Instrumentation and Measurement
Technology Conference – IMTC 2007
Warsaw, Poland, May 1-3, 2007

A Smart Distributed Measurement Data Management System for DSM

A. Amato[1], V. Di Lecce[1] and V. Piuri[2]
[1] DIASS - Politecnico di Bari - V.le del Turismo 8, 74100 Taranto – Italy
[2] DTI - University of Milan - Via Bramante 65, 26013 Crema (CR) – Italy
[email: a.amato@poliba.it, dilecce@poliba.it, piuri@dti.unimi.it]

Abstract ***– This work proposes a model of an intelligent short term demand side management system (DSM) based on a distributed measurement and management data system. The system is designed to avoid peaks of power request greater than a given threshold and to give maximum comfort to user.***

The DSM problem is modeled as a multi objectives scheduling problem and it is solved using a metaheuristic approach based on a multi agent system.

The proposed system is composed of a distributed network of processing nodes (PN). Each PN hosts one agent and it is able to manage a single node of a distribution network allowing or disallowing it to supply power. Each agent reacts to a new critical condition entering in competition with the others to gain the access at a shared limited resource. As the results shown the proposed system can be the consumer's key to take advantage of a DSM program automatically.

I. INTRODUCTION

The deregulation of the power market, the introduction of new technologies in producing and distributing energy, led to the research for smart energy management systems.

Analyzing the daily load profile of a generic user, it is clear that the power request varies during the day and that there are one or more peaks of request in specific hours. From the point of view of an energy provider, this peaks should be avoided or smoothed to meet the characteristics of power stations.

Over the last few years Demand-side management (DSM) systems are becoming very common in many countries. Demand-side management (DSM) refers to measures implemented by utilities that modify end-use electrical energy consumption, avoiding sudden peaks of request and scheduling the energy consumption in the hours in which the tariffs are the cheapest [1]. Such programs use mechanisms like interruptible load tariffs, time-of-use rates, real-time pricing, direct load control, and voluntary demand response programs.

A DSM system can be modeled as a multi-objective scheduling problem where the resource to be shared is the energy, the consumers are the devices and the objectives are:

1. avoid sudden peaks of request
2. schedule the energy consumption in the hours in which the tariffs are the chipset
3. give maximum comfort to user, namely satisfy as many requests as possible spending the minimum time (and/or money).

There are already simple DSM-solutions with programmable timers for each device, but they cannot be used for more complex problems and they are essentially static.

In [2] the author applied some artificial intelligent techniques to implement a DSM system for energy management in building automation. This DSM is based on a parallel genetic algorithm. As the same author highlights, this solution has some critical points such as the high execution time (required by the genetic algorithm) and the high quantity of data to transfer among processing nodes.

This study proposes a multi agent distributed load management system that is able to monitor and control complex power systems with limited power capability. The experiments carried out in this work deal with the Italian power provider. Indeed, it offers contracts with tariffs that rise as the power consumption exceeds some thresholds. From this point of view, the limit to the power capability of power systems has an economic nature. In other words, having a power consumption less than a given threshold means having a reduction of production expenses.

In this work this threshold will be called "critical load" and an overall power request greater than this value will be called "critical condition".

Looking at the proposed system as at a multi-objective scheduler, its objectives are those aforementioned.

These objectives are achieved through the competition of more agents. Each agent handles a socket tap. The characteristics of proactivity of agents and their communication ability give to the proposed system a great scalability. The proposed agency utilizes as knowledge base the set of distributed measurements done in each socket tap.

This DSM System can be used in large business facilities, machine halls, private households and wherever electrical devices are in use.

The proposed system was modeled using a simulation realized by means of Matlab Simulink®.

This work is organized in five sections. Section II introduces the multi-object scheduler. Section III describes the proposed system while section IV presents the results obtained, finally conclusions and future works are included in section V.

II. PROBLEM OVERVIEW

Scheduling is the arrangement of entities (people, tasks, vehicles, lectures, exams, meetings, etc.) into a pattern in space-time in such a way that constraints are satisfied and certain goals are achieved [3].

Constructing a schedule means finding an arrangement of time, space and other limited resources considering some constraints. The constraints are relationship among the entities to be scheduled that limit the construction of the schedule.

There are two kind of constrains: hard and soft. A hard constraint can not be violated under any circumstances while a soft constraint can be violated. Each solution that satisfy all the hard constraints is called feasible. Each feasible solution should satisfy as many soft constrains as possible. For each soft constraint not satisfied a penalty is applied to the solution. From this point of view, the scheduling process can be considered as an optimization problem in which the best feasible schedule (the schedule solution with the lowest number of penalty) is found or as a search problem in which any feasible schedule is retrieved.

In real-world problems, the solution spaces of these problems are huge and exploring them often require the human intervention to bias the search towards promising regions. Other promising automatic and semi-automatic approaches are: Pareto optimization techniques [4] and metaheuristic techniques. A metaheuristic is "an iterative master process that guides and modifies the operations of subordinate heuristics to efficiently produce high quality solutions" [5]. Example of metaheuristic approaches are: simulated annealing, genetic algorithm, neural network, etc.

The class of scheduling problems includes a wide variety of problems such as machine scheduling, events scheduling, energy scheduling and many others.

In this work is proposed an agent based multi objectives scheduler applied to energy scheduling problem (DSM). The targets are: avoid sudden peaks of request, schedule the energy consumption in the hours in which the tariffs are the chipset, give maximum comfort to user, namely satisfy as many requests as possible spending the minimum time (and/or money).

III. SYSTEM OVERVIEW

The proposed system is a distributed data measurement and management system. These characteristics are intrinsic in the nature of the system, indeed it is composed of a distributed network of processing nodes (PN) each one connected to a socket tap. Each PN hosts an agent and an acquisition system to measure the power requests of the device connected to the socket tap. Currently the system introduces a constrain: for each instant of time, it is possible to connect only a device to each socket tap.

An important issue of concern is the modeling of the load for each device, indeed there are devices with a constant load (TV, hairdryer, ecc.), on/off devices (electric oven, refrigerator, ecc.), and devices with variable load (washing machine, dishwasher, etc.).

In the proposed system, were considered some uninterruptible load requests (lights in an house or other very important devices in other environments). These loads were modeled as hard constraints for the distributed scheduler.

Each agent uses the measurement system to create an own knowledge base of the device connected to its socket tap and it handles a single socket tap allowing or disallowing it to supply power.

Each agent knows the load curve of the device connected to its node of distribution and those of the active devices connected to all the other PNs. These load curves are used by the system to predict critical conditions and avoid them.

At each instant of time, the load curves of all the active devices are stored in a table (table of loads) shared among all the agents. Another shared information among all the agents is their own state concerning a given critical condition.

The system is a short term power management system. If the overall power consumption is near to the critical load and a new device requests energy making the limit exceed, the system will react to this event searching for a new configuration in which some devices are switched off.

A fundamental constrain for this system is that not all the devices have to react at each event, but as many as there are necessary to avoid a critical condition.

A key point is the way in which the system chooses what devices have to be switched off. This choice has to achieve two main targets:

1. the overall power request has to be less than the critical load. On the other hand, the selected choice should have an overall power request as close as possible to this limit.
2. give to user the maximum comfort. This means that the system should satisfy as many user requests as possible.

In order to find the load configuration that achieves these targets, each agent computes three different strategies.

The difference among these strategies is the policy they use to give the priority to each device. The three policies are:

1. User defined priority: user attributes a different class of priority to each device. Currently five classes of priority are defined. Each class can contain twenty devices. The devices belonging to the highest priority class can be never stopped. Each device define runtime the own priority level through a process of negotiation with other devices in the same class.
2. The priority is automatically assigned to each device. The priority level for each device is directly proportional to the power that it consumed in the last period.
3. The priority is automatically assigned to each device. The priority level for each device is directly proportional to the power that it will consume in the next period.

The last two methods are useful to model a behavior like a protected section that starts when a device requires much power. Indeed, typically the biggest loads are requested by devices that are heating something (i.e. a dishwasher that is heating the water). If, for any reason, a device is interrupted in this phase, the results will be a waste of energy due to the loss of heat.

Each agent processes the table of loads in order to predict critical condition, but each PN can influence only the device connected to it. For each method there is a copy of the table of loads and all the actions are performed on this copy. When the negotiation phase is over the table of loads handled by the winner method become the current table of loads for all the methods.

Each agent computes the following algorithm using all the three methods defined above:

1. Search for a critical condition in the next T columns of the table of loads.
2. If a critical condition is found at the column I
   a. For each method:
      i. The agent connected to the device with the lowest priority that has no reacted to this event shifts its load curve of one column starting from the column I
      ii. Repeat (i) until the critical condition is avoided.
   b. Choose the best solution and update the table of loads of the losing methods with that of the winner method.
3. Go to point (1).

The best solution:

- Avoids the critical conditions
- Has the biggest number of devices that complete their load curve in the next T columns. This allows to system to complete a major number of loads and simplifies the management of the remaining loads.
- Maximizes power consumption in the next T columns. This means satisfying as many user requests as possible.

The first property is an absolutely necessary condition but the others are not so. The second property has a priority greater than the third one.

T is a parameter of the simulation that could be critical in the hardware implementation phase of the model because it is directly proportional to the computational load of the system. The results shown in the next section are obtained using T equal to twenty.

IV. EXPERIMENTS

In order to test the system, a lot of experiments were carried out using different loads. As case study was utilized a house and the critical load was 3 kW.

The load curves for each device were sampled using a system based on a LEM LTS series current transducer interfaced with a PC using a Labjack U12 acquisition system. Figure 1 shows a load curve of a dishwasher. It should be noticed that a dishwasher belongs to the class of devices with variable loads, for this reason Figure 1 is a curve that characterizes only a certain program of a dishwasher.

In this phase of experimentation the load curves of the following household appliance have been sampled and collected in a library of loads: dishwasher, washing machine, refrigerator, phone, TV and oven.

The experiments were carried out varying the following parameters:

1. the involved number of devices (between three and five)
2. the start up time of each devices (0'-15')
3. the user defined priority policy.

The results obtained were compared with those obtained using a traditional DSM. This DSM is based on static association of priority to each socket tap. The priority policy is defined by a human.

The main limits of this DMS are that it is static and that it requires a strong interaction with a human. These facts made it unfeasible in complex environments.

In order to compare the results obtained using the proposed system with those obtained using the traditional DSM properly, the policy 1 (that applies a user defined priority policy) of the proposed system adopts the same priority assigned to the traditional DSM.

Table 1 shows a summary of mean results obtained carrying out various experiments varying all three aforementioned system parameters.

The effects of the proposed system are that: the peaks are smoothed, the power consumption is well distributed in the time and, as Table 1 shows, it is 22% faster than a traditional DSM system.

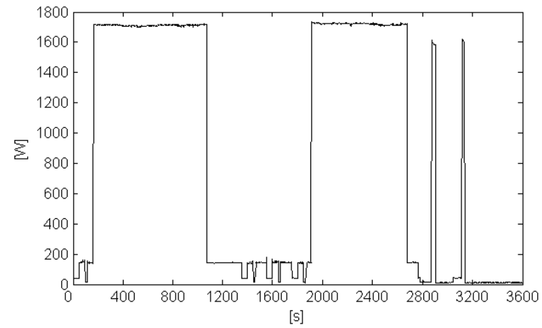

Figure 1 - An example of load curve of a dishwasher

	Average schedule time	Average power consumption
Traditional DSM	177’	2.1 kW
Proposed system	138’	2.5 kW

Table 1 - Comparison among the proposed system and a traditional DSM. The data were obtained carrying out many experiments changing: the number of involved devices (between 3 and 5), the start up time of each devices and the user defined priority policy

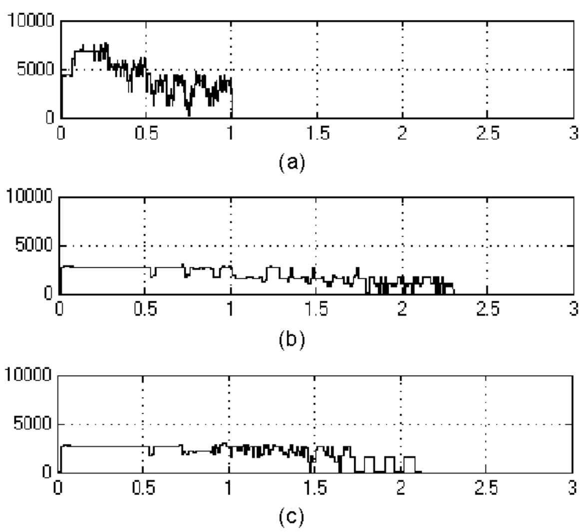

Figure 2 - An example of power request (a) that can not be satisfied without a DSM system. (b) shows how a DSM system based on static priority satisfy this power request. (c) shows how the proposed system do the same. The X axis represents the time expressed in hour

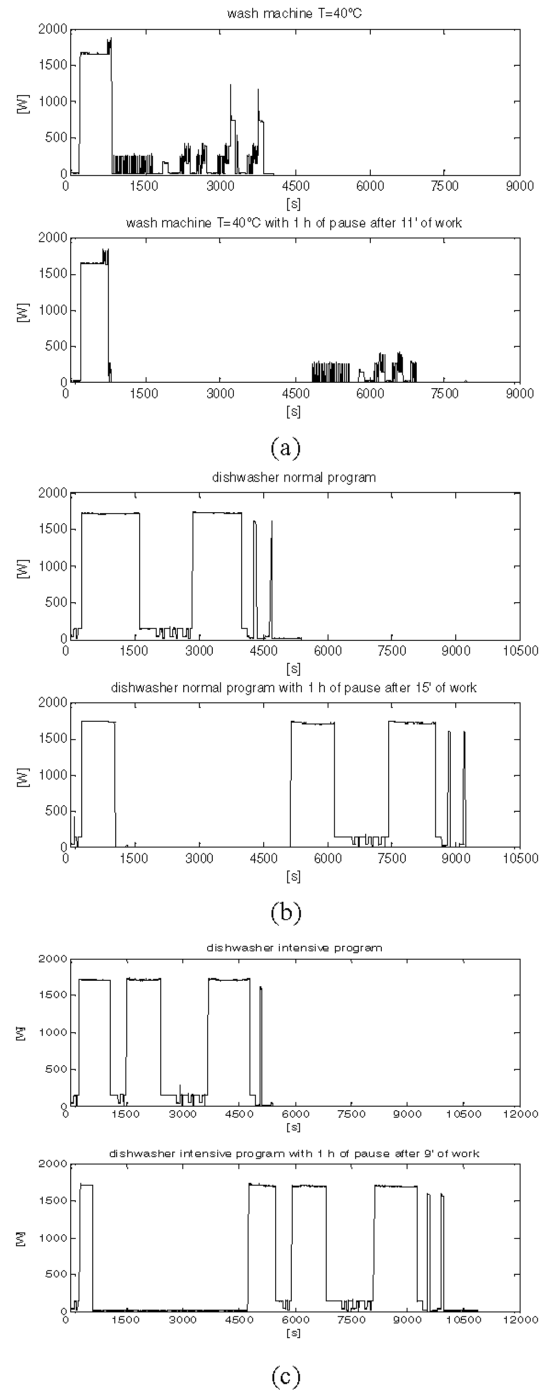

Figure 3 – Example of some load curves without and with power interruptions. (a) washing machine, (b) dishwasher normal program, (c) dishwasher intensive program

Figure 2 shows a heavy power request (Figure 2.a) exceeding for long time the “critical load”. Figure 2.b shows how this request is satisfied by a traditional DSM system while Figure 2.c shows how the proposed system elaborates the same job. In this example the proposed system is 10% faster than the traditional DSM.

A set of experiments was carried out to evaluate the effects of the power interruptions introduced by the proposed system on device programs (see Figure 3).

As expected the behavior of the washing machine is almost unaffected by the power interruption (Figure 3.a). Indeed, this device is programmed as a deterministic finite state machine and once a state is reached (in this example the temperature of 40°C), there is not the possibility to verify if after a power interruption that state is still valid or not.

The same behavior is not observable in Figure 3.b (a dishwasher performing a “normal” program) and Figure 3.c (a dishwasher performing an “intensive” program). Introducing a power interruption of an hour in the cycle of work of a dishwasher causes a mean lost of energy of 12%.

On other hands, it should be highlighted the fact that meanly the power interruptions introduced by the proposed system are smaller than one hour (typically fifteen minutes). Another parameter that play a favorable rule for the proposed system is the high thermal capacity of the water that introduces a big inertia to the waste of energy due to power interruption.

V. CONCLUSIONS

This work proposes a model of an intelligent DSM system. The DSM is modeled using a metaheuristic multi objectives scheduler based on a multi-agent system. This system could be the consumer’s key to take advantage of a DSM program automatically.

Objectives of the system are:

1. avoid sudden peaks of request
2. schedule the energy consumption in the hours in which the tariffs are the chipset
3. give maximum comfort to user, namely satisfy as many requests as possible spending the minimum time (and/or money).

The effects of the proposed system are that: the peaks are smoothed, the power consumption is well distributed in the time and there is a reduction of the running cost.

The proposed system manages the load requests by means of the competition of three different power consumption strategies. Each agent reacts to a new critical condition entering in competition with the others to gain the access at a shared limited resource.

The main advantages of this system in comparison with a system based on the simple user defined priority are:

- It has a great flexibility, indeed user can define a set of devices as uninterruptible loads and he/she has to assign to each device a “class of priority” (each class can contain twenty devices). Each device define runtime the own priority level through a process of negotiation with other devices in the same class.
- It gives a big power reliability because it is able to avoid the critical conditions.
- Its behavior is not predefined by the user, but it is dynamically determined by the competition among various agents.
- The proposed consumption strategies aim to fulfill both user’s whishes (defined by means of the user defined priority) and the intrinsic constrains introduced by the state of work of each device.

Currently load curves sampled using real household devices are used as input to test the system. Since one device, such as a washing machine, can have more than one load curve, future works will add the ability of sampling and storing many load curves into each PN. In this way agents can try to recognize an ongoing power request and they can associate to it a stored load curve. Therefore, the table of loads will be dynamically and automatically built.

From the point of view of a physical implementation of the proposed system, it seems that there are not big technology problems. Small and cheap processors are available and the technologies to enable information interchange among the PNs are available and well tested (for example see: [6, 7])

REFERENCES

[1] Nadel, S. 1992. “Utility Demand-Side Management Experience and Potential - A Critical Review.” Annual Review of Energy and the Environment, 17:507-35. Palo Alto, CA: Annual Reviews, Incentive.

[2] Y.K. Penya, “Last-generation Applied Artificial Intelligence for Energy Management in Building Automation”. Proceedings of the 5th IFAC International Conference on Fieldbus Systems and their Applications, FeT 2003 , July 2003, Aveiro Portugal, pp. 79-83.

[3] Wren A., “Scheduling, Timetabling and Rostering, a Special Relationship?”, In: *Selected Papers from the 1st International Conference on the Practice and Theory of Automated Timetabling (PATAT 1995), Lecture Notes in Computer Science, Vol. 1153, pp. 46-75, Springer, 1996.*

[4] Bagchi T.P., “Pareto-Optimal Solutions for Multi-objective Production Scheduling Problems”, Lecture notes in computer science (Lect. notes comput. sci.) ISSN 0302-9743, pp. 458-471, 2001.

[5] Voss S., Martello S., Osman I.H. and Rucairol C. (eds.), “Meta-Heuristics: Advances and Trends in Local Search Paradigms for Optimization”, Kluwer Academic Publishers, 1999.

[6] Leonard, M. "Networked Controllers talk over power lines", in Electronic Design, pp. 73-79, 17 September 1992.

[7] LonWorks Technology Device Data, Motorola Inc., USA 1995.