Doctoral School on Engineering Sciences
Università Politecnica delle Marche

Extended summary

Modeling and simulation of home automation systems for energy management

Curriculum: Informatics, Managerial and Automatics Engineering

Author

Riccardo Donnini

Tutor

Prof. Giuseppe Conte

Date: February 4th, 2013

Abstract. The following work deals with the modeling of home automation systems based on the theory of Multi Agent Systems (MAS). The product of the work is the development of a complex simulation/emulation environment capable to contain a number of components of an home automation system. The work has led to collaborations with research institutions and sector company. For a correct modeling process of a home automation system is necessary to deepen the aspects that characterize the structure and elements (agents) that populate it. About structure of the model, we refer to the MAS approach that will be used as a preliminary setting. This approach is suitable for the description of domotic environments, in fact is possible to associate them to agents, where each one has its own goals and needs. To reach this goal we based on the UML modeling approach, which allowed the concentration of heterogeneous information within descriptions, such as agents that could be very different one from each other. During the development of the simulation environment there is the need of dealing with a variety of tools, technologies and employees' technical levels; the UML theory allows us to formalize representations which are still valid in different and meaningful contexts. The agents are modeled in this work are hybrid objects, characterized by a finite-state dynamic (which represent their behavior) and a component described by a Petri Net (PN) which models their interaction with the energy source. The simulation tool provided is also used to test control strategies in order to meet the system constraints such as limited availability of resources or additional specifications that the system has to comply.

Keywords. simulation, multi agent, Petri net, UML, domotic.

1 Problem statement and objectives

The problem of managing efficiently the resource represented by electricity in the home environment is becoming more and more important as it cost grows and as contract terms offered by suppliers diversify according to the evolution of the market. In order to develop strategies and to construct home automation systems which can help the user in coping with that problem, an important step consists in modeling the way in which appliances interact and possibly compete in accessing to electricity, when its use is limited by economic factors and other constraints.

The aim of this work is to propose an approach to modeling and to studying home automation systems consisting of a number of appliances (e.g. washing machine, dishwasher, oven, refrigerator, boiler) and other home management devices or subsystems (e.g. for heating, air conditioning, entertainment), which perform specific tasks by exploiting a common source of electricity, whose use may be subject to limitations. In our construction, we employs the basic paradigm of the Multi-Agent System (MAS) Theory and the Petri Net formalism for describing the dynamics of the consumption of electricity in home scenarios. The main advantage of such approach is that of providing a very versatile tool for modeling real, complex situations in a virtual simulation environment and for studying feasibility and performances of control strategies.

The modeling framework we construct provides a simple but powerful representation of domotic environments in which the electric load (at all or only at) some selected plugs can be monitored, together with the global load. Information about loads at selected plugs, to which specific appliances are connected, and about global load represents the basic information needed for implementing a strategy that, by scheduling in some way the operations which consume electricity, attempts to improve the management of that resource. Home automation market offers different kinds of devices, generically called intelligent plugs, which are equipped with a meter and which can make available the results of the measurements for recording or processing to a monitoring or control unit. Such devices can substitute traditional plugs to connect, in a fixed configuration, some appliances, like washing machine, dish washer and refrigerator. Intelligent plugs can also be equipped with switches that disconnect from the power grid according to commands originated by a control unit [1], [2]. In order to provide an abstract, formal notion of the structure that we call Home Automation System, or HAS, we refer to the approach developed in [3–8] on the basis of the Multi Agent System Theory. The use of suitable notions of agent and governing laws provide a way to describe the appliances and the way in which they interact in accessing the common resource represented by electricity and, possibly, exchange, information. The resulting model is a dynamical system that is composed by many independent agents, whose behaviors are subject to both internal (that is: of the agent) and external (that is: of the interaction between agents) rules. A similar approach to modeling domotic systems is that employed in [9].

Agents which model real appliances and domotic devices are constructed as hybrid dynamical systems, consisting of two interconnected components which influence each other. One component is basically a time-driven dynamical system and the other is an event-driven dynamical systems. The time-driven component (called BEHAVIOR) models the functional behavior of the appliance with respect to the electric load it generates, while the

event-driven components (called PLUG) models its interaction with the power grid of the house.
Structuring domotic agents in this way allows us to decouple the aspect related to the functional behavior from those concerning demand and availability of resources. In order to implement the model for study and simulation, BEHAVIOR components are realized as objects in a LabVIEW environment according to a suitable UML procedure, which is based on study and observation of real appliances. PLUG components have a fixed, predefine structure and are modeled by an ordinary, extended, limited and conservative Petri Net (PN).
Interconnections between BEHAVIOR and PLUG is established in term of availability of electricity supplied by the power grid. PLUG components incorporate in their logic structure the actuating mechanism that is needed to implement monitoring and resource leveling strategies, possibly enforced by the controller. In other terms, PLUG components model real, intelligent electric plugs, which incorporate a meter and a controlled switch.
A specific Petri Net procedure is defined in order to combine PLUG components of different agents into an event-driven system which describes the overall behavior of the HAS. Marking of the nodes in the resulting net describes the status of the HAS in terms of actual loads and exhibits the information stored in the system. Agents can easily be added to or removed from the HAS without altering its formal structure and, at the same time, the maximum load the power grid can accept can be modified.
Models of HAS obtained by the approach briefly described above are shown to be easily employable for simulating and studying the behavior and the performances of power leveling strategies that aims at regulating the consumption of electricity under various kinds of constraints and limitations. We show, in particular, how the information provided by the model can be used for managing the problem of keeping the global load under a chosen threshold by means of a suitable leveling strategies and, in another situation, how it can be used for regulating the energy consumption over a given time by means of a sort of scheduling.

2 Research planning and activities

In order to develop a concrete procedure to model real home automation systems as abstract HAS let us start by describing how to represent appliances and devices as agents, according to the point of view illustrated in the previous Section.
Since electricity is the most critical resource in the home environment and the most important from the point of view of saving, in considering the behavior of a generic electric device we are mainly interested in the aspects related to the electric load it generates while functioning. In case of concurrent or heavy use by many appliances and devices, the required amount of electricity may be or become not available, due to limitations of the power grid or to the implementation of saving policies. In the first case we speak of overload situations, which stress the grid and, if they persist, cause black-out. In order to avoid overloads or to allow the implementation of saving policies that limit in real-time (in particular, without employing scheduling policies) the load according to time slots and/or thresholds, it is useful to envisage the use of appliances that can stop their operations when electricity become insufficient and resume them at a later time, when electricity return to be available, without significantly degrading their performances, except possibly with respect to duration. Appliances of that kind that work by performing a preprogrammed cycle of opera-

tions, like dishwasher and washing machine, exhibits two fundamental modes of operation when they are turned on. In the first mode, electricity is assumed to be available and the load of the appliance can be represented as a piecewise constant function of time. When it is in the first mode, the appliance is said to be ON. In the second mode, electricity is assumed to be unavailable for the appliance and the load is set to 0. In this mode, the appliance is said to be in STAND-BY. In STAND-BY, it keeps memory of the step reached in its cycle and it is ready to resume operations from that as soon as power is restored. Note that several appliances, of different brands, found on the market (including oven, washing machines, dishwashers and others) can go to a stand-by condition as the one we postulate in case power is cut off. In general, modern appliances could easily be endowed with this features by means of little technological effort by producers. Home PC connected to UPS units have a behavior that, from the point of view of the load they generate, can be considered of the same kind. This holds also for simple electric devices that do not work by performing a cycle of operations, like lamps. Clearly, in the three mentioned cases, the STAND-BY mode represents a quite different condition from the functional point of view, but, from the point of view of the load, it is characterized in the same way by null load. We can then assume the above behavior as a general model for the appliances in the home environment. For completeness, we say also that the appliance is in OFF mode when it is turned off. Intelligent plugs endowed with switches can be used to cut off power selectively for implementing leveling or saving strategies, making a scenario in which the possibility to stand-by for some time and to resume operations later is largely exploitable.

2.1 BEHAVIOR model

According to the above discussion, we model appliances of interest as hybrid systems which evolve at two levels. The first level concerns the way in which the appliances operates for accomplishing its task. At that level, for any given task (e.g. for any washing program in the case of the washing machine or for any cooking program in the case of the oven), the system may exhibit one of two time-driven dynamics and it may switch from one of them to the other according to the occurrence of a given event. The first dynamics, denoted by w(t), represents the load as function of time in the ON mode. As anticipated above, w(t) is assumed to be a piecewise constant function on a time interval [0, t1] and, for simplicity, we assume that its value is quantized with a minimum step of 100 W. The length of [0, t1] indicates the time nominally required to accomplish the given task in case power is available, which is zero outside [0, t1]. If the appliance is turned on at time t0, its load will be described by w(t+t0). If power is cut off at time tc, while the appliance is working, it switches to the STAND-BY mode, in which the load is represented by a second dynamics w0(t), which is constant and equal to zero. If the power is restored at time tr, the appliance recovers the working phase, which was interrupted by power cut-off. In such case, the system switches again to the ON mode and the load evolves according to w(t-(tr-tc)). Then, the appliance will finish its task at time t1+(tr-tc), if no other cut-off happens. Figure 1 gives a schematic representation of the above model, which describes the evolution of the appliance at the first level and is generically called BEHAVIOR.

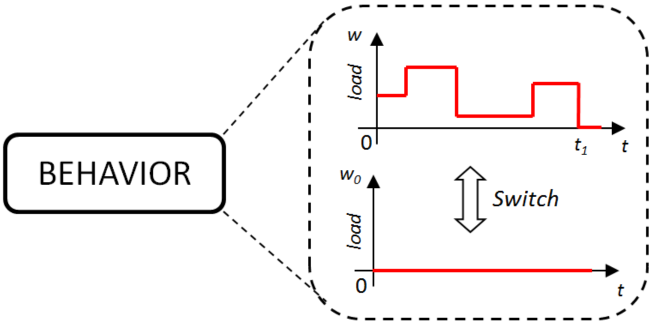

Figure 1 - BEHAVIOR model

An representation of a sate evolution of Agent's BEHAVIOR may be equipped by a state diagram. The sate diagram can move form UML representation to the software implementation using virtual instruments provided by the LabVIEW programming language where state diagram is present like Statechart [10].

2.2 PLUG model

The switching rule σ between the two dynamics can be viewed as a binary function that depends on some external variable. In order to describe it more precisely, we need first to explain how to model the evolution of the appliances at the second level, which concerns the way in which the appliance interacts with the power source. To this aim, we consider the simple Petri Net described in Figure 2. Place S represent the power source and place A represents the appliance at issue. The initial marking of S define the maximum load the power grid can accept: 1 token is here conventionally assumed to represents 100 W. The marking of A represents the current load generated by the appliance. When the first dynamics is active (ON mode), transition T1 fires at any occurrence of a positive variation of the load (in particular at the time the appliance is turned on), while transition T2 fires at any negative variation of the load (in particular at the time the appliance is turned off). Transitions fires a number of times equal to the absolute value of the variation divided by 100 and a token moves each time from S to A or back, so to reach a condition, called operating condition, in which the marking of A is equal to the actual load. We assume that firing and transition of tokens are instantaneous. The event-driven dynamics modeled by the Petri Net is generically called PLUG.

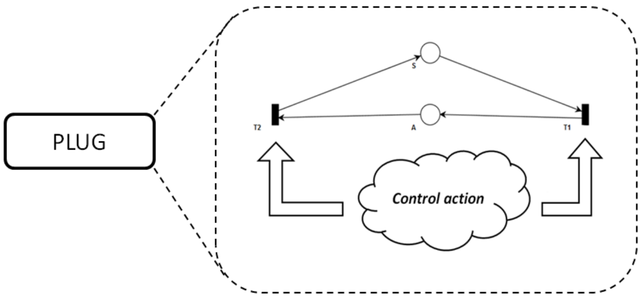

Figure 2 - PLUG model

It may happen that the operating condition expressed by the equality between the actual load and the marking of A cannot be achieved or maintained in two different situations. The first one is that in which the number of token is insufficient to match w(t). This represents an overload situation and, since persistence in it will cause blackout, the model loses validity when it occurs. The second situation is that in which T1 is inhibited by some external action. If this happens while the marking of A is nonzero, we assume that inhibition of T1 implies also that T2 fires until the marking of the place A goes to 0. In such case, the switching rule σ changes value and this forces the BEHAVIOR to switch to the STAND-BY mode. Inhibition of T1 represents, in this way, the effect of a switch that disconnect the appliance from the power grid. We will conceptually represent the situation in which connection to the power grid can be operated by a switch (e.g. the switch that equips an intelligent plug) by a scheme as in *Figure 3*. Note that control can be viewed as the action of an automatic controller which implements some leveling strategy or, more simply, as the action of the user.

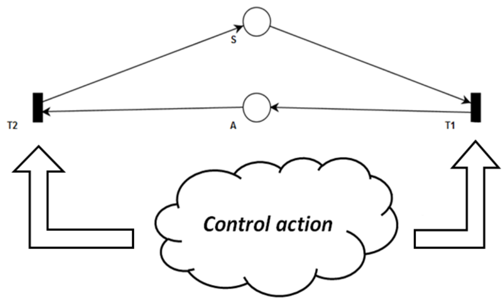

Figure 3 - Controlled PLUG model

Globally, the agent that represents an appliance consists of a BEHAVIOR component, together with a PLUG component, which interacts dynamically as described above. The BEHAVIOR component models the evolution of the load over time, while the PLUG component models the interaction with the power grid. The agent accepts a binary input that represents the on/off command coming from the user and, possibly, it may accept a second binary input that represents the control action on T1. The marking of A represents the output of the agent that we are interested to observe. Schematically, agents can be represented as described in Figure 4.

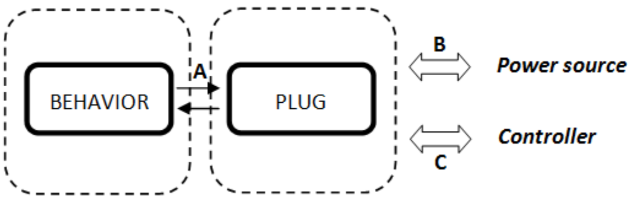

Figure 4 - Agent model

2.3 Structure construction

The construction of the PLUG component (namely the Laws of cooperation) is based on:

- the description of each link to the power line, with a PN to have a matrix formalization of each home automation device
- the assembly of the PNs in a modular way, merging them all in one (B interaction) with a specific tool developed, to include the elements in the system. So we can have the PNs fire control in a single net representative the entire HAS.

Each component in the BEHAVIOR side has a corresponding one tin the PLUG side. Watching the PN realization of a plug is possible to understand how the A, B and C interaction showed in Figure 2 work.

2.3.1 Petri Nets

The management of HAS request to impose a desired behavior, like limit maximum energy consumption or prevent certain situation. This project propose to use the PN theory like modeling formalism to describe the HAS structure (event driven characterization). The need to impose specific laws to each household appliance, is satisfied using the instruments for behavior control available in the PN theory, like GMEC (Generalized Mutual Exclusion Constraints) and inhibitor arcs. An aim of this project is to use the algebra underlying the PN to simplify the model and constrain problem. In this work the PLUG components are modeled with simple PN. Than all the subnets are merged, using the push-out procedure described in the next paragraph, to create the environment structure represented by a single PN. The constraints imposed by the PN about its own state evolution, help to implement the interface with the real wiring. The representation of each watt of energy with the flow of tokens is the way choices to integrate the PNs with the Simulator.

In case of a conflict among appliances, the control system can assign priorities based on own implemented strategy and distribute accordingly the available energy. The development of efficient policies for assigning priorities has to takes into account the peculiarities of the tasks assigned to the single appliances and the preferences of the human user.

The simulated part of the behavior in not known by the controller. A control policy to avoid conflicts between appliances does not use this kind of information. An approach that supposes to have a knowledge of the behavior side of each appliance [An anticipation mechanism for power management in a smart home using multi-agent systems, ICTTA 2008] assume an amount of information that is not available in our case, leading to scheduling oriented solutions.

2.3.2 PLUGs aggregation and HAS construction

To have a general description of the PLUG matrix representation are be evidenced sections related to control action. In fact is possible to extend this representation to many control system that may be represented like PN subnets. The place P2 is the only connection between the PLUG and the control system in this representation like PN. P2 represents the possibility to disconnect the plug from the power source and the control action interact whit each PLUG moving tokens in P2. The control action implementation is hide under the gray part of the matrix representation, allowing any customization.

Now to fuse all the agents and the controller together, in order to obtain the set of L, a PNs merge procedure is required.

The push-out procedure that leads to a single global PN starting the single appliance PNs will be now introduced. When dealing with PNs merging issues, we can have two type of PNs: one equal for all the controlled appliances (Figure 3) and another one for the generic appliance. Both type of PNs represent the available energy with the place P0, so after the push-out procedure all the PNs will be connected by the same place.

Will be used an example with m PNs to merge, each one with the electrical resource represented by the place P0, to explain how Incidence, Inhibition an marking matrices can be merged.

There are m Incidence matrix Ck , one for each PN:

$$[C_1] \quad [C_2] \quad \cdots \quad [C_m] = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

and m marking matrices:

$$[M_1] \quad [M_2] \ldots [M_m] = [0 \quad 33]$$

Analyzing the shared place of the simple PNs with each other PNs (information contained in the arrays Cij which indicate the places of the PN “i” shared with the PN “j”) is possible to identify all the place with some sharing:

$$[C_{12}] \quad [C_{21}] \quad [C_{13}] \quad [C_{31}] \quad \ldots \quad [C_{ij}] = [1]$$

The number of the matrices Cij is the number of the PN 2-combinations:

$$\frac{m!}{2!(m-2)!}$$

After the previous analysis is known in our case that the number of place with some share in 1 for each PN.

In all the PNs to merge the place to share is P1 indentified by the first row of each incidence matrix Cij ,so Cij = 1 $\forall$ i and j. To use this place like fusion point we have to remove this one from each PN and create a new one to connect with all the original PNs. In this way we is possible to remove from the Incidence matrices and marking vectors all the shared elements information. the new Incidence matrices are named Ck- and the new marking vectors are named Mk- :

$$[C_{1-}]_, \quad [C_{2-}] \quad , \ldots , \quad [C_{m-}] = [0 \quad 1]$$

$$[M_{1-}]_, \quad [M_{2-}] \quad , \ldots , \quad [M_{m-}] = [33]$$

Removing the shared row from the matrix representation of PLUGs are build another “m” sub-incidence matrices Ci,r containing the removed row form the Ci matrix that summarizing the transition connection about shared place whit each original PN:

$$[C_{1r}] \; , \; [C_{2r}] \; ,\ldots , \; [C_{mr}] = [1 \quad 0]$$

The marking of the shared place is contained in the vectors Mij which indicate the marking of the places about the PN “i” shared with the PN “j”:

$$[M_{12}] \quad [M_{21}] \quad [M_{13}] \quad [M_{31}] \quad \ldots \quad [M_{ij}]$$

Using the arrays Mij is build a vector Mr 1x1 that indicates the marking of shared place.

With the matrices and vectors described above is possible to merge the “m” original PN in one, building as show below the matrix Cfusion (the indices matrix of the merged PN)

$$\begin{bmatrix} C_{1-} & 0 & 0 \\ 0 & C_{2-} & 0 \\ 0 & 0 & C_{3-} \\ C_{1r} & C_{2r} & C_{3r} \end{bmatrix}$$

and the vector Mfusion (the marking vector of the merged PN):

$$[M_{1-} \quad M_{2-} \quad M_{3-} \quad M_r]$$

Work with the Inhibition matrices is similar and the Inhibition matrix of the merged PN is Hfusion :

$$\begin{bmatrix} H_{1-} & 0 & 0 \\ 0 & H_{2-} & 0 \\ 0 & 0 & H_{3-} \\ H_{1r} & H_{2r} & H_{3r} \end{bmatrix}$$

The procedure can be extended to not equal and not square Incidence matrices and with a place sharing involving more than one place and not the same for each net. An example of merged 3 appliance PNs and on generic appliance is shown in **Errore. L'origine riferimento non è stata trovata.**. After the PNs merge, each agent (wire plug) may detect other agents through the shared places.

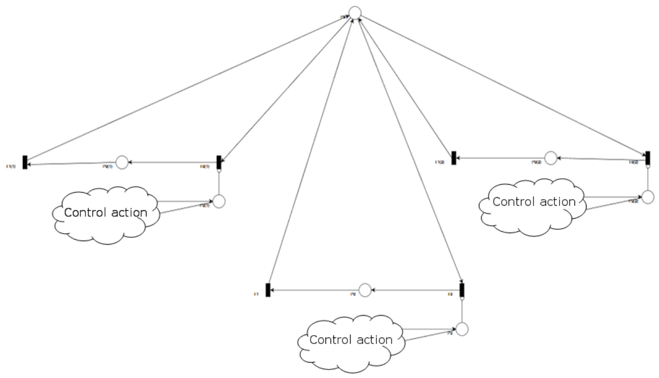

Figure 5 - Merged PNs

3 Analysis and discussion of main results

In the DomoLAB at Information Engineering Department (UNIVPM) are recreated a generic house with read and simulated plug and appliances with three types of plug: actuator, meter, actuator/meter and not controlled (generic plug). Then we extended the modeling procedure to the LeafHouse of Gruppo Loccioni - AEA Srl [11]. The modeling result is reported in *Figure 6* and *Figure 7*.

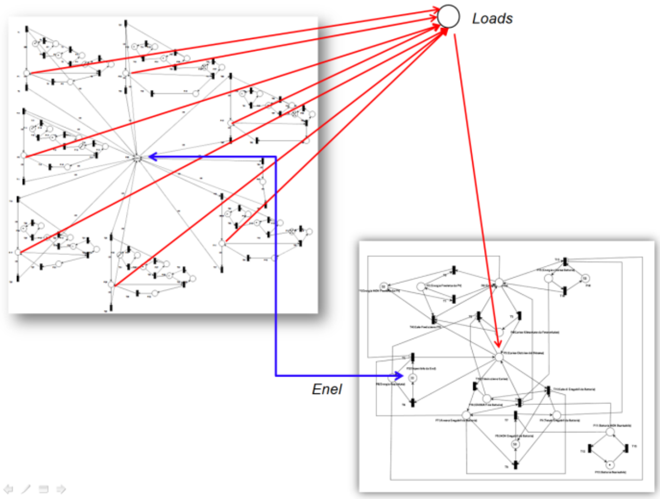

Figure 6 - Apartment's PN

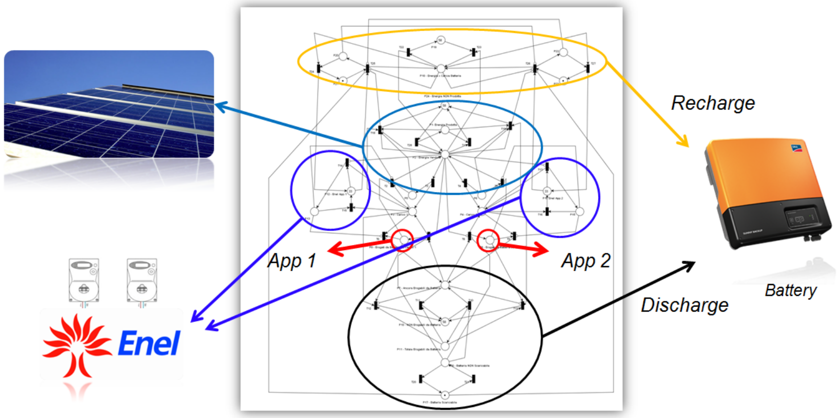

Figure 7 - LeafHouse's PN

4 Conclusions

This work led to the development of a tool for simulation / emulation (software HAS-Sim) modular and effective, able to simulate a wide variety of situations, providing a tool that allows you to quickly build even complex automation structures, complementing each other models of elementary home automation elements. The goal has been reached to allow the simulation of entire month in a few minutes, making it possible to identify quickly the configuration best suited to the needs of users, testing in advance the energy-management policies that are to be taken. It 'been verified that the approaches based on the UML theory of modeling and finite state systems (such as PN) were functional in the characterization of elements such as those that are in the home automation systems by making available at the end of the work a tool to study them.

References

[1] "bticino My Home line," in *www.international.bticino.com*.

[2] "i-bus EIB/KNX from ABB," in *www.abb.com*.

[3] G. Conte, D. Scaradozzi, A. Perdon, and G. Morganti, "Parameter tuning in distributed home automation systems: towards a tabu search approach," in *MED*, 2008.

[4] G. Conte and D. Scaradozzi, "An Approach to Home Automation by means of MAS Theory," *Chapter 15 of Modeling and Control of Complex Systems*, 2007.

[5] G. Morganti, A. Perdon, G. Conte, D. Scaradozzi, and A. Brintrup, "Optimising home automation systems: a comparative study on tabu search and an evolutionary multi-objective algorithm," in *MED*, 2009.

[6] G. Conte, D. Scaradozzi, A. Perdon, M. Cesaretti, and G. Morganti, "A Simulation Environment for the Analysis oh Home Automation Systems," in *MED*, 2007.

[7] G. Conte and D. Scaradozzi, "Viewing home automation systems as multiple agents systems," in *Multi-agent system for industrial and service robotics applications - RoboCUP2003*, 2003.

[8] G. Morganti, A. Perdon, G. Conte, and D. Scaradozzi, "Multi-Agent System Theory for Modeling a Home Automation System," *Lecture Notes in Computer Science*, vol. 5517, pp. 585–593, 2009.

[9] S. Abras, S. Pesty, S. Ploix, and M. Jacomino, "An Anticipation Mechanism for Power Management in a Smart Home using Multi-Agent Systems," in *ICTTA*, 2008.

[10] "National Instruments - LabVIEW," in *www.ni.com/labview*.

[11] Loccioni, "LeafHouse," in *http://energy.loccioni.com/sustainable-community/leaf-house/*.