Paper

An Agent-Based Collaborative Platform for the Optimized Trading of Renewable Energy within a Community

Luca Tasquier and Rocco Aversa

Dipartimento di Ingegneria Industriale e dell'Informazione, Second University of Naples, Aversa, Italy

Abstract—Cities are increasingly recognized for their ability to play a catalytic role in addressing climate and energy challenges using technologically innovative approaches. Since energy used in urban areas accounts for about 40% of total EU energy consumption, a change of direction towards renewable energy is necessary in order to alleviate the usage of carbonized electricity and also to save money. A combination of IT and telecommunication technologies is necessary to enable the energy and resources saving. ICT based solutions can be used to enable energy and money saving not only for a single building, but for the whole community of a neighborhood. In this paper a model for the energy cost minimization of a neighborhood together with an agent-based interaction model that reproduces the proposed formal representation is presented. Furthermore the authors present a prototype implementation of this model and first experimental tests.

Keywords*—*collective intelligence, energy cost minimization, multi-agent systems, Smart Cities.

1. Introduction

The Internet of Things (IoT) paradigm is rapidly gaining ground in the scenario of modern wireless telecommunications. The basic idea of this concept is the pervasive presence around us of a variety of things or objects – such as Radio-Frequency IDentification (RFID) tags, sensors, actuators, mobile phone – which, through unique addressing schemes, are able to interact with each other and cooperate with their neighbors to reach common goals [1]. As application of IoT, Smart Cities mainly focus on applying the next-generation information technology to all walks of life, embedding sensors and equipment to hospitals, power grids, railways, bridges, tunnels, roads, buildings, water systems, dams, oil and gas pipelines and other objects in every corner of the world [2].

The issues related to the climate challenge make Smart Cities even more attractive. Nowadays a change of direction towards renewable energy is necessary in order to alleviate the usage of carbonized electricity and also to save money. One of the renewable energy options is solar electricity, which could be deployed decentralized in urban areas. In Europe, 21.9 GW of photovoltaic systems were connected to the grid in 2011, compared to 13.4 GW in 2010, which is in line with the average of 40% increase during the past 15 years. Under this aspect, ICT based solutions can be used to enable energy and money saving not only for a single building, but for the whole community of a neighborhood. First of all, a formal representation of the problem is needed in order to study the feasibility of a possible solution and to map it on hardware and software structures. This introduces a model for the minimization of energy costs that leads to benefits not only to the single household, but to the entire community of a neighborhood. Each house is represented by an agent acting on its behalf in order to implement the developed model and to automate the optimization operations by using and exchanging the energy within the community according to the house own requirements and capabilities.

The presented work has been conceived within the research activities of CoSSMic project. CoSSMic (Collaborating Smart Solar-powered Micro-grids. FP7 – SMARTCITIES, 2013) is an ICT European project that aims at fostering a higher rate for self-consumption (50%) of decentralized renewable energy production by innovative autonomic systems for the management and control of power micro-grids on users' behalf [16]. Home Area Network (HAN) is formed by all electrical devices of the home connected to the network. In each house there is an agent gateway. Agents are used to manage the HAN with the aim of optimizing self-consumption rates using renewable energy sources. Micro-grids, embedded with renewable energy production, storage capacity and consumption, are combined with an intelligent ICT platform.

The paper is structured as follows. Section 2 reviews some related works, while in Section 3 a formal modeling of the cost minimization problem is presented, as well as an agent interaction model that maps the proposed solution. A prototypal implementation of the agent model and experimental results are described in Section 4. Finally conclusions are drawn in Section 5.

2. Related Work

The scientific community investigates different priorities in the field of smart grids. Some examples are market deregulation, ICT architecture, IT security and data protection, energy efficiency, integration of renewable energies, supply security, grid bottlenecks, grid expansion, decentralized energy production, smart meteorology, storage devices and load flexibilization. Much effort has been spent on the investigation in the field of agents' technology. In [7] the authors describe why they believe that artificial intelligence, and particularly, the fields of autonomous agents and multi-agent systems are essential for delivering the smart grid as it is envisioned. In [8] a multi-agent system architecture simulates and analyses competitive electricity markets combining bilateral trading with power exchange mechanisms. Several heterogeneous and autonomous intelligent agents representing the different independent entities in electricity markets are used and a detailed description of a promising algorithm for decision support is presented and used to improve agents bidding process and counter-proposals definition. Agents are endowed with historical information about the market including past strategies of other players, and have strategic behavior to face the market. In [9] authors consider how consumers might relate to future smart energy grids, and how exploiting software agents to help users in engaging with complex energy infrastructures. Paper [10] presents the architecture of an agent-based platform for power generating and power consuming companies in contract electricity market. An intelligent agent, by using fuzzy logic modification of genetic algorithm in order to accomplish strategy optimization, implements the negotiation process by selecting a strategy using learning algorithms. In [11] another negotiation algorithm using game theory is proposed, where agents act on behalf of end users, thus implying the necessity of being aware of multiple aspects connected to the distribution of electricity related to outside world variables like weather, stock market trends, location of the users etc. In [12] authors define a methodology for predicting the usage of home appliances. An agent based prediction algorithm captures the everyday habits by exploiting their periodic features. In addition, the algorithm uses an episode generation hidden Markov model (EGH) to model the interdependency among appliances. In [13] and [15] an agent-based approach to manage negotiation among the different parties is presented. The goal is to propose adaptive negotiation strategies for energy trading in a deregulated market. In particular, strategies derived from game theory are used, in order to optimize energy production and supply costs by means of negotiation and adaptation. Negotiation strategies in a multi-agent environment are also used in [14] where agents collaborate to assist human activities in safety critical scenarios. In [17], [19], [20] agents' technology is used for the negotiation and brokering of computational resources in cloud markets.

3. Energy Model

In the context of Smart Cities it is possible to model and analyze the energy profile of a house within a neighborhood so that it is possible to identify the best strategies to minimize the energy cost of the single house and of the overall neighborhood. Some notations useful for the discussion are introduced in Table 1. The proposed model is discrete-time, with sampling period T.

Table 1
Energy model parameters

Parameter	Description	Constraints
T	Sampling period	$T \geq 0$
c_a	Auto-consumed energy unit cost	$c_a \geq 0$
c_p	Provider's energy unit cost	$c_p \geq 0$
c_n	Neighbor's energy unit cost	$c_n \geq 0$
f_a	Auto-consumed energy selling indicator	$0 \leq f_a \leq 1$
f_p	Provider's energy selling indicator	$0 \leq f_p \leq 1$
f_n	Neighbor's energy selling indicator	$0 \leq f_n \leq 1$
e_r	Required energy	$e_r \geq 0$
e_{ra}	Auto-consumed required energy	$e_{ra} \geq 0$
e_{rp}	Required energy acquired from provider	$e_{rp} \geq 0$
e_{rn}	Required energy acquired from neighbor	$e_{rn} \geq 0$
e_p	Produced energy	$e_p \geq 0$
C	House total energy cost	
C_a	Auto-consumed energy cost	
C_p	Energy cost acquired from provider	
C_n	Energy cost acquired from neighbor	

3.1. House Cost Minimization

In principle it is possible to define the house required energy e_r as the sum of three contributions: the part of the required energy that is auto-consumed from the produced one, the part of the required energy acquired from a neighbor and the part of the required energy acquired from the energy provider:

$$e_r(kT) = e_{ra}(kT) + e_{rn}(kT) + e_{rp}(kT), \forall k \in \mathbb{Z}. \quad (1)$$

The auto-consumed energy cost C_a can be defined as:

$$C_a(kT) = c_a e_{ra}(kT), \forall k \in \mathbb{Z}, \quad (2)$$

where c_a can be decomposed in a constant part and a part that takes into account costs and fees for the energy consumption (f_a):

$$c_a = c + c f_a. \quad (3)$$

Thus C_a becomes:

$$C_a(kT) = c e_{ra}(kT) + c f_a e_{ra}(kT). \quad (4)$$

$$C(kT) = c\Big[\underbrace{\big(e_{ra}(kT)+e_{rn}(kT)+e_{rp}(kT)\big)}_{e_r(kT)} + \underbrace{\big(f_n e_{rn}(kT)+f_p e_{rp}(kT)\big)}_{F(kT)}\Big] \geq 0, \quad \forall k \in \mathbb{Z}$$

Fig. 1. House total cost.

In the same way it is possible to evaluate the energy cost acquired from a neighbor (C_n) and the energy cost acquired from provider (C_p):

$$\begin{cases} C_n(kT) = c_n e_{rn}(kT) \\ c_n = c + c f_n \end{cases} \rightarrow C_n(kT) = c e_{rn}(kT) + c f_n e_{rn}(kT) \ , \forall k \in \mathbb{Z}$$

$$\begin{cases} C_p(kT) = c_p e_{rp}(kT) \\ c_p = c + c f_p \end{cases} \rightarrow C_p(kT) = c e_{rp}(kT) + c f_p e_{rp}(kT) \ , \forall k \in \mathbb{Z} \tag{5}$$

The house total energy cost C is the sum of the contribution calculated in Eqs. (4) and (5):

$$C(kT) = C_a(kT) + C_n(kT) + C_p(kT)\,. \tag{6}$$

By expanding the Eq. (6) we obtain:

$$\begin{aligned} C(kT) &= c_a e_{ra}(kT) + c_n e_{rn}(kT) + c_p e_{rp}(kT) = \\ &= c e_{ra}(kT) + c f_a e_{ra}(kT) + c e_{rn}(kT) + c f_n e_{rn}(kT) + \\ &\quad + c e_{rp}(kT) + c f_p e_{rp}(kT) = \\ &= c\big[(e_{ra}(kT) + e_{rn}(kT) + e_{rp}(kT)) + \\ &\quad + (f_a e_{ra}(kT) + f_n e_{rn}(kT) + f_p)\big], \quad \forall k \in \mathbb{Z} \end{aligned} \tag{7}$$

Assuming that for the auto-consumed energy fees are nullable, $f_a = 0$ could be considered. The equation becomes as shown in Fig. 1.

Derived from the equation, the house total energy cost depends on the required energy and on a part that takes into account the fees for purchasing energy from neighbor and provider, weighed by a scale factor on the amount of required energy F. Since C is a non-negative value, to minimize the house energy cost is equivalent to tend C to zero. Given the fact that the naive solution $e_r = 0$ is a non-feasible solution (the authors are supposing that the house needs energy to power its devices), it is possible to analyze two situations:

1. The house produces more energy than it requires, $e_p \geq e_r$. In this case the best strategy is to tend F to zero, that translates in tending e_{rn} and e_{rp} to zero:

$$\begin{aligned} \min\big\{C(kT)\big\} &= \lim_{e_{rn}(kT)\to 0} \lim_{e_{rp}(kT)\to 0} C(kT) = \\ &= c\,(e_{ra}(kT) + 0 + 0)\,. \end{aligned} \tag{8}$$

   The Eq. (8) means that the best efficiency in terms of house's consumption cost is when $\boldsymbol{e_r} = \boldsymbol{e_{ra}}$, i.e., the best strategy is to auto-consume the produced energy.

2. The house requires more energy than it produces (or it is unable to produce energy), $e_p \leq e_r$. In this case the house has to acquire the required energy (or part of this) from two of the possible energy sellers, i.e., the neighborhood and the energy provider. Usually energy providers introduce significant fees and ancillary costs. Thus it is possible to assume that

$$f_p \gg f_n\,. \tag{9}$$

   In order to minimize C to minimize e_{rp} would be necessary:

$$\begin{aligned} \min\{C(kT)\} &= \lim_{e_{rp}(kT)\to 0} C(kT) = \\ &= c\,[(e_{ra}(kT) + e_{rn}(kT) + 0) + f_n e_{rn}(kT)]\,. \end{aligned} \tag{10}$$

By unifying the results reached in cases 1 and 2 it is evident that the best strategy to minimize the house energy cost is to auto-consume the produced energy and to acquire the remaining requested part from the neighborhood, thus minimizing the exchange with the energy provider.

3.2. Neighborhood Cost Minimization

The neighborhood is composed by several buildings, that can be handled as houses in presented model. In general a neighborhood is composed by NH houses that can consume and/or produce energy.

Define C_{NH} as neighborhood's total energy cost, that is:

$$C_{NH}(kT) = f\,[S_i\,(kT)]\,, \forall i \in NH, \quad \forall k \in \mathbb{Z}\,, \tag{11}$$

where S_i is the energy state of each house.

From Eq. (11) it is possible to understand that in order to find the best energy exchange in the neighborhood's that leads to a minimization of C_{NH} the neighborhood should know the energy state of the houses at any time. This requirement implies a number of technological issues:

- **Needing of a centralized controller**. In order to evaluate the best energy exchange a global vision of the neighborhood's energy state is needed. Thus there is the necessity of a centralized controller that collects data about S_i and manages energy exchanges among the houses, having scalability and efficiency losses;

- **Real-time constraints**. Time instant t depends on the sample time of the sensors that gather data within the houses and on the processing capacity of the controller. The efficiency of the minimization algorithm is bound to the performances of the used technologies;

- **Communication overhead**. Even when the houses don't need an energy exchange, they must communicate to the controller their state, thus increasing traffic on the neighborhood's network and leading the controller to become a bottleneck.

C_{NH} can be described as the sum of contribution coming from each house:

$$C_{NH}(kT) = \sum_{i=1}^{NH} C_i(kT),\ \forall i \in NH,\ \forall k \in \mathbb{Z}. \tag{12}$$

Thus it is possible to define:

$$\min\left\{C_{NH}(kT)\right\} = \min\left\{\sum_{i=1}^{NH} C_i(kT)\right\},\ \forall i \in NH,\ \forall k \in \mathbb{Z}. \tag{13}$$

In order to minimize the neighborhood's total energy cost it is possible to lighten model's requirements. Assume that each S_i is independent of any S_j:

$$S_i(kT) \perp\!\!\!\perp S_j(kT),\ \forall k \in \mathbb{Z},\ \forall i \in NH,\ j \neq i. \tag{14}$$

Equation (14) means that every house can look only at itself in order to minimize the energy cost, by acting autonomously without a centralized orchestrator. Due to the unfeasibility of a centralized solution and by taking a cue from the assumption of energy status independence, the minimization of the neighborhood's total energy cost can be processed as the minimization of the each house local energy cost:

$$\min\left\{C_{NH}(kT)\right\} = \sum_{i=1}^{NH} \min\left\{C_i(kT)\right\},\ \forall i \in NH,\ \forall k \in \mathbb{Z}. \tag{15}$$

By combining Eqs. (12) and (15), we obtain:

$$\begin{aligned} \min\left\{C_{NH}(kT)\right\} &= \sum_{i=1}^{NH} \min\{C_i(kT)\} = \\ = \sum_{i=1}^{NH} &\left\{ \begin{array}{c} \lim_{e_{rn_i}(kT)\to 0} \lim_{e_{rp_i}(kT)\to 0} C_i(kT),\ \text{if } e_{p_i}(kT) \geq e_{r_i}(kT) \\ \lim_{e_{rp_i}(kT)\to 0} C_i(kT),\ \text{if } e_{p_i}(kT) < e_{r_i}(kT) \end{array} \right\} = \\ = \sum_{i=1}^{NH} &\left\{ \begin{array}{c} ce_{ra_i}(kT),\ \text{if } e_{p_i}(kT) \geq e_{r_i}(kT) \\ c[(e_{ra_i}(kT) + e_{rn_i}(kT)) + f_{n_i} e_{rn_i}(kT)],\ \text{if } e_{p_i}(kT) < e_{r_i}(kT) \end{array} \right\}, \\ &\forall i \in NH,\ \forall k \in \mathbb{Z}. \end{aligned} \tag{16}$$

This approach brings an optimization of energy costs by using a selfishly behavior of each house, where the collaboration and communication among the houses is limited to the energy demand in case of its unavailability to auto-consume. This solution is completely distributed and doesn't need a centralized management and coordination, being highly scalable and efficient.

3.3. Energy Characterization

In order to characterize the house behavior, it is necessary to identify the constraints that the home must comply with, in energy state $S(t)$ terms. First of all, it is assumed that each house has an accumulator to store the produced energy to use or to sell if needed. Notations are introduced in Table 2.

Table 2
Energy characterization parameters

Parameter	Description
$P_p(t)$	Power produced by photovoltaic system (PV system)
$A_c(t)$	Current consumed by the load
V	House supply voltage
$P_{acc-max}$	Maximum power supplied by the accumulator
$E_{acc}(t)$	Energy stored in the accumulator in time t

The consumed power $P_c(t)$ is:

$$P_c(t) = A_c(t)V, \tag{17}$$

while $S(t)$ is defined as:

$$S(t) = P_c(t) - P_p(t). \tag{18}$$

Due to Eqs. (17) and (18), it is possible to understand that the condition for the auto-consumption is:

$$\int_{t_0}^{t_0+\Delta t} S(t)dt - E_{acc}(t_0) \leq 0. \tag{19}$$

Since the accumulator are characterized by a maximum amount of power that it is able to provide ($P_{acc-max}$), the condition for the auto-consumption becomes:

$$\left\{ \begin{array}{l} \int_{t_0}^{t_0+\Delta t} S(t)dt \leq E_{acc}(t_0) \\ S_{max} \leq P_{acc-max} \end{array} \right. . \tag{20}$$

Due to the fact that the accumulator is modeled as a capacitor, the maximum amount storable energy is:

$$E_{acc-max} = \frac{1}{2} C_{acc} V^2. \tag{21}$$

Thus:

$$E_{acc}(t) \leq \frac{1}{2} C_{acc} V^2. \tag{22}$$

Let us suppose that the production profile of the PV system and the consumption profile of the load can be predicted. This translates in estimates a-priori $P_p(t)$ and $A_c(t)$:

$$\begin{aligned} \int_{t_0}^{t_0+\Delta t} S_{est}(t)dt &= \int_{t_0}^{t_0+\Delta t} P_{\mathrm{c\ est}}(t)dt - \int_{t_0}^{t_0+\Delta t} P_{\mathrm{p\ est}}(t)dt = \\ &= V \int_{t_0}^{t_0+\Delta t} A_{\mathrm{c\ est}}(t)dt - \int_{t_0}^{t_0+\Delta t} P_{\mathrm{p\ est}}(t)dt. \end{aligned} \tag{23}$$

$$E_{acc}(t_0-\Delta t)=E_{acc}(t_0-\Delta t)+E_{surplus}[t_0-\Delta t;t_0]-E_{sold}[t_0-\Delta t;t_0]$$

$$E_{exc}[t_0;t_0+\Delta t]=E_{acc}(t_0-\Delta t)+\int_{t_0}^{t_0+\Delta t}P_{\mathrm{p\ est}}(t)dt-V\int_{t_0}^{t_0+\Delta t}A_{\mathrm{c\ est}}(t)dt$$

$$E_{\text{to-sell}}[t_0-\Delta t;t_0]=\begin{cases}E_{\mathrm{exc}}[t_0;t_0+\Delta t], \text{if } E_{exc}[t_0;t_0+\Delta t]>0\\0, \text{otherwise}\end{cases}$$

$$E_{\text{to-buy}}[t_0-\Delta t;t_0]=\begin{cases}\left|E_{\mathrm{exc}}[t_0;t_0+\Delta t]\right|, \text{if } E_{\mathrm{exc}}[t_0;t_0+\Delta t]<0\\0, \text{otherwise}\end{cases}$$

Fig. 2. Prediction algorithm.

The authors assume that only energy stored in $t_0-\Delta t$ is available so to plan the necessary actions to undertake in $t_0<t<t_0+\Delta t$. By predicting $P_p(t)$ and $A_c(t)$ it is possible to forecast the amount of exceeding energy:

$$E_{exc}[t_0;t_0+\Delta t]=E_{acc}(t_0-\Delta t)+ \\ +\int_{t_0}^{t_0+\Delta t}P_{\mathrm{p\ est}}(t)dt-V\int_{t_0}^{t_0+\Delta t}A_{\mathrm{c\ est}}(t)dt \qquad (24)$$

By forecasting the amount of exceeding energy, it can be evaluated if there is energy to sell or buy:

$$E_{exc}[t_0;t_0+\Delta t]=\begin{cases}E_{\text{to-sell}}[t_0-\Delta t;t_0], \text{ if } E_{exc}[t_0;t_0+\Delta t]\geq 0\\E_{\text{to-buy}}[t_0-\Delta t;t_0], \text{ if } E_{exc}[t_0;t_0+\Delta t]<0\end{cases}. \qquad (25)$$

Since the load optimization operations and the prediction starts in $t_0-\Delta t$, and the forecasting is valid for the period between t_0 and $t_0+\Delta t$, **the time limit for publishing the proposal and the energy requests and for closing the evaluations is Δt**.

By taking a cue from the described relations, the steps that the houses should do each timespan Δt are described by the algorithm in Fig. 2. The authors suppose that is possible to get information about the energy stored by the accumulator at any instant. This value is not the real amount of energy available to be auto-consumed because there is the possibility that the house decided the selling part of energy to a neighbor in the previous timespan. If $E_{to-sell}[t_0-\Delta t;t_0]$ is greater than zero, the house publishes a proposal to sell energy that is valid until t_0. By the contrary, if $E_{to-buy}[t_0-\Delta t;t_0]$ is greater than zero, the house publishes a energy request in order to buy the future consumed energy from someone in the neighborhood. The search and the evaluation are allowed up to t_0, if the evaluation fails or there are not proposals during this period, the house buys the needed energy from the provider. Since it is possible to acquire from a neighbor more energy than the required one, it is possible to store the exceeding amount in the accumulator. This energy is taken into account by $E_{surplus}[t_0-\Delta t;t_0]$. In this way there is the possibility that also a building that has not production facilities can become a seller.

The algorithm relies on the knowledge about the power production and consumption in the future. For $P_p(t)$ it is possible to use historical data about the production of the PV panels and to rely on short-term weather forecasts. For the estimation of $A_c(t)$ it is possible to use historical series and the current scheduling of the expected loads (dishwasher, washing-machine, etc.) generated according to some optimization actions of the house's loads. Boundaries such as $P_{acc-max}$ and $E_{acc-max}$ can be used as evaluation's parameters in the proposals and in the energy requests.

The one-step prediction used in the algorithm could lead to performances that aren't the best for the single house. In fact, suppose that in $t_0-\Delta t<t<t_0$ an house decides to sell energy because it predicts that in $t_0<t<t_0+\Delta t$ it has an energy surplus. After that, if the evaluation succeeds, it predicts that in $t_0+\Delta t<t<t_0+2\Delta t$ it needs to buy energy because it doesn't have enough energy stored in order to satisfy the load in this timespan and it is forced to make an energy request in $t_0<t<t_0+\Delta t$. In this case it is evident that the best for the house would have been not to sell the energy so to have enough energy stored to auto-consume also in $t_0+\Delta t<t<t_0+2\Delta t$. Even if it seems that a multi-step prediction has better performances for the single house, the one-step prediction has a lower computational complexity which corresponds to a higher reactivity in the application of the algorithm, which becomes crucial by dealing with a system with strong real-time constraints. Moreover, since the algorithm is based on the usage of historical data and forecasts, a short-term prediction has more accuracy than a long-term one, that impacts positively on the prediction performances.

One way to give more robustness to the algorithm is to change the evaluation of $E_{\text{to-sell}}[t_0-\Delta t;t_0]$ and $E_{\text{to-buy}}[t_0-\Delta t;t_0]$ as follows:

$$E_{\text{to-sell}}[t_0-\Delta t;t_0]=\begin{cases}E_{\mathrm{exc}}[t_0;t_0+\Delta t]-\varepsilon_{sell}, \text{ if } E_{exc}[t_0;t_0+\Delta t]>0\\0, \text{ otherwise}\end{cases},$$

$$E_{\text{to-buy}}[t_0-\Delta t;t_0]=\begin{cases}\left|E_{\mathrm{exc}}[t_0;t_0+\Delta t]\right|+\varepsilon_{buy}, \text{ if } E_{exc}[t_0;t_0+\Delta t]<0\\0, \text{ otherwise}\end{cases}, \qquad (26)$$

where ε_{sell} and ε_{buy} are parameters that take into account possible forecasting errors in selling and buying energy.

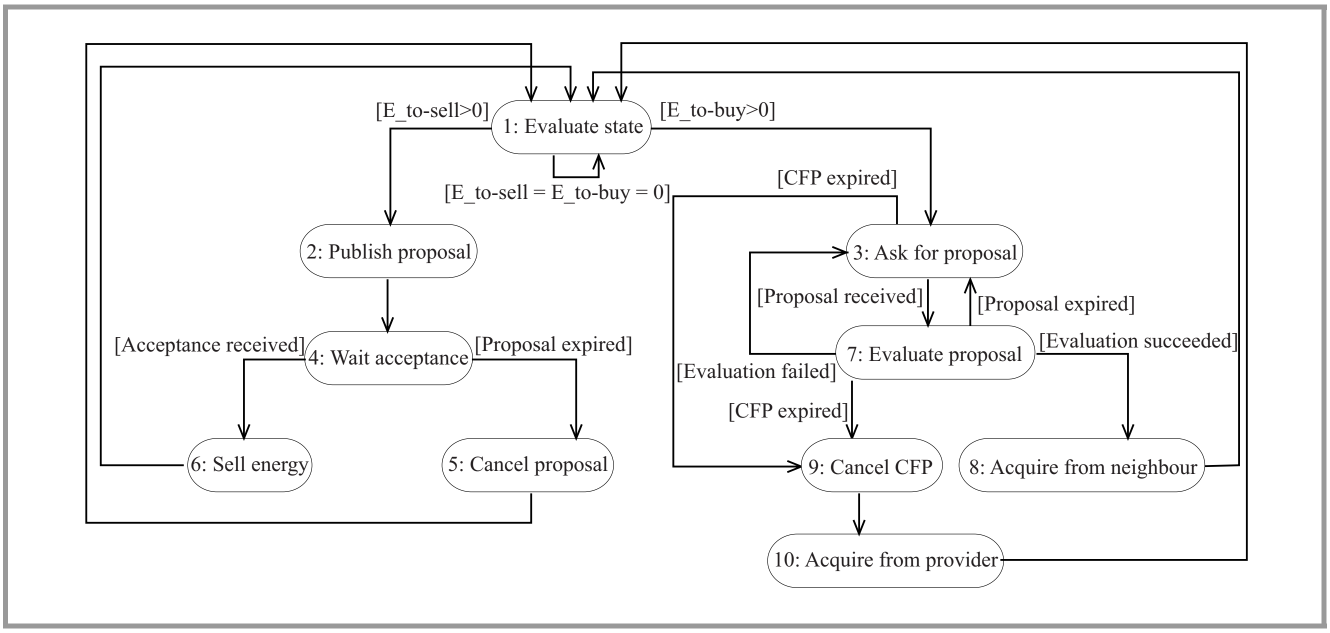

Fig. 3. Agent interaction model for energy cost minimization.

3.4. Agent Interaction Model for Cost Minimization

To implement the abovementioned strategy, the agent paradigm is used, building up an interaction model for Collective Intelligence that aims at minimizing the overall neighborhood's cost. Each house is modeled by an agent that adapts its behavior in order to maximize auto-consumption of energy and minimize the exchange with the energy provider. Thus the neighborhood is represented by a number of agents that are distributed within a "virtual" community and run autonomously in order to implement their own strategy. Since every house might have different sensors in order to retrieve information about the energy consumption/production of the devices, the connection between sensors and the agent is implemented by a RESTFul gateway that is in charge of translating the events in an agent common language and forwarding them to the agent [18]. Thanks to its reactiveness and proactiveness capabilities, the agent paradigm is able to match the described selfish behavior with on-demand collaboration in a distributed environment by using an asynchronous communication approach. The agent technology allows to easily react to environment's changes in order to reach the cost minimization goals. Moreover, the architecture is highly scalable and can easily grow and decrease with the neighborhood by simply adding and removing agents from the platform, thus exploiting the complete decoupling among the agents.

The minimization's strategy can be translated in three agent's macro-behaviors:

- **maximize auto-consumption** – whatever state the agent is in, if the agent needs energy and an energy production's event occurs, this event triggers a series of state transitions that lead it to consume the produced energy;
- **minimize energy requests to the provider** – if there is an energy request and the produced energy is not sufficient to completely satisfy the request, the agent asks for the needed energy to the neighborhood;
- **collaborative approach** – if the house has an excess of produced energy, the agent provides this energy to the neighborhood.

The agent interaction model is drawn in Fig. 3 while the description of each state is provided in Table 3.

Being consistent with the discrete-time model presented in Section 3, even if the interaction model is event-based, the full set of operations is marked by Δt. Every Δt the automata returns to its initial state, starting a new round of estimation-trading-purchasing/selling.

As it is possible to understand, Δt becomes a crucial parameter for the algorithm performances. Too small a value of Δt makes stressing the prediction algorithm and might be too short to complete the negotiation phase. Too high a value of Δt makes the energy performances of the house too bind to the accuracy of the forecasting. For these reasons, the tuning of Δt strongly impacts on the house cost minimization.

4. Prototype Implementation

As described in Section 3.4, it is possible to map the house behavior to an agent in charge of performing the operations aimed at minimizing the energy cost. The designed interaction model has been implemented by using the agent technology. Execution environment for agents and communication facilities are provided by the JADE agent platform [3], that supplies an execution environment of soft-

Table 3
Agent's state description

No.	State	Description
1	Evaluate state	In this state are performed all the operation described in Fig. 2.
2	Publish proposal	If the house produces some exceeding energy, the agent publishes a proposal in order to sell the energy to other houses in the neighborhood.
3	Ask for proposal	If the house needs energy and it has not produced one, it asks the neighborhood for energy to buy by using a Call for Proposal (CFP).
4	Wait acceptance	In this state the house waits for acceptance of a proposal published in state 2.
5	Cancel proposal	If during the waiting of a proposal acceptance notification t passes, the proposal is canceled.
6	Sell energy	If a proposal acceptance notification has been received, the agent sells the agreed energy to the buyer.
7	Evaluate proposal	If a proposal is received, the agent evaluates it in order to the buy neighbor's energy.
8	Acquire from neighbor	If a proposal evaluation succeeded, the agent buys the agreed energy from the seller.
9	Cancel CFP	If during a proposal evaluation or the waiting of proposals t passes, the CFP is canceled.
10	Acquire from provider	This is the worst state in which the agent can be. If the agent needs energy and no acceptable proposals come within Δt, the only thing that the agent can do is to acquire the needed energy from the energy supplier.

ware agents, an Agent Communication Channel (ACC) and some protocol implementation to support communication. AMS and DF provide standard services of FIPA compliant agent platforms [4]. A management system for agents and a yellow pages registry for publication and discovery of agent based services. Agents will communicate among them via standard ACL (Agent Communication Language). JADE is completely written in Java so that each agent is represented by a Java class as well as the behaviors of every agent.

The agent representing the house is called Energy Agent (EA). It is composed by a number of behaviors that implement the Finite State Machine (FSM) designed in Fig. 3. Each behavior contains the particular operations that characterize the state of the house. For example, the *Evaluate State* behavior includes the forecasting of $P_p(t)$ and $A_c(t)$ as well as the prediction algorithm described in Fig. 2, while the *Evaluate Proposal* behavior embeds the algorithms used to evaluate a given proposal against a submitted CFP.

In order take in account the temporal constraint given by Δt, it is used a *Watchdog* behavior that runs in parallel with the ones representing the state of the FSM. If Δt passes and it marks the proposal/CFP as expired by writing a particular variable in memory. Each state of the FSM controls this variable and adapts its behavior according to the read value, being compliant with the described interaction model. When the EA is in the *Evaluate State* behavior, it resets the *Watchdog* behavior in order to restart all the operations.

To ensure the scalability of the distributed platform, it has been used a bus-based approach. When an agent wants to sell some energy, it publishes the proposal on the bus and waits for an acknowledgement coming from someone in the neighborhood that is interested in buying its energy. When the proposal expires, it simply withdraws this from the bus. If someone is evaluating the proposal, it is notified about the withdrawal. On the other hand, if an agent is interested in buying some energy, it can retrieve a proposal from the bus (if any) and can evaluate it. The bus usage for the communication within the neighborhood allows also the synchronization among sellers and buyers. When a buyer asks for a proposal, the bus gives to the asker the first proposal in the queue that is not yet under the evaluation by another agent. In fact, when an agent is evaluating a proposal, it puts a lock on it in order to prevent that someone can evaluate at the same time the same proposal. If the evaluation succeeds, the seller is alerted and it starts to give the agreed energy to the buyer. If the evaluation fails,

Table 4
Agent Bus operations

Method	Description
Publish proposal	This method allows a seller to publish a new proposal to be evaluated by other agents in the neighborhood.
Ask for proposal	It is used by a buyer to retrieve the first unlocked proposal in the queue (if any). It returns a proposal and locks it in case of success, a null value otherwise.
Release proposal	This method allows a buyer to unlock a proposal that has been evaluated and refused.
Accept proposal	In order to mark a proposal as accepted, a buyer can use this method.
Receive acceptance notification	This method is used by a seller in order to ask the bus about the acceptance of a published proposal by a buyer. It returns a boolean value, true if accepted, false otherwise.
Cancel proposal	When a proposal expires, the publisher can use this method to withdraw the proposal.

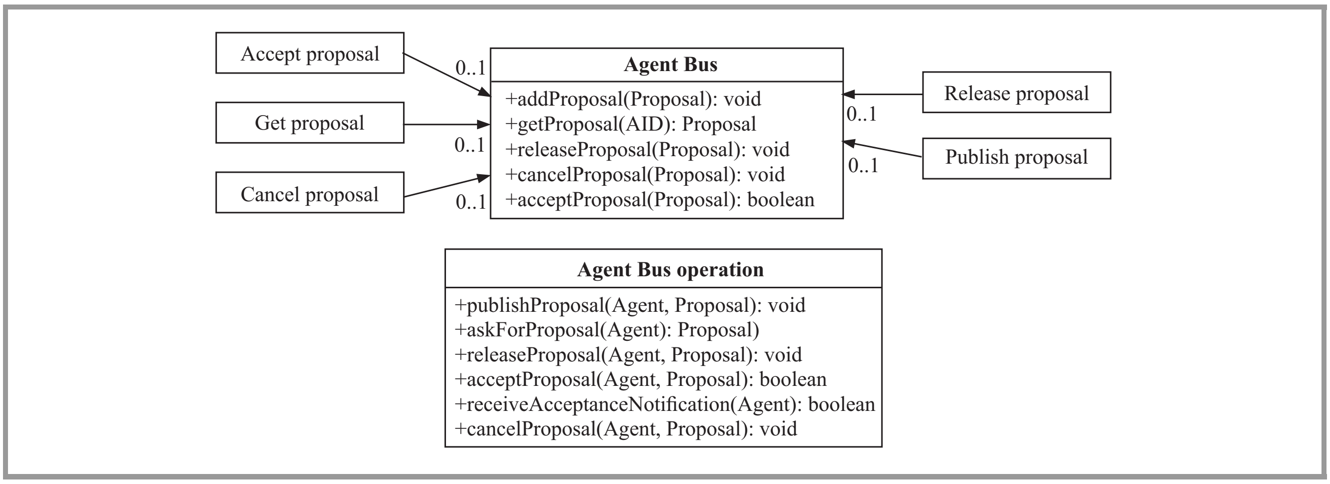

Fig. 4. Agent Bus class diagram.

the buyer unlocks the proposal so that it can be evaluated by others.

The bus can be realized by using different technologies, such as queue servers like ActiveMQ [5], RabbitMQ [6], etc. In first prototype, the bus has been implemented as an agent within the platform, called Agent Bus (AB). AB runs at boot time and provides to the EAs all the operations they need in order to perform the overmentioned operations. In particular, the methods that the bus makes available are described in Table 4.

As it is possible to understand from the AB class diagram in Fig. 4, the Agent Bus Operation exposes operations that are used by each EA and embeds the ACL messages sent and received to/from the AB in order to perform the chosen action: in other words, EAs and AB are connected by using Agent Bus Operation via messages' exchange. On the other hand, for each operation, the AB has a particular behavior that allows to receive the specific message and to act on its data structures in order to perform the requested action.

4.1. Experimental Results

In order to validate the proposed approach a synthetic workload built up by using five buildings in a neighborhood is used. We define *consumer* a building that has not energy production facilities and, in its normal behavior, it has only the possibility to consume energy. By the contrary, a *prosumer* is a building that has energy production capabilities. In presended experiments the energy profiles of three consumers and two prosumers are used and the attention is focused on a consumer, called *target*. As previously said, the predefined energy profiles for each building are used, thus zeroing the time for $P_p(t)$ and $A_c(t)$ estimations. Furthermore, it is assumed that these estimations are correct, thus not introducing errors in the prediction phase.

The experiments aim at evaluating the impact of Δt on the house performances varying the buildings in the neighborhood. The Δt is set on two consumers and two prosumers to a fixed value $\Delta t_{others} = 1000$ ms. The Δt_{target} was varied at 500, 1000, 2000 ms, gradually introducing buildings in the neighborhood.

In order to understand the performances of the prototype, the percentage of occurrences of the *Acquire from Provider* state is analyzed, that represents the less favorable state of the agent (Fig. 5). As it is possible to see, Δt_{target} strongly impacts on the number of occurrences of this state. In fact a greater value of Δt_{target} provides much time to evaluate proposal in the neighborhood before the CFP expires. However the performances are also influenced by the ratio among consumers and prosumers within the neighborhood. If there are too many consumers with respect to prosumers, the speed in evaluating the proposals becomes crucial and consequently the energy performances are closely linked to the performances of the evaluation algorithm. The introduction of a new prosumer radically changes the scenario, as reported in the chart.

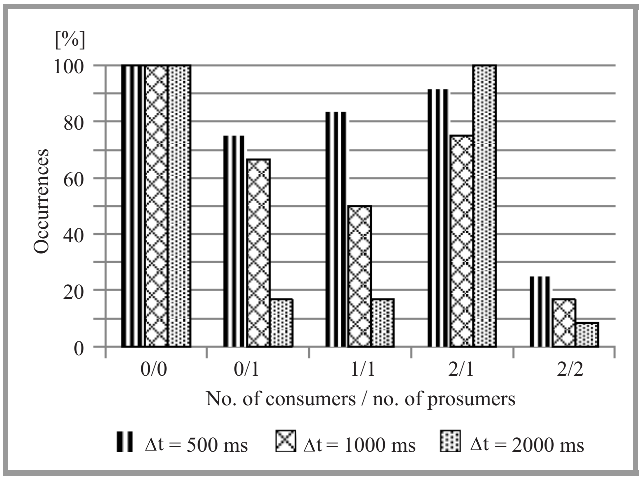

Fig. 5. Acquire from provider percentage of occurrences.

Another interesting result coming from the experiments is evincible by looking at Figs. 6 and 7. The fact that the values in these charts are not completely null denotes the

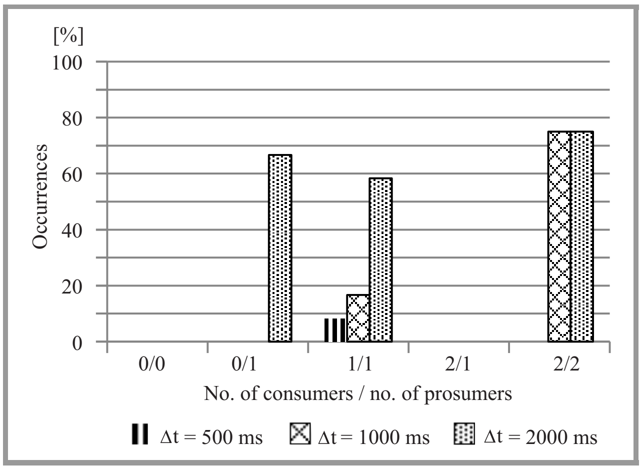

Fig. 6. Cancel proposal percentage of occurrences.

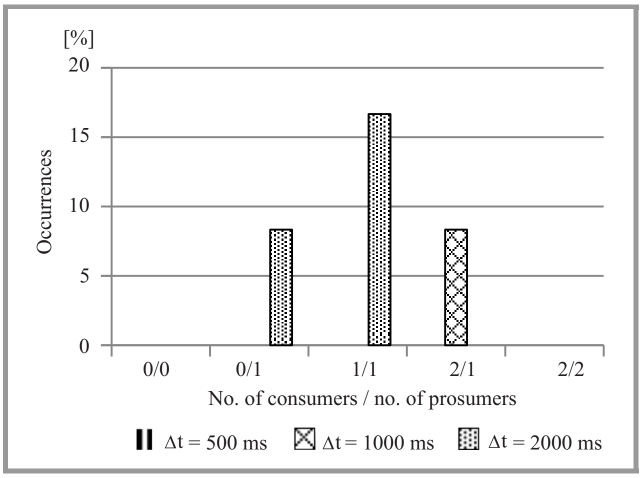

Fig. 7. Sell energy percentage of occurrences.

situation in which target bought more energy than it needed and it converts itself to a seller, publishing proposals and, in some cases, being able to sell excess energy.

5. Conclusion

In this paper authors present a model for the energy cost minimization of a neighborhood. The energy cost function of a single house at first is analyzed and modeled. After that the authors model, under houses' independence hypothesis, the neighborhood energy cost function and how to minimize it. Also a characterization of the house behavior is proposed in terms of energy production and consumption and a way to reach the cost minimization by using predictions and load estimation. On this basis, an agent-based interaction model that aims at maximize the auto-consumption of the produced energy and at buying the needed one from neighbors instead of supplier is presented. The validation of the interaction model has been performed by developing and testing a prototypal model implementation. Experimental results highlight how a correct tuning of the operations timespan has a strong impact on the performances, as well as a balanced ratio among the number of consumers and prosumers can play a crucial role on the performances of the whole neighborhood. The authors are planning other experiments aimed at evaluating the performances of the prototype by having different timespans for each building within the neighborhood. Furthermore, future works will deal with the introduction of constraints on the house devices control in order to optimize the cost function as well as the introduction of algorithms for the estimation of produced and consumed energy.

Acknowledgements

This work has been partially supported by the European Community's Seventh Framework Programme as part of the ICT CoSSMic project (FP7-ICT-608806)

References

[1] D. Giusto, A. Lera, G. Morabito, and L. Atzori, *The Internet of Things*. Springer, 2010.

[2] K. Su, J. Li, and H. Fu, "Smart city and the applications," in *Proc. Int. Conf. Elec. Commun. Contr. ICECC 2011*, Ningbo, China, 2011, IEEE, pp. 1028–1031.

[3] F. Bellifemine, A. Poggi, and G. Rimassa, "JADE–A FIPA-compliant agent framework," in *Proc. Pract. Appl. Intell. Agents Multi-Agents PAAM 1999*, London, 1999, vol. 99, no. 97–108.

[4] "FIPA" [Online]. Available: http://www.fipa.org

[5] B. Snyder, D. Bosnanac, and R. Davies, *ActiveMQ in Action*. Shelter Isl., USA: Manning, 2011.

[6] A. Videla and J. J. Williams, *RabbitMQ in Action*. Shelter Isl., USA: Manning, 2012.

[7] A. Rogers, S. D. Ramchurn, and N. R. Jennings, "Delivering the smart grid: Challenges for autonomous agents and multi-agent systems research", in *Proc. 26th AAAI Conf. Artif. Intell. AAAI-12*, Toronto, Canada, 2012, pp. 2166–2172.

[8] I. Praça, C. Ramos, Z. Vale, and M. Cordeiro, "Intelligent agents for negotiation and game-based decision support in electricity markets", *Engin. Intell. Syst. Elec. Engin. Commun.*, vol. 13, no. 2, p. 147, 2005.

[9] T. A. Rodden, J. E. Fischer, N. Pantidi, K. Bachour, and S. Moran, "At home with agents: exploring attitudes towards future smart energy infrastructures", in Proc. SIGCHI Conf. Human Fact. Comput. Syst., Paris, France, 2013, pp. 1173–1182.

[10] Y. Jia-hai, Y. Shun-kun, and H. Zhao-guang, "A multi-agent trading platform for electricity contract market", in *Proc. 7th Int. Power Engin. Conf. IPEC 2005*. Singapore, 2005, pp. 1024–1029.

[11] D. Whitehead, "The El Farol bar problem revisited: Reinforcement learning in a potential game", *ESE Discussion Papers*, no. 186, Edinburgh School of Economics, 2008.

[12] N. C. Truong, L. Tran-Thanh, E. Costanza, and D. S. Ramchurn, "Activity prediction for agent-based home energy management", in *Proc. 4th Int. Worksh. Agent Technol. for Energy Syst. ATES 2013*, Saint Paul, Minnesota, USA, 2013.

[13] N. Capodieci, E. F. Alsina, and G. Cabri, "A context-aware agent-based approach for deregulated energy market", in *Proc. 21st IEEE Int. Worksh. Enabling Technol. Infrastruc. Collabor. Enterpr. WETICE 2012*, Toulouse, France, 2012, pp. 16–21.

[14] R. Aversa, B. Martino, M. Ficco, and S. Venticinque, "Simulation and support of critical activities by mobile agents in pervasive and ubiquitous scenarios," in *Proc. 10th Int. Symp. Parall. Distrib. Process. Appl. ISPA 2012*, Madrid, Spain, 2012, pp. 815–822.

[15] N. Capodieci, G. Cabri, G. A. Pagani, and M. Aiello, "Adaptive game-based agent negotiation in deregulated energy markets," in *Proc. IEEE Int. Conf. Collabor. Technol. Syst. CTS 2012*, Denver, CO, USA, 2012, pp. 300–307.

[16] A. Amato *et al.*, "Software agents for collaborating smart solar-powered micro-grids", in *Smart Organizations and Smart Artifacts*, L. Caporarello, B. Di Martino, and M. Martinez, Eds. *Lecture Notes in Information Systems and Organisation*, vol. 7, pp. 125–133, Springer, 2014.

[17] S. Venticinque, L. Tasquier, and B. Di Martino, "A restfull interface for scalable agents based cloud services", *Int. J. of Ad Hoc and Ubiquitous Comput.*, vol. 16, no. 4, pp. 219–231, 2014.

[18] L. Tasquier, M. Scialdone, R. Aversa, and S. Venticinque, "Agent based negotiation of decentralized energy production", in *Intelligent Distributed Computing VIII*, D. Camacho, L. Braubach, S. Venticinque, and C. Badica, Eds. *Studies in Computational Intelligence*, vol. 570, pp. 59–67. Springer, 2015.

[19] S. Venticinque, L. Tasquier, and B. Di Martino, "Agents based cloud computing interface for resource provisioning and management", in *Proc. 6th IEEE Int. Conf.Complex, Intell. Softw. Intens. Syst. CISIS 2012*, Palermo, Italy, 2012, pp. 249–256.

[20] R. Aversa, L. Tasquier, and S. Venticinque, "Cloud agency: A guide through the clouds," *Mondo Digitale*, vol. 13, no. 49, 2014.

Luca Tasquier is Ph.D. student in Computer and Electronic Engineering at Department of Industrial and Information Engineering of the Second University of Naples. He received his Master degree in Computer Engineering in 2011. He is involved in research activities dealing with parallel and cloud computing and mobile/intelligent agents for distributed systems. He participated to research projects supported by international and national organizations.

E-mail: luca.tasquier@unina2.it
Dipartimento di Ingegneria Industriale e dell'Informazione
Second University of Naples
Via Roma, 29
81031 Aversa, Italy

Rocco Aversa graduated in Electronic Engineering at University of Naples in 1989 and received his Ph.D. in Computer Science in 1994. He is Associate Professor in Computer Science at the Department of Information Engineering of the Second University of Naples. His research interests are in the area of parallel and distributed systems. The research themes include: the use of the mobile agents paradigm in the distributed computing, the design of simulation tools for performance analysis of parallel applications running on heterogeneous computing architectures, the project and the development of innovative middleware software to enhance the grid and cloud computing platforms. He is associate editor of International Journal of Web Science (IJWSM Inderscience Publishers). Dr. Aversa participated to various research projects supported by national organizations (MURST, CNR, ASI) and by EC in collaboration with foreign academic institutions and industrial partners.

E-mail: rocco.aversa@unina2.it
Dipartimento di Ingegneria Industriale e dell'Informazione
Second University of Naples
Via Roma, 29
81031 Aversa, Italy