources when it has to optimise all consumers and their appliances all together. The simulation performed with consumers by adding their preferences randomly (strategy D) shows similar cost when compared to strategy C. By contrast, the *Aggregator* under strategy A optimises consumers preferences consecutively in an individual way. A new variant of RR is to perform this strategy when the first consumer starts randomly (strategy B). Both strategies act equally, though dispersed. Strategy A appears as the most appropriate strategy on our system.

Further conclusions can be extracted globally for all the strategies. The CC is higher in four possible situations: when users set a very flexible demand need, when the community is large, when the number of appliances is also large and when a high demand is needed (Figure 11, last branches). A better performance is achieved under strategy B when consumers demand low variable load, in a selfish and small community (Figure 11, first branch). The distribution of appliances also impacts the CC, being higher in large communities with uneven number of appliances per neighbour. The highest CC, which exceeds the half an hour of computation, is obtained in strategy C scenario under a high demand flexibility for an optimisation of 30 neighbours with different appliances distribution per dwelling (Figure 11, last branch).

Finally, Figure 12 compares the CC considering two different $\mathcal{RW}$ vector structure provided from the available sources at the *Utility*: uniform $\mathcal{RW}$ vector and the 50% standard deviation of $\mathcal{RW}$ values. For the eight different cases, and using strategy B and SA, testing is performed for communities of 5–30 consumers. In terms of the chosen strategy, both situations display similar behaviours.
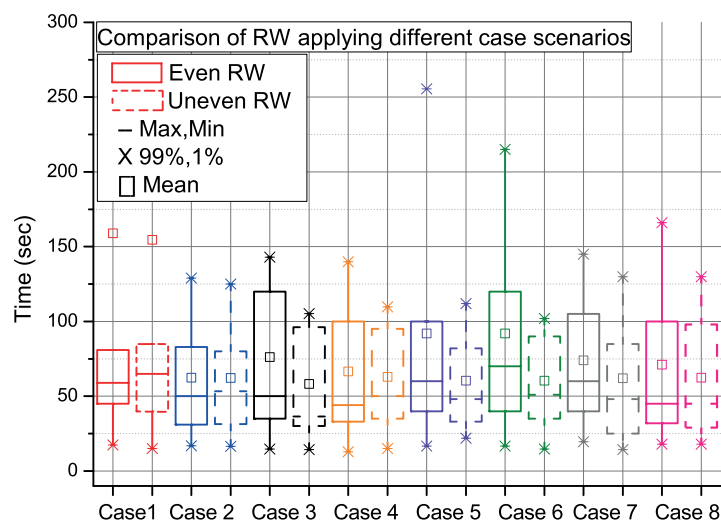

Figure 12. Comparison of RW factor from cases 1 to 8 by applying strategy A under SA.

5. Technical Considerations: Communication, Security and Hardware

Discussion on the development of a pilot testbed for our system over the existent smart home technologies, their security properties and more feasible communication protocols are included in this section. Extensive work on networking infrastructures has been proposed for smart metering data transmission [14,39]. Some approaches focus on fiber-optics for a high-speed data exchange transmission [40], whose deployment cost would only worth when high data transmission rates are required. Power Line Carrier (PLC) is generally applied to computer networks, wired smart meters among other purposes such as remote monitoring and direct control applications offered by utility companies to consumers [41,42]. Note that regulation should be taken into account to allow the use PLC technologies in outdoor deployments as discussed in the work by the authors of [43].

Infrastructures in a Home Area Network (HAN) comprise the communication technologies for deploying HEMS integrating the household appliances. Communication protocols for data

transmission between appliances can be provided with a variety of unwired techniques [28] such as (1) *ZigBee*, which offers an adequate range communication with low data rate and power consumption; (2) *Z-Wave*, which has been used for short range communication due to the low latency communication of small data packets in scalable environments; (3) *6LoWPAN*, which can be applied to building automation designs [44] and to home automation architectures [45]; (4) *Bluetooth*, which is widely used to exchange data over short distances in low energy usage and fast data exchange [46]; and (5) *GSM* networks and *WLAN*, which provide low latency robust communications [47].

Neighbourhood Area Network (NAN) connects customers' HEMS on a two-way communication infrastructure responsible for transmitting their demands and time preferences to the *Aggregator*, as well as the traditional control messages and power grid sensing data. Wireless cellular is widely used in this type of scenarios as described in the work by the authors of [14].

A Wide Area Network (WAN) establishes communication between the *Aggregator* and the Utility substations. Distance to cover is in a radius of a thousand meters comprising of two interconnected networks [48]. Protocols *LoRaWan* and *5G* demonstrate high speed, bandwidth and responsiveness while operating on various licensed and unlicensed frequency bands. Moreover, *LPWAN (LoRa)* will fulfil most of the IoT challenges and applications. By contrast, the introduction of *5G* into the IoT world is still slow and other technologies sound more promising at present time. Table 5 summarises the main features of the discussed technologies and includes recommendations on more appropriate application areas.

In terms of security and privacy, HEMS involve the deployment of physical controls, cyber-security countermeasures as well as privacy leakage prevention [49]. In addition, a gateway architecture for high system availability is proposed in the work by the authors of [50]. Anonymous authentication applying zero-knowledge proof of knowledge could be the solution to provide anonymous authentication between consumers and *Aggregator*. The latter needs to guarantee compliance with the General Data Protection Regulation (GDPR). Furthermore, a methodology to assess the security risks within the HAN domain should be developed as in the work by the authors of [51]. Further details can be found in the work by the authors of [52], where the authors explain the different IoT security threats scenarios (e.g., personal information leak) and provide an evaluation method within a situational smart home framework.

Table 6 identifies the most promising hardware platforms to build our HEMS emphasising low-cost, compatibility, scalability, easy programming and lightweight properties [48,53]. Raspberry Pi 3 [54] is a single-board computer with integrated Bluetooth and WiFi module and enough resources to control the smart appliances and send/receive our system messages. The emergence of cheap microcontrollers like the Arduino has enabled the implementation of low-cost HEMSs mainly devoted to obtain the consumption data as to generate demand/load profiles [55]. For example, the work by the authors of [56] designs and implements a remotely controlled, energy-efficient and highly scalable HEMS using Zigbee in Arduino Mega board as a central controller. In [57] it is discussed and evaluated the performance of *BeagleBone blue* for HEMS developments, an open-source hardware platform with similar principles of Raspberry Pi. Similarly, the proposal by the authors of [58] develops a remote monitoring system using "Libelium Waspmot", a modular device that allows the integration of different sensors and radio transceivers. Additionally, deep learning implementations on Field Programmable Gate Array (FPGA) performs fast due to the exploration of parallel computing [59]. Particularly in the work by the authors of [60], Zedboard implementation (Zynq-ARM Cortex-A9 processor) allows the control of unpredictable loads in a deterministic demand management model. In the work by the authors of [61], the algorithm is modelled in Verilog language on a FPGA allowing dynamic reconfiguration of the HAN. A HEMS prototype is developed on a Cubietruck board (Linux based cortex A7 processor) using a WiFi module [62]. It transmits real-time sensing data using TCP/IP communication protocols.

Table 5. Wireless networks.

Technology	Standard	Data Rate	Frequency Band	Power Consumption	Complexity Transmission Range	Strengths	Application Areas	Encryption/Authentication
Bluetooth	IEEE802.15.1	24 Mbps (v3.0)	2.4 GHz	Low	10 m typical	Small networks Security, speed Easy access Flexibility	HAN	Challenge response scheme/CRC32
WiFi	EEE802.11x	11,54 to 300 Mbps outdoor	2.4 GHz 5 GHz	Very high	Up to100 m	Popular in HAN Speed, flexibility	HAN	4-Way handshake/ CRC32
Z-Wave	802.11	100Kbps	2.4GHz 868.42 MHz (EU)	Low	30 m indoor; 100 m outdoor	No interferences	HAN, NAN	AES128/ 32bit home I.D
Zigbee	IEEEE802.15.4	256 Kbps	2.4 GHz	Very low	10–100 m	Low cost Low consume Flexible topology	HAN,NAN	ENC-MIC-128 Encrypted key/ CRC16
LPWAN	SigFox LoRaWAN NB-IoT	0.3 to 50 kbit/s per channel	915 MHz	Low	10 km in rural settings	Low power Low cost	NAN,WAN	Symmetric key cryptography/AES 128b
6LoWPAN	IEEEE802.15.4	250 Kbps	2.4 GHz	Low	Up to 200 m	Low energy use	HAN, NAN	Symmetric key cryptography/AES 128b
GSM/GPRS	ETSI GSM EN 301349 EN 301347	14.4 Kbps (GSM) 114 Kbps (GPRS)	935 MHz Europe 1800 MHz	Low	Several Km	Low cost Signal quality	HAN, NAN WAN	64 bit A5/1 encryption/ Session key generation
WLAN	IEEE 802.11	150 Mbps	2.4 GHz Europe	Low	250m	Robustness	HAN, WAN	WEP, WPA, WPA2/ Open, Shared EAP
5G	5G Tech Tracker	Up to 20 Gbps	3400-3800 MHz awarding trial licenses (EU)	Very Low	46 m indoor; 92m outdoor	High speed Low latency	HAN, WAN	Symmetric key encryption/ Mobility management entity
3G	UMTS	Up to 14.4 Mbps	450,800 MHz 1.9 GHz	Low	Up to 100 m	Fast Data Transfer	HAN,WAN	CDMA2000/ Authentication and Key Agreement

Table 6. Hardware platforms.

Hardware	Features	Communication Transceivers	Operating System	Power Consumption	Strengths/Weakness
Raspberry Pi 3	1.2 GHz Quad Core BCM2837 64bit CPU 1GB	4 USB, Wi-Fi, Bluetooth, optional ZigBee and Z-Wave	Raspbian Ubuntu Windows 10	1.8 W	Open source platform; Use Python or C++; Cost: 50€
Arduino	32 MHz Micro controller based on ATmega2560 32 kB	WiFi, Bluetooth, ZigBee, GSM	Processing-based	0.2W	Open source platform hardware/software; High flexibility. Cost: 30€; Appliances compatibility
BeagleBone	720 MHz MR Cortex-A8 processor 512 MB	1 USB port, PLC, Bluetooth, Ethernet	Angstrom Linux	1 W	Open source platform similar to Raspberry; Easy setting up; Cost: 90€
RADXA	ROCK Pi 4 is a Rockchip RK3399 based SBC six core ARM processor, 1GB	WiFi, Bluetooth 5.0, USB Port, GbE LAN	Linux	2.3 W	Open source platform; High flexibility; Cost:50€
Libelium Waspmote	14.7 MHz ATmega1281 28 kB	1USB, 802.15.4/ZigBee LoRaWAN,WiFi PRO GSM/GPRS,4G modules	Linux	2 W	High flexibility; Starter kit:200€; ZigBee,WiFi and LoRaWAN support
Xilinx Spartan FPGA	16 Mb SPI flash memory, 100 MHz	Ethernet, USB port	Linux	2 W	SH, Deep Learning, Autonomous System
PYNQ	Embedded systems Xilinx Zynq Systems on Chips (SoCs)	Bluetooth, Ethernet, USB port	Linux	2.3 W	IoT hardware development in Python
Control4Home Automation	Control4Home owners enjoy personalised smart living experiences	Bluetooth, WiFi Z-Wave and ZigBee	Licensed	-	Operation with internet connection; Not user installation
Nexia	Smart home automation system	Z-Wave	Licensed	-	No knowledge of installation required/ Only Z-Wave support; Low compatibility
LG smart appliance	Control key features on LG smart appliances from your smartphone	WiFi	Licensed	-	No knowledge of installation required/ Only for LG appliances; Closed source

In terms of our DR system, and from preliminary design decisions, our prototype will consist of a Raspberry testbed as the main processor of the HEMS, as it offers good computing performance at a very low price. Its interoperability will provide the performance of alternatives protocols such as ZigBee, WiFi or Z-wave. In the proposed architecture, the WiFi wireless communication between the Aggregator, HEMS and appliances can transfer the data at around a hundredth millisecond, a suitable speed for our current application. The proposed centralised DR system (*Aggregator*) will also operate in an open-source HW platform.

6. Conclusions

Globally, smart communities are envisioned more efficient as residents gain autonomy and self-organisation for reducing and shifting any resource consumption. Strategies for energy demand response applied to smart residential communities can lead to improved scenarios of energy efficiency. Consumers have the opportunity to reduce their electricity cost and/or peak-to-average ratio through scheduling their power consumption. In this article, we have described a DR model that integrates the electricity supply available from renewable energy sources into the scheduling process, which is centralised via the community *Aggregator*. We have showed details of community scheduling algorithm implementation and evaluated it in terms of its computational cost. Empirical comparison of our algorithm design on different implementation strategies for player turn selection and optimisation heuristics as well as on a series of case scenarios of community's consumption patterns showed feasible results in all cases (less than 1 minute to compute the rescheduled community vector). Simulations are conducted with data from our own benchmark of appliance power cost. We have also illustrated development decisions of a mobile app for consumers introducing their demands and time preferences. Finally, we included the discussion of the preliminary decisions on the hardware requirements and communication protocols for a pilot deployment. Immediate future work includes the pilot deployment comprising the algorithm/Aggregator running on the most suitable HW platform as well as the home controllers that autonomously activate/deactivate the smart appliances. We also plan to refine the scheduling algorithm as to consider the usage or purpose of the consumption along with the device type. Furthermore, Utilities and Aggregators in possession of the real-time data from microgeneration and other energy harvesting generators would enhance the conceptual demand response model. Finally, a study of the community patterns will be conducted through game theory and evolutionary computation methods.

Abbreviations

The following abbreviations are used in this manuscript.

$\mathcal{N}$	Consumer number
$\mathcal{A}_i$	Appliance number
i	Consumer identifier
j	Appliance index
a_{ij}	Consumer i's appliance identifier
t	Certain time
$\mathcal{RW}$	24-hour supply vector from renewables
t_{beg}	Earliest start time appliance
t_{end}	Latest final time appliance
t_{sched}	Scheduled start time of appliance
$\mathcal{D}$	Consumer demand
$v\mathcal{D}$	Variable demand
$f\mathcal{D}$	Fixed demand
CF	Consumer Flexibility
$\mathcal{L}$	Duration of the planned operation of appliance a_{ij} in the next day
SH	Smart Home
$HEMS$	Home Energy Manager System
HAN	Home Arena Network
NAN	Neighbour Area Network
WAN	Wide Area Network
IoT	Internet Of Things
ICT	Information and Communication Technologies
SG	Smart Grid
DSM	Demand System Manager
$MILP$	Mixed Integer Linear Programming
SA	Simulates Annealing
PSO	Particle Search Optimisation
GA	Genetics Algorithm
PS	Pattern Search
RR	Round-Robin
PLC	Power Line Carries
CC	Computational Cost

References

1. Steg, L. Promoting household energy conservation. *Energy Policy* **2008**, *36*, 4449–4453. [CrossRef]
2. Lui, T.J.; Stirling, W.; Marcy, H.O. Get Smart. *IEEE Power Energy Mag.* **2010**, *8*, 66–78. [CrossRef]
3. Gubbi, J.; Buyya, R.; Marusic, S.; Palaniswami, M. Internet of Things (IoT): A Vision, Architectural Elements, and Future Directions. *Future Gener. Comput. Syst.* **2013**, *29*, 1645–1660. [CrossRef]
4. Schaffers, H.; Komninos, N.; Pallot, M.; Trousse, B.; Nilsson, M.; Oliveira, A. Smart Cities and the Future Internet: Towards Cooperation Frameworks for Open Innovation. In *The Future Internet*; Lecture Notes in Computer Science; Springer: Berlin, Germany 2011; Volume 6656, pp. 431–446.
5. Palensky, P.; Dietrich, D. Demand Side Management: Demand Response, Intelligent Energy Systems, and Smart Loads. *IEEE Trans. Ind. Inform.* **2011**, *7*, 381–388. [CrossRef]
6. Babayo, A.A.; Anisi, M.H.; Ali, I. A Review on energy management schemes in energy harvesting wireless sensor networks. *Renew. Sustain. Energy Rev.* **2017**, *76*, 1176–1184. [CrossRef]
7. Yang, Z.; Zhou, S.; Zu, J.; Inman, D. High-Performance Piezoelectric Energy Harvesters and Their Applications. *Joule* **2018**, *2*, 642 – 697. [CrossRef]
8. Palomar, E.; Liu, Z.; Bowen, J.P.; Zhang, Y.; Maharjan, S. Component-based modelling for sustainable and scalable smart meter networks. In Proceedings of IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks, Sydney, Australia, 19 June 2014; pp. 1–6. [CrossRef]

9. Palomar, E.; Chen, X.; Liu, Z.; Maharjan, S.; Bowen, J. Component-Based Modelling for Scalable Smart City Systems Interoperability: A Case Study on Integrating Energy Demand Response Systems. *Sensors* **2016**, *16*, 1810. [CrossRef] [PubMed]

10. Ponds, K.T.; Arefi, A.; Sayigh, A.; Ledwich, G. Aggregator of Demand Response for Renewable Integration and Customer Engagement: Strengths, Weaknesses, Opportunities, and Threats. *Energies* **2018**, *11*, 2391. [CrossRef]

11. Jacobsen, R.H.; Gabioud, D.; Basso, G.; Alet, P.; Azar, A.G.; Ebeid, E.S.M. SEMIAH: An Aggregator Framework for European Demand Response Programs. In Proceedings of the 2015 Euromicro Conference on Digital System Design, Madeira, Portugal 26–28 August 2015; pp. 470–477. [CrossRef]

12. Correa-Florez, C.A.; Michiorri, A.; Kariniotakis, G. Comparative Analysis of Adjustable Robust Optimization Alternatives for the Participation of Aggregated Residential Prosumers in Electricity Markets. *Energies* **2019**, *12*, 1019. [CrossRef]

13. Lin, Y.H.; Hu, Y.C. Residential Consumer-Centric Demand-Side Management Based on Energy Disaggregation-Piloting Constrained Swarm Intelligence: Towards Edge Computing. *Sensors* **2018**, *18*, 1365. [CrossRef]

14. Shareef, H.; Ahmed, M.; Mohamed, A.; Al Hassan, E. Review on Home Energy Management System Considering Demand Responses, Smart Technologies, and Intelligent Controllers. *IEEE Access* **2018**, *6*, 24498–24509. [CrossRef]

15. Qayyum, F.; Naeem, M.; Khwaja, A.S.; Anpalagan, A.; Guan, L.; Venkatesh, B. Appliance scheduling optimization in smart home networks. *IEEE Access* **2015**, *3*, 2176–2190. [CrossRef]

16. Agnetis, A.; De Pascale, G.; Detti, P.; Vicino, A. Load scheduling for household energy consumption optimization. *IEEE Trans. Smart Grid* **2013**, *4*, 2364–2373. [CrossRef]

17. Sarris, T.; Messini, G.; Hatziargyriou, N. Residential demand response with low cost smart load controllers. In Proceedings of the Mediterranean Conference on Power Generation, Transmission, Distribution and Energy Conversion (MedPower 2016), Belgrade, Serbia, 6–9 November 2016; pp. 1–8. [CrossRef]

18. Logenthiran, T.; Srinivasan, D.; Phyu, E. Particle swarm optimization for demand side management in smart grid. In Proceedings of the 2015 IEEE Innovative Smart Grid Technologies–Asia (ISGT ASIA), Bangkok, Thailand, 3–6 November 2015; pp. 1–6. [CrossRef]

19. Fadlullah, Z.M.; Quan, D.M.; Kato, N.; Stojmenovic, I. GTES: An optimized game-theoretic demand-side management scheme for smart grid. *IEEE Syst. J.* **2014**, *8*, 588–597. [CrossRef]

20. Poolla, C.; Ishihara, A.K.; Milito, R. Designing near-optimal policies for energy management in a stochastic environment. *Appl. Energy* **2019**, *242*, 1725–1737. [CrossRef]

21. Iqbal, Z.; Javaid, N.; Mohsin, S.; Akber, S.; Afzal, M.; Ishmanov, F. Performance analysis of hybridization of heuristic techniques for residential load scheduling. *Energies* **2018**, *11*, 2861. [CrossRef]

22. Mahapatra, C.; Moharana, A.K.; Leung, V.C.M. Energy Management in Smart Cities Based on Internet of Things: Peak Demand Reduction and Energy Savings. *Sensors* **2017**, *17*, 2812. [CrossRef] [PubMed]

23. Mohsenian-Rad, A.; Wong, V.W.S.; Jatskevich, J.; Schober, R. Optimal and autonomous incentive-based energy consumption scheduling algorithm for smart grid. In Proceedings of the 2010 Innovative Smart Grid Technologies (ISGT), Gaithersburg, MA, USA, 19–21 January 2010, pp. 1–6. [CrossRef]

24. Alshehri, K.; Liu, J.; Chen, X.; Bacsar, T. Privacy-Preserving Multi-Period Demand Response: A Game Theoretic Approach. *arXiv* **2017**, arXiv:1710.00145 .

25. Kelly, F.P.; Maulloo, A.K.; Tan, D.K.H. Rate Control for Communication Networks: Shadow Prices, Proportional Fairness and Stability. *J. Oper. Res. Soc.* **1998**, *49*, 237–252. [CrossRef]

26. Yaïche, H.; Mazumdar, R.R.; Rosenberg, C. A Game Theoretic Framework for Bandwidth Allocation and Pricing in Broadband Networks. *IEEE/ACM Trans. Netw.* **2000**, *8*, 667–678. [CrossRef]

27. Fan, Z. A Distributed Demand Response Algorithm and its Application to PHEV Charging in Smart Grids. *IEEE Trans. Smart Grid* **2012**, *3*, 1280–1290. [CrossRef]

28. Zhou, B.; Li, W.; Chan, K.W.; Cao, Y.; Kuang, Y.; Liu, X.; Wang, X. Smart home energy management systems: Concept, configurations, and scheduling strategies. *Renew. Sustain. Energy Rev.* **2016**, *61*, 30–40. [CrossRef]

29. Myerson, R.B. Conference structures and fair allocation rules. *Int. J. Game Theory* **1980**, *9*, 169–182. [CrossRef]

30. Schwarz, D. *Jump Start Adobe XD*, 1 ed.; Sitepoint: Melbourne, Australia, 2017.

31. Bouveret, S.; Lang, J. Manipulating picking sequences. *ECAI* **2014**, *14*, 141–146.

32. Nahar, S.; Sahni, S.; Shragowitz, E. Simulated Annealing and Combinatorial Optimization. In Proceedings of the 23rd ACM/IEEE Design Automation Conference, Las Vegas, NV, USA, 29 June–2 July 1986; pp. 293– 299. [CrossRef]

33. Goldberg, D.E. *Genetic Algorithms in Search, Optimization and Machine Learning*, 1st ed.; Addison-Wesley Longman Publishing Co., Inc.: Boston, MA, USA, 1989.

34. Abramson, M. Pattern Search Algorithms for Mixed Variable General Constrained Optimization Problems. Ph.D. Thesis, Rice University, Houston, TX, USA, 2003.

35. Kennedy, J.; Eberhart, R. Particle swarm optimization. In Proceedings of the ICNN'95–International Conference on Neural Networks, Perth, WA, Australia, 27 November–1 December 1995; Volume 4, pp. 1942–1948. [CrossRef]

36. *MATLAB.* version 9.5.0 (R2018b); The MathWorks Inc.: Natick, MA, USA, 2010.

37. Laicane, I.; Blumberga, D.; Blumberga, A.; Rosa, M. Evaluation of Household Electricity Savings. Analysis of Household Electricity Demand Profile and User Activities. *Energy Procedia* **2015**, *72*, 285–292. [CrossRef]

38. Ross, J.; Meier, A.K. Whole-House Measurements of Standby Power Consumption. In *Energy Efficiency in Household Appliances and Lighting*; Technical Report; Springer: Berlin, Germany, 2000; pp. 278–285.

39. Andreadou, N.; Guardiola, M.O.; Fulli, G. Telecommunication Technologies for Smart Grid Projects with Focus on Smart Metering Applications. *Energies* **2016**, *9*, 375. [CrossRef]

40. Shakerighadi, B.; Anvari-Moghaddam, A.; Vasquez, J.C.; Guerrero, J.M. Internet of Things for Modern Energy Systems: State-of-the-Art, Challenges, and Open Issues. *Energies* **2018**, *11*, 1252. [CrossRef]

41. Pitì, A.; Verticale, G.; Rottondi, C.; Capone, A.; Lo Schiavo, L. The Role of Smart Meters in Enabling Real-Time Energy Services for Households: The Italian Case. *Energies* **2017**, *10*, 199. [CrossRef]

42. Fang, X.; Wang, N.; Gulliver, T.A. A PLC channel model for home area networks. *Energies* **2018**, *11*, 3344. [CrossRef]

43. Galli, S.; Scaglione, A.; Wang, Z. Power Line Communications and the Smart Grid. In Proceedings of the Power Line Communications and the Smart Grid, Gaithersburg, MD, USA, 4–6 October 2010; pp. 303–308. [CrossRef]

44. Han, S.; H. Cao, Q.; Alinia, B.; Crespi, N. Design, Implementation, and Evaluation of 6LoWPAN for Home and Building Automation in the Internet of Things. In Proceedings of the IEEE/ACS 12th International Conference of Computer Systems and Applications (AICCSA), Marrakech, Morocco, 17–20 November 2015. [CrossRef]

45. Aradindh, J.; Srevarshan, V.B.; Kishore, R.; Amirthavalli, R. Home automation in IOT using 6LOWPAN. *Int. J. Adv. Comput. Eng. Netw.* **2017**, *5*, 26–28.

46. Collotta, M.; Pau, G. A Solution Based on Bluetooth Low Energy for Smart Home Energy Management. *Energies* **2015**, *8*, 11916–11938. [CrossRef]

47. Wen, M.H.; Leung, K.C.; Li, V.O.; He, X.; Kuo, C.C.J. A survey on smart grid communication system. *APSIPA Trans. Signal Inf. Process.* **2015**, *4*, e5. [CrossRef]

48. Saleem, Y.; Crespi, N.; Rehmani, M.H.; Copeland, R. Internet of things-aided smart grid: technologies, architectures, applications, prototypes, and future research directions. *IEEE Access* **2019**, *7*, 62962–63003. [CrossRef]

49. El-hajj, M.; Fadlallah, A.; Maroun, C.; Serhrouchni, A. A Survey of Internet of Things (IoT) Authentication Schemes. *Sensors* **2019**, *19*, 1141. [CrossRef] [PubMed]

50. Lin, H.; Bergmann, N.W. IoT Privacy and Security Challenges for Smart Home Environments. *Information* **2016**, *7*, 44. [CrossRef]

51. Ali, B.; Awad, A. Cyber and Physical Security Vulnerability Assessment for IoT-Based Smart Homes. *Sensors* **2018**, *18*, 817. [CrossRef] [PubMed]

52. Park, M.; Oh, H.; Lee, K. Security Risk Measurement for Information Leakage in IoT-Based Smart Homes from a Situational Awareness Perspective. *Sensors* **2019**, *19*, 2148. [CrossRef] [PubMed]

53. Froiz-Míguez, I.; Fernández-Caramés, T.; Fraga-Lamas, P.; Castedo, L. Design, Implementation and Practical Evaluation of an IoT Home Automation System for Fog Computing Applications Based on MQTT and ZigBee-WiFi Sensor Nodes. *Sensors* **2018**, *18*, 2660. [CrossRef] [PubMed]

54. Qureshi, M.U.; Girault, A.; Mauger, M.; Grijalva, S. Implementation of home energy management system with optimal load scheduling based on real-time electricity pricing models. In Proceedings of the IEEE 7th International Conference on Consumer Electronics—Berlin (ICCE-Berlin), Berlin, Germany, 3–6 September 2017; pp. 134–139. [CrossRef]

55. Amer, M.; M El-Zonkoly, A.; Aziz, N.; M'Sirdi, N. Smart Home Energy Management System for Peak Average Ratio Reduction. *Ann. Univ. Craiova.* **2014**, *38*, 180–188.

56. Baraka, K.; Ghobril, M.; Malek, S.; Kanj, R.; Kayssi, A. Low Cost Arduino/Android-Based Energy-Efficient Home Automation System with Smart Task Scheduling. In Proceedings of the 2013 Fifth International Conference on Computational Intelligence, Communication Systems and Networks, IEEE Computer Society, Washington, DC, USA, 5–7 June 2013; pp. 296–301. [CrossRef]

57. Nayyar, A.; Puri, V. A Review of Beaglebone Smart Board's-A Linux/Android Powered Low Cost Development Platform Based on ARM Technology. In Proceedings of the 9th International Conference on Future Generation Communication and Networking (FGCN), Jeju Island, Korea, 25–28 November 2015; pp. 55–63.

58. Quintana-Suárez, M.A.; Sánchez-Rodríguez, D.; Alonso-González, I.; Alonso-Hernández, J.B. A Low Cost Wireless Acoustic Sensor for Ambient Assisted Living Systems. *Appl. Sci.* **2017**, *7*, 877. [CrossRef]

59. Alhafidh, B.M.H.; Daood, A.I.; Alawad, M.M.; Allen, W. FPGA Hardware Implementation of Smart Home Autonomous System Based on Deep Learning. In Proceedings of the International Conference on Internet of Things, Santa Barbara, CA, USA, 15–18 October 2018; Springer: Berlin, Germany 2018; pp. 121–133.

60. Khoury, J.; Mbayed, R.; Salloum, G.; Monmasson, E. Design and implementation of a real time demand side management under intermittent primary energy source conditions with a PV-battery backup system. *Energy Build.* **2016**, *133*, 122–130. [CrossRef]

61. Bazydło, G.; Wermiński, S. Demand side management through home area network systems. *Int. J. Electr. Power Energy Syst.* **2018**, *97*, 174–185. [CrossRef]

62. Vivek, G.V.; Sunil, M.P. Enabling IOT services using WIFI–ZigBee gateway for a home automation system. In Proceedings of the IEEE International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN), Kolkata, India, 20–22 November 2015, pp. 77–80. [CrossRef]