

# Development and Interaction between LMS Services and Remote Labs

E. Sancristóbal<sup>1</sup>, S. Martín<sup>1</sup>, R. Gil<sup>1</sup>, C. Martínez<sup>1</sup>, E. López<sup>2</sup>, N. Oliva<sup>1</sup>, F. Mur<sup>1</sup>, G. Díaz<sup>1</sup> and M. Castro<sup>1</sup>

<sup>1</sup> UNED (Spanish University for Distance Education) / Madrid, Spain

<sup>2</sup> Niedax, Madrid, Spain

**Abstract**—Nowadays there is a great number of universities and organizations working in e-learning and i-learning solutions. One of the most well-known is the learning management system or LMS that allows displaying theoretical content in an organized and controlled way. In some jobs and studies it is necessary for the student to get a practical knowledge as well as a theoretical one. To obtain this practical knowledge, the universities and organizations are developing Virtual, Remote and Web labs.

At these moments the LMS and Web labs are working independently. We are studying a new architecture allowing the integration of the LMS with different Web labs. This architecture must allow the students, teachers and administrators to use the services of LMS and virtual lab's features as if they were working with the same software.

**Index Terms**—E-learning, Learning Management System (LMS), On-Line Learning, Practical Knowledge, Theoretical knowledge, Virtual Labs and Web Labs, Web Services.

## I. INTRODUCTION

The improvement in the communication networks (bandwidth, transfer rate, etc.) and the emergence of Internet have induced important changes in the conception of learning. E-learning has given an opportunity to study those people who cannot attend the traditional courses due to geographical problems (living far away from the educational institutions) or schedule problems.

Today a lot of organizations are working in solutions to manage and display theoretical content for their workers or students. This content is displayed in an organized and controlled way (content packets, modules, user profiles) usually under a learning management system being implemented under solutions that have third-party developed, such as WebCT-Blackboard, Moodle, .LRN, etc.

In some cases, the theoretical knowledge is not enough and it is needed a practical knowledge and task-oriented capacitating.

Virtual labs have been developed so that the students can obtain this practical knowledge without having to go to the real laboratory or in some cases to reduce the time devoted to do real practices and to get the same capabilities.

## II. LEARNING MANAGEMENT SYSTEMS (LMS)

A LMS (Figure 1) is a software program that allows displaying theoretical content in an organized and

controlled way. Besides, it must offers a set of features and services, some of them are:

- Administration. It must be able to manage user registrations, roles, assign tutors, logs and student records, user payments, etc
- Content packing. It organizes the content in a hierarchical structure and sets up a mechanism to swap content between different learning management systems. To do it, it's usually used the specification IMS content packaging [1] or the specification SCORM (Shareable Courseware Object Reference Model) [2].
- Synchronous and asynchronous Communication Tools. It must allow collaborative work. So that they can share information, opinions and experiences.
- Evaluation. The tutors and teachers must be able to evaluate the student's progress. Also the students can do test where they can see their progress. To do it, it's possible to use the specification IMS QTI (Question and Test Interoperability).

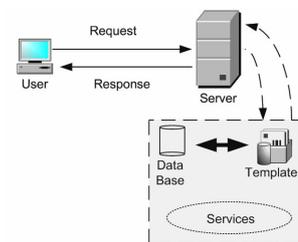


Figure 1. Basic architecture of a Learning Management System.

There are a great number of organizations and universities that have implemented some e-learning solution or Learning Management Systems, such as WebCT, Blackboard, Moodle, .LRN, Sakai, etc., in different approaches, from private companies and closed and proprietary solutions to open software and collaborative oriented applications.

In some cases, (for example .LRN and Moodle), the organizations and universities are allowed to add new services that they have created. These services can be used in some cases, as well or with a few changes, by other organizations. So they allow code reusing and open software solutions. We are working with these kinds of LMS where we can create a middleware that allows interconnecting different remote, virtual and web labs with a LMS without having to make many changes using new defined interconnection services.

### A. DotLRN

DotLRN (or .LRN) is an open source e-learning system (Figure 2) developed at MIT, based on AOLserver [3] and OpenACS [4] collaborative applications.

- AOLserver is an open source, multithreaded web server. This server uses Tool Command Language (TCL) and you can use the AOLserver TCL API.
- OpenACS (Open Architecture Community System) is a toolkit for building scalable, community-oriented web applications. It is the foundation for .LRN e-learning platform.

### B. Moodle

Moodle is an open source software package for producing Internet based courses. It is made up by Modules. Some of them are:

- Assignment module
- Chat module
- Forum module
- Lesson module
- Quiz module
- Workshop module
- Wiki module

We can create new modules and use them with the other existing modules. The Moodle programmers work with PHP and SQL databases as MySQL.

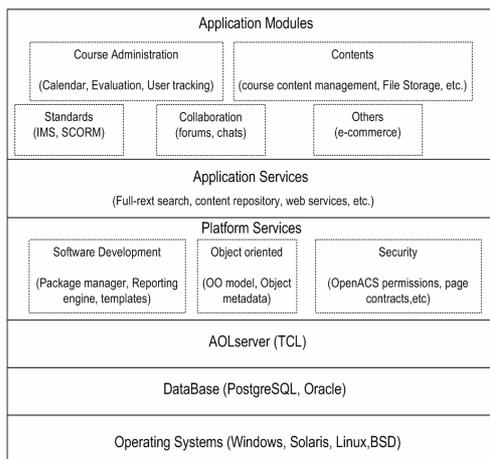


Figure 2. DotLRN architecture.

### C. E-learning standards

Moodle and DotLRN allow the developers to create new packages and applications modules. So, an University or organization can use and modify a new package that has been created by another organization because it is possible to know how the package has been programmed.

We have just seen a brief description of dotLRN and Moodle. Besides being open source platform, they fulfill some e-learning standards or specifications that it allows exchanging content (information, test, courseware, etc.) among different Learning Management Systems. Some of the most important are:

- IEEE LOM specifies a conceptual data schema that defines the structure of a metadata instance for a learning object.

- IMS content packing administration. It's developed by global learning consortium and it promotes the reuse of e-learning content. To get it, a structure it's defined to pack the theoretical contents.
- ADL SCORM. It's a collection of standards and specifications adapted from multiple sources (for example IMS content packing) to provide a comprehensive suite of e-learning capabilities that enable interoperability, accessibility and reusability of Web-based learning content.
- IMS QTI. It's developed by global learning consortium and describes a data model for the representation of question (assessmentItem) and test (assessmentTest) data and their corresponding results reports.

### III. SOFTWARE, VIRTUAL AND REMOTE LABS

A virtual lab is a program that allows a student to carry out his experiments from a PC any time and anywhere without leaving home or from his study place.

There are different types of virtual labs, some summarized in [5]-[8]:

- Software labs
- Virtual web labs
- Remote web labs

In some cases and scenarios these labs can have mixed functionalities. For example, a teacher wants a student to carry out operations that could damage the instrument being in these cases a good solution to use simulation programs besides a remote lab[9]-[13].

#### A. Software labs

They are based on software programs that are being executed in the student's computer (Figure 3).

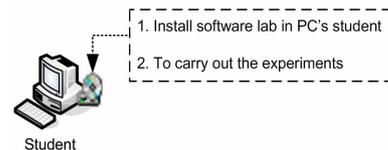


Figure 3. Software Lab.

The student's computer must have the hardware and software requirements and an Internet connection it is not required. The advantages and disadvantages are further detailed

Advantages:

- The students could carry out their experiments at any time from their house. He only has to switch on his computer.

Disadvantages:

- The teacher can't have information about the student's progress (problems that the student has found, time he has spent, etc.)
- Version problems. The software programs are used to be stored in a CD or DVD. In many cases it is necessary to update them in further distribution media.
- They didn't include collaborative tools.
- The students don't work with real instruments but they work with simulation programs.

### B. Virtual web labs

They are programs that use web resources (Figure 4). The student's computer must have an Internet connection with a wide bandwidth connection. The advantages and disadvantages are further detailed.

Advantages:

- The students could carry out their experiments at any time from their house. He only has to switch on his computer and is available for him an Internet network connection.
- In contrast to the software labs a Web lab allows the student to download the updates of the programs from Internet.
- The web server allows the student to work with collaborative tools.

Disadvantages:

- The students don't work with real instruments else he works with simulation programs.
- The students must have a good Internet connection.

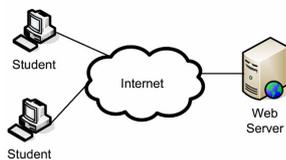


Figure 4. Web Lab.

### C. Remote web labs

A remote lab allows the student to carry out his experiments with real instruments (Figure 5).

It is necessary a driver to manage the instrument. The advantages and disadvantages are further detailed.

Advantages:

- The students can carry out their experiments with real instruments. So that it allows the students to work from their houses although the physical instruments are in a real laboratory whose door is locked.
- The teacher can have information about the student's progress (problems that the student has found, time that he has spent, etc.).
- The web server allows the student to work with collaborative tools.

Disadvantages:

- Every educational organization develops its own solution so it is very difficult to reuse programming code.
- It is necessary to use a good Internet connection, because data, audio and video are usually going to be transmitted.

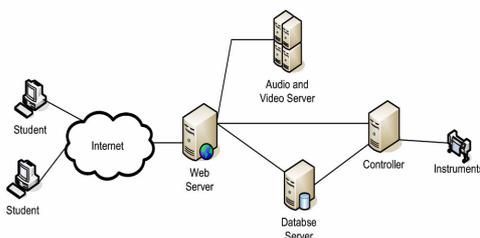


Figure 5. Remote web lab architecture.

## IV. REMOTE WEB LAB ARCHITECTURE

The definition of a Remote Lab was mentioned in the above paragraphs.

In this section we are going to explain a possible structure of a Remote lab being formed by:

- Client or user interface.
- Web server.
- Audio and video server.
- Database server.
- Controller.

### A. User interface

The user interface is what the student is going to see in his PC's screen. The student must perceive as if he was in the physical laboratory. It is very important not to overload the interface or the Internet connection with unnecessary information. The user must be able to decide what he wants to watch or do, of course, within some limits.

In order to create a user interface it is possible to use programming or mark up languages, for example, JAVA, JavaScript, VRML, HTML, XML, etc.

### B. Web server

The web server contains templates with the experiments that the students must carry out with the laboratory instruments. Also it must send commands to the controller and receive the results of the operation from the controller or from the database server.

Normally the web server must attend only a user laboratory request at a certain moment, but it can and must attend other types of request while a user laboratory request is been served.

In our study, the web server will have implemented the learning managements system's services, the templates and the communication with the remote lab.

### C. Audio and video server

The audio and video server sends video and audio data to the user interface. The user could manipulate the web cams, of course within some limits, to see and not lose any detail of the experiments.

The user's Internet connection must be good enough to receive the audio and video data in a quick and simple way. For example, if the video data is displayed too slowly the user could leave the experiment after getting bored.

### D. Database server

This server will store all the information of the student's experiments. The web server can write user data in a table of the database, for example (the time in which the student accesses to the instrument, his name, etc.).

In some remote labs the databases are integrated in the web server.

### E. Controller

The controller is going to send commands to instruments and receive the results of the executed commands. Normally the Controller is communicated with the instrument using a GPIB, an USB, Firewire bus, etc., a hardware data acquisition or a digital signal processor. When the controller receives the results of the

executed commands, they are sent to the web server or to the database server.

## V. ILABS

In the previous section we have seen a general example of remote labs design. So, you can create different remote labs, web labs or software labs with different programming languages, database, etc. In some cases you can have several virtual labs with a different architecture and therefore you have to create for everyone an authentication service, a document storage service, etc.

The MIT and Microsoft are working on the ILab project [14]-[15]. This project is based on allowing to share and to manage online complex laboratory experiments from different virtual labs. To reach this goal, MIT is developing a suite of software tools that make it possible.

The iLabs shared architecture has the following goals:

- Minimize development and management effort for users and providers of remote labs.
- Provide a common set of services and development tools.
- Allow multiple universities with diverse network infrastructures to share access.

### A. Classification of laboratory experiments

Ilab Project defines three types of laboratory experiments:

1. Batched Experiment: the student specifies all parameters that govern the execution of the experiment before the experiment starts. The lab session consists of submitting an experiment protocol, executing the experiment, and then retrieving and analyzing the results.
2. Sensor Experiment: the student usually can not specify any parameters although he may be able to select the particular sensor data that he wishes to receive. Running the experiment consists of subscribing to real time sensor data, usually presented in a graphical user interface such as a virtual strip chart.
3. Sensor experiments frequently have very asymmetric data flows. It takes few bits to subscribe to a sensor, but the resulting data stream from the sensor to the student's client may require a great deal of bandwidth
4. Interactive Experiment: the student typically sets a series of parameters, initiates the experiment, and then monitors the experiment's course, changing control parameters if it is necessary.

The record of an experiment session typically includes the time-stamped control and the sensor data as well as other forms of documentation that may include images or video.

### B. Ilab Architecture

As you can see, there are a great variety of experiments and equipments (MIT has classified these in three groups), then why not look for similar characteristics so that they can be supported by a common software infrastructure.

MIT has designed an architecture that resembles a three-tier web architecture (batched experiments) (Figure 6)

- The first tier is the student's client application that runs as an applet or as a downloaded application on the student's workstation.
- The middle tier, called the Service Broker. It will be backed by a standard relational database such as a SQL Server or Oracle.
- The Service Broker authenticates students, checks on their authorization to contact a particular lab server, accepts an experiment specification from the student's client, and waits for retrieving the result once the experiment completes (batched experiment).
- The third tier is the Lab Server which executes the specified experiments and notifies the Service Broker when the results are ready to be retrieved.

The Lab server doesn't know anything about the students that are using the system, and it only stores experiment specifications and temporarily results.

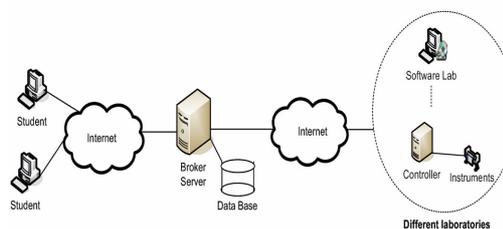


Figure 6. Ilab Architecture

### C. Batched Experiments and Interactive experiments

Now we will show the way in which a batched experiment and an interactive experiment are carried out by the students, (Figure 7)

The batched experiments:

- The user authenticates himself to the Service Broker, which then uses the standard web session technique of a session cookie to remember the identity of the user.
- The Server Broker vouches for the user. The lab server has no need to maintain independent user accounts.
- Service Broker allows the user to log off while his experiment is waiting in a queue. As a consequence, users have no need to schedule their time on the lab equipment.

In the interactive experiments the user must be in communication with the lab server continuously. Therefore if we use the server broker as in the batched experiments then every control message from the students to the lab server and every status or result message from the lab server to the client would require two network hops instead of one.

So, in some case the student and lab server must communicate with each other and therefore it is necessary another service called experiment storage service (ESS) (Figure 8) This service stores XML and binary data from a particular experiment. The administrative information is managed by the Service Broker.

Also you have to make sure while the student is using the laboratory instrument no one can access to them. That is the reason why MIT has developed scheduling service that coordinates reservations to use a lab of different students from multiple campuses.

D. Ilab services and APIs

Finally we are going to enumerate some Iilabs services:

- Authentication and Authorization.
- Storage service
- Scheduling service
- Remote Management

All universities can create their own service broker and share the different virtual labs that have created. Every service has its own specification document where is explained the APIs to communicate with it. Some of them are:

- The Client to Service Broker API describes the external web service API that specifies the connection between the student (e.g., an applet) and the Service Broker.
- The Service Broker to Lab Server is an external web service API that specifies the connection between the Service Broker and the domain-dependent Lab Server.
- The Service Broker Data Storage API describes the internal methods used to access and manipulate experiment records in the Service Broker.
- The Service Broker Authorization API describes the Service Broker extensible authorization architecture, the related methods, and the default authorization policy.
- The Service Broker Authentication API defines an interface to user authentication that enables a modular replacement of the Service Broker authentication mechanism on a campus-by-campus basis.

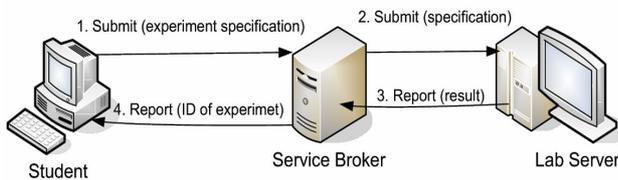


Figure 7. Example of execution of a Batched experiment.

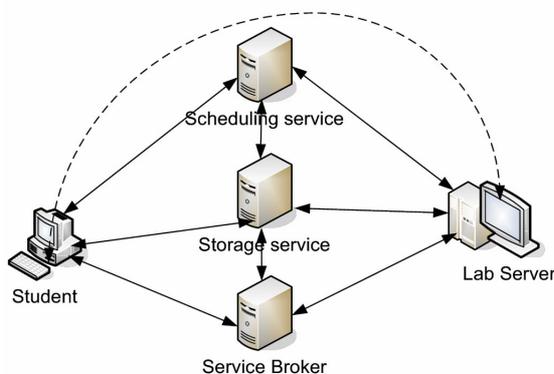


Figure 8. Interactive experiment and Service Broker.

VI. INTEGRATION OF LABS AND LMS SERVICES

Nowadays many organizations are developing virtual labs (software labs, virtual web labs, remote web labs).

These virtual labs are programmed with different programming tools (AJAX, J2EE, MATLAB, LABVIEW [16], etc.) and designed with their own authentication service, communication tools, etc.

Why won't remote lab use the Learning Management System's services? We are working in the development of a middleware. This middleware allows communication between services and data of LMS with different remote lab.

Virtual lab must be an integrated module or service to be used from e-learning solution. This module must manage the laboratory instruments (in the case of a remote lab) and the results that the students obtain of doing their practices.

VII. NEW ARCHITECTURE

The main idea is to create a architecture able to allow the communication among existing services from LMS and the virtual labs (fig 9). To do it, we are working with several open source e-learning platforms as .LRN and Moodle. Also we are developing several virtual labs as Microprocessor Web-lab, a fluid remote lab, etc.

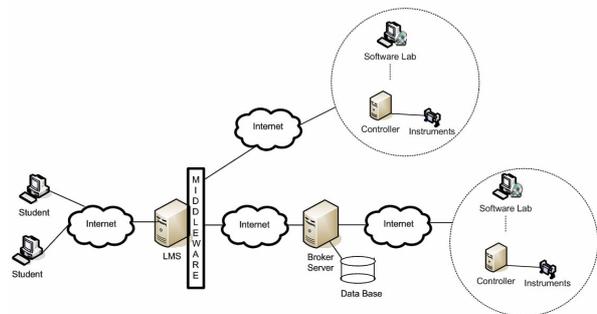


Figure 9. Integration LMS, iLabs and virtual labs.

In this paper we are focusing on .LRN (user interface). To solve this, we are working in five areas:

1. To study the types of existing experiments

It's necessary to establish a common architecture for the different virtual labs. For example the MIT divided great variety of experiments and equipments in batch experiments and interactive experiments.

The idea is find the common aspect (data, architecture, functionalities, etc.) among all experiments.

2. Communication between .LRN and Virtual Labs

User data and experiments data must be sent from the LMS to virtual lab and vice versa. To do this we are working in the development of web services that allows use different virtual labs and can working with the existing LMS's services as user tracking, assessments.

The idea is using SOAP (Simple Object Access Protocol) with the web services. So, the idea of using of web services is to reuse different virtual labs without having to do a lot of changes.

3. .LRN package

The packages in .LRN are applications that work in LMS. For this task it is necessary to develop a new dotLRN package. These packages are formed by a set of directories and files, the most significant are:

- WWW directory where are stored the files with extension .adp and .tcl.
  - ADP files are used to display content. (HTML and TCL)
  - TCL files are executed before ADP files and are used to set up variables, structures, etc.
- SQL directory is where the database files are stored. In this sense, it is important to remind the possibility of using an oracle or a postgresql database.
  - SQL files are used to create the data model that the package is going to use
- TCL directory contains tcl and xql files and are used to create the API package.
  - XQL Files are executed to request information to the data model.

So, we have to create a package with the next characteristics:

- A data model able to store the experiment information, student's actions, etc.
- Creating a communication with web services.

#### 4. Service LRN

The idea is to describe a LMS service with an API that it can be used by the different learning management system's application.

#### 5. User interface

When the user wants to do a practical exercise with the remote lab, he should access through the LMS. Once he has accessed, the LMS must display a template divided in several portlets:

- A portlet contains the experiment documents. These documents will help students to carry out their experiments. These documents could use e-learning standards as IMS or SCORM because of .LRN supports them.
- A portlet that contains links to LMS's collaborative tools.
- A set of portlets that allows the user to write commands and to receive the result of them.
- In the case of a remote lab a portlet that receives the video and audio to see the remote lab.

To design some portlets we are programming them in AJAX. The idea of displaying content without having to reload the page is very important because in the user interface there are another portlets, as communications tools, that mustn't be reloaded.

### VIII. CONCLUSION

This paper describes the idea of designing a project e-learning where different virtual labs can use the advantages of a learning management system (user administration, content packing, etc.) and vice versa. The LMS and the virtual labs must carry out with the e-learning standards like SCORM, IMS or QTI.

It is necessary to build a middleware that allows interconnecting different remote web lab with a LMS

without having to make many changes. The middleware will use the LMS API.

The tendency to standardize in order to be more practical and make good use of the chance of new technology is something that we will coexist with.

### ACKNOWLEDGMENT

The authors would like to acknowledge the Spanish Science and Education Ministry and the Spanish National Plan I+D+I 2004-2007 the support for this paper as the project TSI2005-08225-C07-03 "mosaicLearning: Aprendizaje electrónico móvil, de código abierto, basado en estándares, seguro, contextual, personalizado y colaborativo".

### REFERENCES

- [1] <http://www.imsglobal.org/question/> consulted on October 2007.
- [2] <http://www.adlnet.gov/scorm/> consulted on October 2007.
- [3] <http://www.aolserver.com/> consulted on October 2007.
- [4] <http://openacs.org/> consulted on October 2007.
- [5] Ko, C.C., Chen, B.M., Chen J.: Creating Web-based Laboratories. Ed. Springer, 2004
- [6] García, A., Rodríguez, S., Rosales, F., Pedraza, J.L.: Automatic Management of Laboratory Work in Mass Computer Engineering Courses. IEEE Transactions on Education. IEEE Vol. 48, N° 1, January 2005
- [7] Lang, J., Nungent, G.C., Samal, A., Soh, L.-K.: Implementing CSI with Embedded Instructional Research Design in Laboratories. IEEE Transactions on Education, IEEE Vol. 49, Num. 1. January 2006
- [8] Domínguez, M., Reguera, P., Fuertes, J.J.: Laboratorio Remoto para la Enseñanza de la Automática en la Universidad de León (Spain). RIAI, Vol. 2, Num. 2, April 2005
- [9] Castro, M.: Laboratorios Virtuales para Enseñanza en Internet – Situación y Desarrollo. I Jornadas Tendencias sobre eLearning (TEL 2005). Madrid, 2005
- [10] Candelas, F.A.: Propuesta de Portal de la Red de Laboratorios Virtuales y Remotos de CEA. Documento elaborado para la Red Temática DocenWeb. 2003
- [11] R. Pastor, R. Hernández, S. Ros, M. Castro, Especificación Metodológica de la Implementación y Desarrollo de Entornos de Experimentación. Revista Iberoamericana de Tecnologías del/da Aprendizaje, Vol 1, N° 1. 2006
- [12] Yang, A.: Design of a Distributed Computer Security Lab. Consortium for Computing Sciences in Colleges. 2004
- [13] MIT: <http://icampus.mit.edu/iLabs/Architecture/downloads/>. Queried on: December 2007.
- [14] AOL Server: What is AOLserver, 2007. <http://www.aolserver.com/>. Queried on: December 2007. National Instruments, 2005.
- [15] <http://www.ni.com/> consulted on October 2007.

## AUTHORS

**E. Sancristobal** is with the Electrical and Computer Engineering Department, UNED (Spanish University for Distance Education), Madrid, Spain (e-mail: elio@ieec.uned.es).

**S. Martín** is with the Electrical and Computer Engineering Department, UNED (Spanish University for Distance Education), Madrid, Spain (e-mail: smartin@ieec.uned.es).

**R. Gil** is with the Electrical and Computer Engineering Department, UNED (Spanish University for Distance Education), Madrid, Spain (e-mail: rgil@ieec.uned.es).

**C. Martínez** is with the Research Methods and Diagnosis on Education Department, UNED (Spanish University for Distance Education), Madrid, Spain (e-mail: cmarme@edu.uned.es).

**E. López** is with NIEDEX and collaborates with the Electrical and Computer Engineering Department, UNED (Spanish University for Distance Education), Madrid, Spain (e-mail: eugeniolopezaldea@gmail.com).

**N. Oliva** is with the Electrical and Computer Engineering Department, UNED (Spanish University for Distance Education), Madrid, Spain (e-mail: noliva@pas.uned.es).

**F. Mur** is with the Electrical and Computer Engineering Department, UNED (Spanish University for Distance Education), Madrid, Spain (e-mail: fmur@ieec.uned.es).

**G. Díaz** is with the Electrical and Computer Engineering Department, UNED (Spanish University for Distance Education), Madrid, Spain (e-mail: gdiaz@ieec.uned.es).

**M. Castro** is with the Electrical and Computer Engineering Department, UNED (Spanish University for Distance Education), Madrid, Spain (e-mail: mcastro@ieec.uned.es).

Manuscript received 25 March 2008. Published as submitted by the authors.