HEMS: A Home Energy Market Simulator

Andrea Monacchi, Sergii Zhevzhyk, Wilfried Elmenreich
Institute of Networked and Embedded Systems / Lakeside Labs,
Alpen-Adria-Universität Klagenfurt
9020 Klagenfurt, Austria
{*name.surname*}@aau.at

***Abstract*—Stability issues in the electric power grid originate from the rising of renewable energy generation and the increasing number of electric vehicles. The uncertainty and the distributed nature of generation and consumption demand for optimal allocation of energy resources, which, in the absence of sufficient control reserve for power generation, can be achieved using demand-response. A price signal can be exploited to reflect the availability of energy. In this paper, market-based energy allocation solutions for small energy grids are discussed and implemented in a simulator, which is released for open use. Artificial neural network controllers for energy prosumers can be designed to minimize individual and overall running costs. This enables a better use of local energy production from renewable sources, while considering residents' necessities to minimize discomfort.**

***Keywords*—Market-based allocation, demand response, home energy management, smart home, smart appliance**

I. Introduction

The rise in renewable energy installations and the increasingly diffusion of electric vehicles is destabilizing the supply and demand of energy in the grid. To compensate for this problem, a modern energy infrastructure equipped with a bidirectional communication infrastructure will soon allow for the implementation of demand-side management (DSM), i.e., the possibility for utilities to control the energy required by consumers in order to balance supply and demand. Direct load control is based on the remote control of selected loads in households and industries. Indirect load control exploits a price signal to reflect fluctuations in the availability of energy in the grid. Pricing schemes encourage the operation of devices when energy cost is lower, fostering awareness to increase conservation and efficiency [1] or to automatically schedule the operation of selected devices to off-peak periods [2]. Many existing studies deal with energy management by finding allocations given the early and truthful revelation of the agents' preferences. However, the energy management process typically has to deal with decentralised resources controlled by self-interested agents. Accordingly, a centralized scheduler would not be able to optimally cope with the local perception and individual preferences of loads, which becomes relevant especially in multi-user environments. Electronic markets provide a framework for allocation of limited resources within communities of distributed agents [3]. As opposed to classic scheduling based on optimization, agents can keep their preferences private and act based on their local view of the environment. Auctions can efficiently regulate the access to a shared resource demanded by competing agents, using a price to balance demand and supply in the system. The energy price increases when demand exceeds supply and decreases otherwise. Dynamic pricing allows for adaptive control of distributed resources, in a way that is optimal locally (according to the individual utility of agents) and globally (in terms of social welfare), and leading to the emergence of global coordination of autonomous controllers.

Auctions can be generally distinguished into single-sided and double-sided auctions, according to the number of agents allowed to submit offers per each side [4]. In this study we focus on double auctions, as in presence of a small number of traders they provide a more balanced trading environment than monopolistic and monopsonistic ones. In particular, continuous-double auctions were shown leading to high market efficiency and quick convergence to the theoretical equilibrium even with a few traders [5]. In spite of the limited number of energy producers in dwellings, the benefits of using a double auction for energy management becomes more straightforward in rural scenarios and in presence of storage units (e.g., batteries and electrical vehicles). Groups of buildings can share a power and data infrastructure to manage the overall community, given multiple families and businesses, with different needs and expectations. Towards this vision, Alam et al. [6] showed energy exchange leading to an improved system efficiency and to the 65% reduction of storage use. Ygge and Akkermans [7] present an early work in scheduling household appliances using computational markets. In PowerMatcher [8] device agents representing supply and demand are organized in a hierarchy, so that the whole network can be seen as a virtual prosumer (i.e., device and power plant). Auctions are held between siblings, whilst the cluster supernode concentrates offers from the lower level into one that is delivered to the upper level. The root coincides with the auctioneer agent, which manages the price formation for the whole tree. A tree-based topology of computational markets can be easily associated to the underlying power distribution network, and it allows for reducing the complexity of the auction. Although the attempt to create an energy market targeted at both, load-level and the whole grid, there has been little research into designing market policies considering inhabitant's behavior and preferences to minimize the discomfort yielded by control strategies. DRSim [9] exploits real-world datasets to extract inhabitants' models describing consumption attitude and activities, related to contextual factors such as weather. Residents are described by a time-dependent price sensitivity, an energy perception sensitivity representing user's energy awareness, as well as a DR communication sensitivity, that is, the responsiveness to price changes.

Given such a background, the design of controllers for energy producing and consuming appliances becomes increasingly important. While trading agents for double auctions have already received considerable attention, control strategies need to consider system-level effects, which go beyond the pure utility an individual agent assigns to a resource or task. Besides aiming at the minimization of operational costs and a better use of local resources, appliances should also consider effects on users and other appliances, such as discomfort and fairness. Moreover, cooperative behaviors might be able to achieve a task with better performance or lower cost than pure competitive market-oriented mechanisms. We address this demand for self-organization by using ANN-based controllers which are trained using evolutionary methods, based on the methodology presented in [10]. The main contribution of this paper is the introduction of a novel framework for enabling household appliances and local generators to strategically decide on trading energy across the building and with the power distribution network. After reviewing existing auction types and trading strategies we present a user-driven efficiency measure extending existing market efficiency measures. In Section III, we demonstrate an electronic market implementation for small, local energy grids. A discussion of such approaches led to the implementation of a plugin for the FREVO evolutionary computing framework [11], which we release for open use[1]. The environment allows researchers for experimenting self-organization in technical applications [12], and in particular, we expect our plug-in to provide a validation testbed for designers of energy management policies.

II. Double-sided auctions

Market-based allocation defines the dynamics under which a set of agents A interact for the exchange of money and goods. Computational markets allow for decentralised control of resources, such as allocation and scheduling, in which self-interested agents can make effective decisions based only on local information. Thus, markets represent a scalable solution, requiring minimal communication and computational effort [3]. A double-sided auction allows for the many-to-many trade of goods, between a seller $s_i \in S$ and a buyer $b_j \in B$ respectively advertising an ASK and a BID offer. Each offer consists of a quantity $q \in \mathbb{N}^+$ of the traded resource and an offered unit price $p \in \mathbb{R}_{>=0}$. Every trader has a private value ψ for the good, denoting the worth for the agent based on which a rational agent constrains its offers to avoid losses. To prevent unreasonably high BID offers due to competitive and possibly irrational behavior, the unit price is bounded to a limit price p_{max} the market can accept. The market keeps an order book O of all submitted offers and provides traders with the current minimal (or outstanding) ask a_{min} and the current maximum (or outstanding) bid b_{max}, as well as the price history for old transactions. The shared values a_{min} and b_{max} determine the market status, and are initialized to $a_{min} = p_{max}$ and $b_{max} = 0$ at the beginning of each trading day. A trading day is the period during which traders submit offers to the market and for which unmatched offers persist in the orderbook. The trading day takes place in trading rounds or iterations, in which offers are submitted and ordered according to their price (DESC for ASK offers and ASC for BID offers) and arrival time. At each iteration, traders can submit new offers to the market. The New York stock exchange policy (also called spread reduction rule) constraints the market so that only offers that improve the current market status can be accepted, which is, only a BID $b > b_{max}$ and an ASK $a < a_{min}$ can be accepted. A match occurs when there exists a bid b_j whose unit price $p_{b_j} >= p_{a_i}, \forall b_j, a_i \in O$. The k-pricing is commonly employed to compute a price between the bid and ask offer prices so that the surplus can be distributed among the participants. Specifically, given the ASK $a_i = \langle q_i, p_i \rangle$ and the BID $b_j = \langle q_j, p_j \rangle$, a transaction θ occurs if and only if $p_j >= p_i$ and the price $p_\theta = k \cdot p_j + (1-k) \cdot p_i$, with $k \in [0,1]$. This allows for controlling the assignment of the surplus to the traders, for instance having it equally distributed when $k = 0.5$. Depending on the auction type, matched offers can be either committed to a transaction or queued until the end of the trading day. The discrete-time double auction, also called clearing house (CH) or synchronous, takes place over a time interval in which traders can place offers, whilst transactions are created at the end of the trading period. For buyers, the mechanism implies waiting until the end of the period before receiving traded resources. Therefore, non-delayable resources are not suitable for such a kind of market. In the NOBEL project a discrete-time double auction (DDA) is used to implement a district energy market [13], in which traders are required to predict their electricity demand/supply every 15 minutes. The continuous-time double auction (CDA) is a discriminative price double-auction, meaning that the market can clear offers each time a transaction is possible. A day-ahead CDA was used in [14] to balance supply and demand in a power network. Although the CDA leads to a higher number of matched offers, it yields high volatility in the market price, which can create dissatisfaction among traders. To overcome this issue, the stable continuous double auction (SCDA) implements a price adjustment mechanism [15]. The uniform-price double auction (UPDA) combines advantages of both CDA and DTDA to provide a non-discriminative price while maximizing the number of matched offers as in the CDA. Accordingly, offers are continuously matched, although traders have to wait the end of the trading period to get the transaction price and obtain the traded resource. Market-based allocation mechanisms are commonly evaluated under the market allocation efficiency. Decision policies are considered efficient when they maximize social welfare, which is the sum of utilities delivered to traders in a certain outcome. Therefore, efficiency is maximized when all the possible profit is extracted from the traders that operate in it. This is strictly related to Pareto optimality, as a market is efficient when no agent can get a better condition without worsening someone else. In effect, as utility can be exchanged between agents through payments the two measures coincide [4]. The efficiency can be computed as the ratio between the actual surplus of all traders pr^a and the maximum possible surplus pr^e that would be obtained in a centralized and optimum allocation.

$$\epsilon = \frac{pr^a}{pr^e} = \frac{pr^a_b + pr^a_s}{pr^e_b + pr^e_s} \tag{1}$$

This translates into Eq. 1. The profit pr_b for a buyer j is given by $\sum_{i \in \mathbb{N}} (\psi_{b_j} - p_{ij}) q_{ij}$ with ψ_{b_j} sensitivity price and q_{ij} quantity bought from seller i at the unit price p_{ij}. The profit pr_s for a seller i is given by $\sum_{j \in \mathbb{N}} (p_{ij} - \psi_{s_i}) q_{ij}$ with ψ_{s_i} reservation price and q_{ij} quantity sold to buyer j at the unit price p_{ij}. The actual overall profit pr^a is given by the sum of the actual profits of all buyers and sellers, computed as difference between the agent's private value and the actual

[1] http://frevo.sourceforge.net

unit price paid. The equilibrium profit pr^e is given by the sum of equilibrium profits of all buyers and sellers, computed as difference between the agent's private value and the market equilibrium price p_0. The equilibrium price represents the intersection of the supply and demand curve and indicates the clearing price for the auction. Given the Walrasian tâtonnement for which convergence to the equilibrium can be reached after enough trading rounds even without intervening influences, an approximation of the equilibrium price can be computed using the last transaction price over a Marshallian path [16] (i.e., the sequence of trades over the supply and demand curves). This situation can be computed as the price of an auction in which agents declare their private value for the good being traded [17]. While the market efficiency captures the traders' profit, the user-centric efficiency was proposed to assess responsiveness of services [18]. Each request to operate a device can be associated to a utility function measuring the value associated to such a resource. To avoid delays on responsive devices, the valuation of the device should reflect its immediacy, so that its utility decays to zero over time. For a buyer $b_j \in B$, $V_j(r) = \delta \cdot U_j(r)$, with δ expressing the discomfort received from the delayed allocation over the utility U_j to receive r amount of resources. The user-centric efficiency of a market is then given by the overall value delivered to the users at the allocation time, that is $\sum_{j=1}^{n} V_j$.

III. HEMS: Home Energy Market Simulator

In this section we introduce the HEMS simulator, an environment in which trading agents for energy management can be designed and assessed. The simulator is a plug-in for the FREVO evolutionary computing framework, a software tool for the evolutionary design of distributed agents' behavior.

A. Problem statement

The chosen market implements a uniform-price double auction where the trading day takes place over multiple durationless iterations. We selected an allocation interval of 1 second, as this allows for responsively changing the planned scheduling actions in presence of uncritical environment changes. The trading day is also 1 second long, as this time guarantees minimal waiting time for appliances before they are entitled to run. We assume devices to truthfully report their energy necessities, while rationally aiming at maximizing their profit. This allows us to focus on the allocation strategy, as we do not need to model uncertainty in the forecasted power demand and supply. A balancing mechanism such as the one presented in [14] could be employed to charge and discourage deviations from forecasted demand and production. Transactions between loads and local generators are priced under a k-pricing scheme with $k = 0.5$, whereas transactions involving a grid agent Γ are charged under the given tariffs: the feed-in price p^b_Γ and the energy price p^s_Γ (see III-B). Electrical devices can be distinguished according to the possibility to postpone their operation (i.e., shiftable appliances) and to reduce their consumption level (i.e., curtailable appliances). While we handle deferrable devices in this study we do not exploit curtailable devices. The power of matched offers is allocated at the end of the trading day only if it is enough to operate the target device, or freed otherwise. Partially matched offers resulting from different BID and ASK quantities offered are split into new offers with same unit price. However, Initial supply fragmentation is prevented by matching only indivisible BID offers for the first iterations, with exception of offers made from grid agents. Similarly, all BID offers exceeding available supply are removed at the the end of each iteration, as they could be only partially fulfilled. The overall problem denotes the minimization of the cost to supply each appliance and prevent user discomfort (Equation 2), as well as the maximization of the profit resulting from selling local generation (Equation 3):

$$\underset{\forall b_j \in B}{\text{minimize}} \sum\nolimits_{t \in T} \sum\nolimits_{b_j \in B} p^t_j q^t_j + \sum\nolimits_{b_j \in B} \delta t' \tag{2}$$

$$\underset{\forall s_i \in S}{\text{maximize}} \sum\nolimits_{t \in T} \sum\nolimits_{s_i \in S} p^t_i q^t_i \tag{3}$$

where t is the time, p is the energy price and q the quantity for a specific state, whereas δ is the user discomfort attributed to a t' allocation delay. The optimization problem is also subject to various constraints. The demand of a load depends on the energy bought to be stored $q^{t,s}_j$ and to operate $q^{t,o}_j$, which translates into $q^t_j = q^{t,o}_j + q^{t,s}_j, \forall b_j \in B$. The constraints $p^t_j \leq \psi_{b_j}, \forall b_j \in B$ and $p^t_i \geq \psi_{s_i}, \forall s_i \in S$ prevent agents from trading with losses. Also, $q^t_j - q^t_i \geq 0$, with $b_j = s_i, \forall b_j \in B, \forall s_i \in S$ prevents traders from snatching up all power to sell it back to the grid for an increased price. Constraint $b_\theta \neq s_\theta, \forall \theta \in \Theta$ prevents prosumers from exploiting the difference between ψ_{b_j} and ψ_{s_i} to self-provide energy, such as when closing the connection on the grid agent to exploit incentives on the feed-in tariff.

B. Agent-based modeling

This work assumes the presence of power grid agents, each managing a connection with the main power grid (e.g., smart meters), along with a set of electrical loads and local generators. Each *grid agent* is a truth-telling agent, which simply exposes an *energy tariff* and a *feed-in tariff* modeling the price per kilowatt-hour received for energy fed into the grid. The highest amount of power that can be supplied by or fed into the grid at a certain time is respectively modeled by a *power availability* and a *power capability* function. Smart electrical loads and local generators embed a controller to trade towards profit maximization (See Fig. 1). Each agent

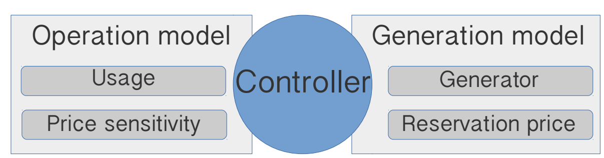

Fig. 1. The agent structure

is described by a name, a credit and its expenses to operate. An *operation* and a *generation model* describe energy demand and production, controlled respectively under a *sensitivity price* and a *reservation price* model. The *price sensitivity* model ψ defines the maximum unit price the residents would be willing to pay to operate the load at a certain time, that is the utility associated to the appliance. Therefore, the price sensitivity can be used to compute the utility delivered to users upon completion of the service operation. For an inflexible operation model, the sensitivity price equals the market limit price. The *reservation* *price* expresses production costs, influenced by technological costs and storage (if employed). Unlike flexible generation models (e.g., battery), an inflexible generation model (e.g., photovoltaic plant) is required to get rid of all produced power. In this way, we can distinguish the inconvenience resulting from device deferral, as this might affect either user comfort or safety-critical applications. Generation models can either import external time series of measured data or calculate the production based on a generation model and a weather model. At the time of writing we support importing of time series of both wind and photovoltaic generation, and, additionally, a model of photovoltaic (PV) plants based on the work presented in [19]. A PV model is specified as the maximum peak power the generator can supply and an efficiency factor that can be used to model aging conditions. The plant is also described through its position in terms of height, latitude and longitude, and size in terms of square meters. Electrical appliances provide multiple services, each described as an operation model. Within an operation model, an *appliance profile* describes the execution of a process on an electrical machinery, that is, the coordinated execution of the system components [20]. A service can be modeled as a sequence of states, where each state σ_i is associated a peak power level $P_i \in \mathbb{N}^+$ in Watts, as well as a duration $d_i \in \mathbb{N}^+$ in seconds. A *device start delay sensitivity* χ^b in seconds models the responsiveness under which appliances are required to start from the time of request. The delay sensitivity is always 0 for inflexible services, while it is necessary to model flexible user-driven devices, such as a coffee machine. Device operations result in a *device begin discomfort* δ_b proportional to the overwaited time between the first offer and the beginning of the operation. A *state start delay sensitivity* χ^s in seconds models the effects resulting from a delayed start of intermediate states. This affects the whole device operation, yielding a *state begin discomfort* δ_s proportional to the overwaited time between the ending of a state and the beginning of the next. For instance, in a coffee machine the water heating state should always be followed by the following states within a critical deadline, as otherwise the whole operation would fail leading to a terrible cold coffee. An *interruption sensitivity* χ^i defines the severity under which device operation can be interrupted, that is, the intra-state delay sensitivity. While certain continuous devices might be interruptible (e.g., electric heaters), device operation should be completed on a state-by-state basis. Device operation is therefore a list of atomic states and handover between devices should be achieved exploiting inter-state delay sensitivity. Although a short interruption (e.g., in the order of seconds) might not greatly affect user's activities, it has severe influence on the overall operation of the device resulting in an *interruption discomfort* δ_i. A *usage model* models the probability to operate an appliance at a specific time of the day. In the simplest setting, we may have a static willingness value $\omega^* \in [0, 1]$ expressing the probability to start at a certain time instant. The value can be defined statically for the whole simulation time or for time intervals (e.g., hourly, quarterly). The probability P_{use} to have a transition from OFF to ON within a time interval of length N is given by $P_{use} = 1 - (P_{hold})^N$, which translates into $P_{use} = 1 - (1 - \omega^*)^N$ and consequently $(1-\omega^*)^N = 1-P_{use}$ and $\omega^* = 1- \sqrt[N]{1 - P_{use}}$. For operations are not independent, we need a willingness decay $\lambda \in \mathbb{R}$ updating the probability to run for the current time interval (e.g., the same hour) as $P_{use} = P_{use}(1 - \lambda)^n$, with $n \in \mathbb{N}$ number of operations for the current interval. For instance given $P_{use} = 0.8$ to use the coffee machine in the interval $9 - 10AM$, $\omega^* = 1 - \sqrt[3600]{0.2} = 0.00045$ is the probability to start at each second. After the first operation of the device, we might use a decay $\lambda = 0.5$ which would produce $P_{use} = 0.4$ and consequently $\omega^* = 1 - \sqrt[3600]{0.6} = 0.00014$ for the remaining time. Nevertheless, this simple usage model does not express contextual factors and can not model situations such as "two washing machine operations in a row might be more common than just one, although three consecutive ones are very unlikely to occur". A more sophisticated usage model based on machine learning techniques could be learned from a consumption dataset. To this purpose, in [21] we show how a Bayesian network can be used to model the usage behavior of devices in an Austrian household.

C. Evolutionary approach

The design of appliance controllers follows the methodology presented in [10]. The authors proposed the application of evolutionary methods to train ANN-based controllers for a team of robotic soccer players. The general idea of evolutionary algorithms (EA) is competing for limited resources [22]. Typically, EAs work with a pool of candidates, each containing a genotype which fully encodes an agent's behavior. The initial population is filled with randomly generated candidates. In order to find an appropriate solution we run our algorithm through the defined number of generations. In each generation we run the simulation of the HEMS for each candidate of the population. The outcome of this simulation is a feedback encoded as a fitness function. Elite candidates, which have the highest fitness values, are kept in the population for the next generation to fix the success behaviors. Additionally to the elite, some candidates are selected randomly, where the likelihood of selection is the higher, the better the fitness rank of a candidate. In order to find better candidates we apply mutation (variation operator) to the elite and selected representations. This means modifying the weights of synaptic connections of the neural network with a random small amount. Mutated candidates are also selected for the next generation. Another variation operator is recombination that is applied to pairs of candidates which are randomly selected from the selected representations. Two parent solutions are combined to produce child representation(s) from them. Mutation, recombination, and selection are used in each generation to drive the evolution.

D. Simulation interface

The HEMS tool acquires a scenario definition written in the Javascript object notation (JSON) format, and consisting in a weather model, connections with the power grid and a model of each trader [23]. A graphical user interface is also provided to display the simulation status and to plot charts of the selected measures. As shown in Fig. 2, the top left panel shows the logical topology of the power provisioning, whereas the top right component lists the balance of each appliance. The market view is meant to display the step by step evolution of the market over the trading day, by showing the state of the orderbook and pending transactions upon reception of offers. At the end of the simulation, a report prompts the simulation result, which includes all components used for the

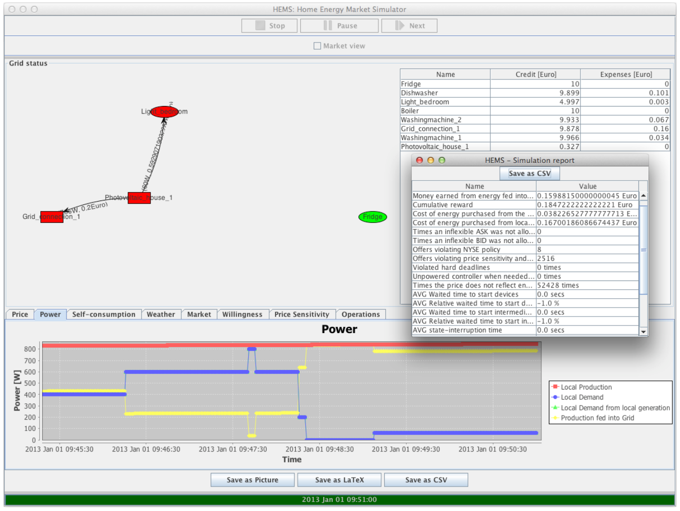

Fig. 2. Graphical user interface of the FREVO plugin

fitness function. In the bottom, tabs report various metrics characterizing the ongoing simulation. Charts and data can also be exported to external files, as a picture, LaTeX TikZ and comma-separated values (CSV) format. The *price* tab displays the average grid energy price and the local energy price. The latter is computed as the average energy price for all running transactions, which is the grid tariff when no energy is traded and it is equal to the feed-in tariff when selling all produced energy to the grid. This denotes the minimum price a device has to pay to start operating. The *power* tab displays aggregate power produced and demanded locally, while the *self-consumption* tab shows the exploitation of local generation. Weather conditions such as the sun factor are displayed in the *weather* tab. The *market* tab displays the market efficiency of the current allocated devices. Since energy bought from the grid is charged under grid tariffs, the efficiency in presence of such transactions is 1 because $pr^a = pr^e$. Consequently, we compute the efficiency only for transactions between local generators and loads. The *willingness* tab shows the trading willingness of agents, i.e., the tendency to buy or sell energy over time. We also show the price sensitivity and the reservation price of all agents in the next tab. Finally, the *operations* tab reports all ran and running states per device, in terms of delays and state interruptions.

IV. Learning device controllers

The design of appliance controllers follows the methodology presented in [10] for the application of evolutionary methods to train artificial neural network (ANN) controllers. In HEMS, a controller for an energy prosumer is implemented as a fully-meshed ANN. It is important to remark that when an appliance is off its starting willingness is given by its usage model, as $\omega = \omega^*$. After making a first BID offer, $\omega = 1.0$ reflects the rationality to complete the operation of the device. This value is kept by the device agent to decide whether to use the controller to formulate new offers. In this way, after a first offer the agent is committed to pursue the allocation of its future offers, so as to complete the operation of the device. As for local generation, it is always used first to satisfy local demand, as given by the operation model. Based on the leftover amount of energy, which is negative when no production is locally available to the agent, a trading tendency $\tau \in [-1.0, 1.0]$ (-1 to sell, 0 to skip the trade and $+1$ to buy) is computed. While the tendency is discrete for pure loads and generators, this value becomes essential for prosumers, such as batteries, where it can be used to reflect the amount of charge. The controller is therefore queried each time the agent needs to buy or sell energy, and it is based on the following structure:

- *seller's inputs* which include the reservation price (ψ_{s_i}/p_{max}), the unit price of the outstanding ASK (a_{min}/p_{max}), the position in the ASK orderbook (with 1.0 denoting first and 0.0 the last), the percentage of already matched ASK offer ($P_i^{reserved}/P_i^{demanded}$);
- *context information* meaning the time the decision is being taken, i.e., the hour (midnight is 0.0, 11 pm is 1.0), month (january is 0.0, december is 1.0) and weekday (sunday is 0.0, weekdays are 0.5, saturday is 1.0);
- *trading tendency* which include the offer importance (1.0 for inflexible and 0.0 for flexible) and trading tendency τ;
- *buyer's inputs* which model the delayed start tolerance left (χ_l^b/χ^b , with χ_l^b initially equal to χ^b and progressively decreased), the price sensitivity (ψ_{b_j}/p_{max}), the unit price of the outstanding bid (b_{max}/p_{max}), the position in the BID orderbook, and the percentage of already matched BID offer.

As noticeable, inputs are provided as relative values. Moreover, the controller outputs a real value between 0 and 1, which is then scaled to $[-p_{max}, +p_{max}]$ using $p = 2 * p_{max} * p_{output} - p_{max}$. A market threshold p^{th} is then used to decide whether to formulate a BID ($p > p^{th}$), ASK ($p < p^{th}$) or an opt out otherwise. The primary goal of the controller is to minimize costs, while selecting a price which rationally reflects the presence of local production or the willingness to start and complete an ongoing task. Since FREVO is using an absolute ranking-based selection, there was no need to normalize fitness to positive values or to squeeze fitness values into a given number range. The formulated fitness function is:

$$F = R + (\delta_g * I_{grid}) - C, \tag{4}$$

The reward R is the sum of the utility delivered to users upon completion of device operation, which is the price sensitivity multiplied to the duration and power of each state described in the device profile. The incoming I_{grid} resulting from feeding energy into the grid is also considered. We then subtract the costs (Eq. 5) weighted through various penalties δ.

$$\begin{aligned} C =& \delta_g * C_{grid} + \delta_b \frac{1}{B_f^o} \sum\nolimits_{b_j \in B_f^o} d_{b_j} \\ & \delta_s \frac{1}{B_f^s} \sum\nolimits_{b_j \in B_f^s} d_{s_j} + \delta_i \frac{1}{B_f^o} \sum\nolimits_{b_j \in B_f^o} d_{c_j} + \\ & \delta_i (\sum_{j=1}^{B} v_{i_j} + \sum_{i=1}^{S} v_{i_i}) + \delta_m (\sum_{j=1}^{B} v_{m_j} + \sum_{i=1}^{S} v_{m_i}) + \\ & \delta_l (\sum_{j=1}^{B} v_{p_j} + \sum_{i=1}^{S} v_{p_i}) + \delta_n (\sum_{j=1}^{B} v_{n_j} + \sum_{i=1}^{S} v_{n_i}). \end{aligned} \tag{5}$$

Costs include: the energy purchased from the main energy grid C_{grid}, the discomfort resulting from user interaction (i.e., average delayed device start d_b, average delayed state start d_s, and average interruption time within states d_c), the cost v_m of violating the NYSE market policy and the cost v_p for trading irrationally (i.e., with losses). For devices might have different tolerances, we normalize each average to the delay tolerance of each device. User discomfort results only from flexible loads B_f, which were operated B_f^o and have more than a state B_f^s. For inflexible services all time-instants are considered in which offers were not allocated, which results in v_i and is penalized through δ_i. We ran 250 generations with a population of 80 candidates, in a scenario with a main big supplier (grid) and a small PV plant. We observed that the fitness tends to saturate around 150 generations. The simulation converged upon a state where the electrical loads learned to operate properly (without interruptions and excessive delay before starting states) and the generators are able to sell all of their produced energy. Nevertheless, the selected market yields suboptimal results in scenarios with multiple small generators, which would require better coordination to fully supply big loads.

V. Conclusions and future work

This paper discussed the market-based allocation of energy resources in small local grids. We described the design and implementation of a simulation tool allowing for modeling trading behavior of loads and local energy generators. The environment is part of the FREVO evolutionary computing framework and can be used to experiment self-organization in demand response scenarios. We are currently working towards the inclusion of external appliance usage models, as identified in [21]. Improvement is targeting various aspects of the tool, such as weather and generation models. The ultimate goal is providing a complete framework for the assessment of energy management policies, targeting both efficiency and the comfort of inhabitants. The tool is expected to foster discussion within the demand response community, by constituting a testbed where different market mechanisms and trading approaches can be compared.

Acknowledgment

Work supported by Lakeside Labs, Klagenfurt, Austria and funded by the European Regional Development Fund (ERDF) and the Carinthian Economic Promotion Fund (KWF) under grant 20214-22935-34445 (Smart Microgrid) and 20214-23743-35469-35470 (MONERGY). We would like to thank P. Grippa, M. Pöchacker and D. Egarter for the feedback.

References

[1] A. Monacchi, W. Elmenreich, S. D'Alessandro, and A. M. Tonello, "Strategies for energy conservation in carinthia and friuli-venezia giulia," in *Proc. of the 39th Annual Conference of the IEEE Industrial Electronics Society*, Vienna, Austria, 2013.

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

[3] S. H. Clearwater, Ed., *Market-Based Control: A Paradigm for Distributed Resource Allocation*. River Edge, NH: World Scientific, 1996.

[4] S. Parsons, J. A. Rodriguez-Aguilar, and M. Klein, "Auctions and bidding: A guide for computer scientists," *ACM Comput. Surv.*, vol. 43, no. 2, pp. 10:1–10:59, Feb. 2011.

[5] V. L. Smith, "An Experimental Study of Competitive Market Behavior," *The Journal of Political Economy*, vol. 70(2), pp. 111–137, April 1962.

[6] M. Alam, S. D. Ramchurn, and A. Rogers, "Cooperative energy exchange for the efficient use of energy and resources in remote communities," in *Proc. of the 2013 International Conference on Autonomous Agents and Multi-agent Systems*, ser. AAMAS '13, 2013, pp. 731–738.

[7] F. Ygge and H. Akkermans, "Power load management as a computational market," in *Second International Conference on Multiagent Systems (ICMAS)*, Kyoto, 1996, pp. 393–400.

[8] K. Kok, B. Roossien, P. MacDougall, O. van Pruissen, G. Venekamp, R. Kamphuis, J. Laarakkers, and C. Warmer, "Dynamic pricing by scalable energy management systems - field experiences and simulation results using powermatcher," in *IEEE Power and Energy Society General Meeting*, July 2012, pp. 1–8.

[9] T. Wijaya, D. Banerjee, T. Ganu, D. Chakraborty, S. Battacharya, T. Papaioannou, D. Seetharam, and K. Aberer, "Drsim: A cyber physical simulator for demand response systems," in *Proc. of the IEEE International Conference on Smart Grid Communications (SmartGridComm)*, Oct 2013, pp. 217–222.

[10] I. Fehérvári and W. Elmenreich, "Evolving neural network controllers for a team of self-organizing robots," *Journal of Robotics*, 2010.

[11] A. Sobe, I. Fehervari, and W. Elmenreich, "FREVO: A tool for evolving and evaluating self-organizing systems," in *Proc. of the IEEE Sixth International Conference on Self-Adaptive and Self-Organizing Systems Workshops (SASOW)*, Sept 2012, pp. 105–110.

[12] W. Elmenreich and H. de Meer, "Self-organizing networked systems for technical applications: A discussion on open issues," in *Proc. of the Third International Workshop on Self-Organizing Systems*, J. S. K.A. Hummel, Ed. Springer Verlag, 2008, pp. 1–9.

[13] D. Ilic, P. Da Silva, S. Karnouskos, and M. Griesemer, "An energy market for trading electricity in smart grid neighbourhoods," in *Proc. of the 6th IEEE International Conference on Digital Ecosystems Technologies (DEST)*, June 2012, pp. 1–6.

[14] P. Vytelingum, S. D. Ramchurn, T. D. Voice, A. Rogers, and N. R. Jennings, "Trading agents for the smart electricity grid," in *Proc. of the 9th International Conference on Autonomous Agents and Multiagent Systems (AAMAS '10)*, 2010, pp. 897–904.

[15] Z. Tan and J. Gurd, "Market-based grid resource allocation using a stable continuous double auction," in *8th IEEE/ACM International Conference on Grid Computing*, Sept 2007, pp. 283–290.

[16] P. Brewer, M. Huang, B. Nelson, and C. Plott, "On the behavioral foundations of the law of supply and demand: Human convergence and robot randomness," *Experimental Economics*, vol. 5, no. 3, pp. 179–208, 2002.

[17] S. Phelps, K. Cai, P. McBurney, J. Niu, S. Parsons, and E. Sklar, "Auctions, evolution, and multi-agent learning," in *Adaptive Agents and Multi-Agent Systems III. Adaptation and Multi-Agent Learning*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2008, vol. 4865, pp. 188–210.

[18] B. N. Chun and D. E. Culler, "User-centric performance analysis of market-based cluster batch schedulers," in *Proc. of the 2Nd IEEE/ACM International Symposium on Cluster Computing and the Grid*, 2002, pp. 30–39.

[19] M. Pöchacker, T. Khatib, and W. Elmenreich, "The microgrid simulation tool RAPSim: description and case study," in *Proc. of the 2014 IEEE Innovative Smart Grid Technologies (ISGT) Conference - Asia*, Kuala Lumpur, Malaysia, May 2014.

[20] D. Egarter, A. Monacchi, T. Khatib, and W. Elmenreich, "Integration of legacy appliances into home energy management systems," *arXiv preprint arXiv:1406.3252*, 2014.

[21] A. Monacchi, D. Egarter, W. Elmenreich, S. D'Alessandro, and A. M. Tonello, "GREEND: an energy consumption dataset of households in Italy and Austria," in *Proc. of IEEE International Conference on Smart Grid Communications (SmartGridComm)*, Venice, Italy, Nov 2014.

[22] A. E. Eiben and J. E. Smith, *Introduction to evolutionary computing*. Springer, 2003.

[23] A. Monacchi, S. Zhevzhyk, and W. Elmenreich, "Home energy market simulator: The scenario definition format," Alpen-Adria Universität Klagenfurt, Technical Report, September 2014.