

An Experience of CACSD for Networked Control Systems: From Mechatronic Platform Identification to Control Implementation

Cristina Losada, Felipe Espinosa, *Senior Member*, IEEE, Carlos Santos, Manuel Gálvez, Emilio Bueno, *Member*, IEEE, Marta Marrón and Francisco J. Rodríguez, *Member*, IEEE.

Abstract—Continual advances in information and communication technologies (ICT) are revolutionizing virtual education, and bringing new tools on the market that provide virtual solutions to a range of problems. Nevertheless, non-virtual experimentation using Computer-Aided Control System Design tools is still fundamental for future engineers. This paper describes a control course lab element, designed to teach electronic control engineering, in which students use a networked control system to steer and drive an electric car prototype. Multidisciplinary in nature, this lab allows students to take advantage of their previous academic training as they model the plant from an open-loop test, and contrast the simulated behavior with experimental results obtained on the prototype. A goal in this work was that students, across multiple engineering disciplines, should be able integrate the knowledge acquired throughout their degree studies. An evaluation of students' grades, and their perception of the course, was conducted; this confirmed the educational benefits of the approach.

Index Terms—CACSD, computer-aided instruction, computing, control engineering, engineering education.

I. INTRODUCTION

CONSTANT advances in information technologies have resulted in virtual education being highly promoted in academia [1], [2], [3]. Nevertheless, the experimental component is an essential element of engineering students' academic training [4]. The authors of this paper consider that the proliferation of simulation software tools can complement, but not substitute for, practical activities where students can validate theoretical hypothesis, and analyze the differences between the experimental and simulation results.

Several works in the literature take this approach. For example, [5] describes the digital control (zero-pole network) of an analog plant with a PC, using Matlab/Simulink for the design stage, and DAC and ADC converters.

In [6], students control commercially-available tools and prototypes, so that they can focus on the control engineering aspects. The authors defend their use of an "Internet

This work has been developed under the teaching project UAH/EV713 "Metodología docente para la realización de proyectos conjuntos en materias de control electrónico" supported by the University of Alcalá.

The authors are with the Electronics Department, University of Alcalá. 28805. Alcalá de Henares, Spain. (e-mail: losada@depeca.uah.es).

laboratory," which limits students' remote (but secure) intervention to modifying control set-points. Similarly, the studies in [7] and [8] propose the development of remote laboratories for teaching control engineering, using Easy Java Simulations (EJS) and (in [8]) Simulink.

The authors of [9] had planned to use a double PID to simultaneously control the velocity and armature current, of a DC motor in real time, but the actual implementation was carried out using a dsPIC. In [10], the PID control of a LEGO motor system, in real time, was designed using the software tools Matlab/Simulink. The freely-available software nxtOSEK [10] was used as interface between Simulink and the LEGO prototype.

This paper presents a project-based learning (PBL) proposal that has been implemented in the teaching of electronic control systems across various engineering degree subjects at the University of Alcalá, Spain. Integrating software and hardware tools, the work inculcates technical and engineering training skills, including:

- *Multidisciplinarity*, enhancing students' previously-acquired knowledge across fields such as mathematics, physics, linear systems modeling and simulation, electronics, communications, telematics and informatics.
- *Design of a rapid prototyping environment* from commercial permanent magnet DC (PMDC) motors, a general purpose electronic interface configured by the student, general purpose design tools (Matlab/Simulink) and a free operating system for real-time computer control of machine tools (Linux CNC [12]).
- *Development of a networked control application* in which the mechatronic platform and the controller (PC) are linked via Ethernet, which requires students to perform typical stages in a control engineering process: experimental identification of the model plant, design and simulation of the control solution, automatic code generation with Real-Time Workshop (RTW), and design validation from the experimental results, Fig. 1.
- *Modeling, analysis, design, validation and implementation for real control problems*, establishing a close relationship between theory and practice, and comparing simulated and experimental results and addressing any disagreements in these.

From the point of view of training future engineers, the PBL

methodology implemented encourages active and collaborative learning, improves the teaching-learning process and increases students' motivation and capabilities, in a renewed version of the well-known Kolb learning style [11].

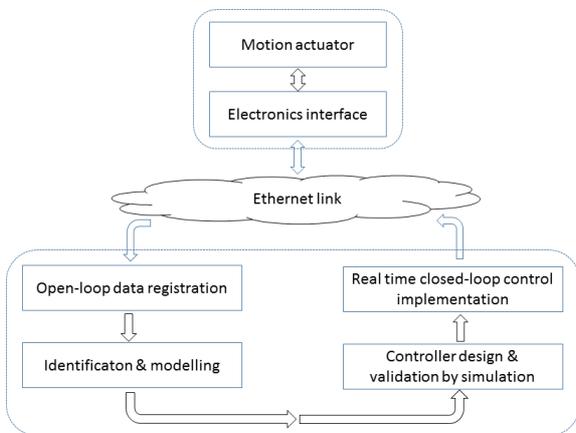


Fig. 1. Recursive process for the identification and remote control of the plant, based on a motion actuator.

The main new contributions of this work are that:

1. Students take part in configuring the Computer Aided Control System Design (CACSD) tools and electronic interface, as well as in the design, simulation and experimental validation of the control system.
2. Students verify the advantages of remotely controlling the plant under study, via an Ethernet link, using the proposed configuration. Several alternatives are explored, Fig. 2: simultaneous control of multiple plants from one remote PC; non-simultaneous control of one plant from multiple PCs sharing the same local area network; or a combination of these. Students can also evaluate the secondary effects of the communication channel, such as delay or latency.
3. The emphasis is on the modeling of the linear and nonlinear components of the real plant. The mechatronic platform is previously identified from an open loop experimental test, taking into account that inputs and outputs are acquired from the remote controller.
4. CACSD tools are used: a) the Linux CNC operating system [12] that includes the real-time application interface (RTAI) [13], suitable for real-time control of industrial applications, and b) Matlab/Simulink and its toolboxes, especially the Real-Time Workshop (RTW).

Student motivation is an important factor in the learning process. This consideration motivated the choice of practical activity: the remote control of steering and driving an electric vehicle prototype in which the motion actuators are two PMDC gear motor blocks, each one with encoder and electronic interface, Fig. 3.

The paper is organized as follows. Sections II and III, describe the development environment. Section IV explains the servo-system for any of the movement actuators. The main academic results are shown in Section V, and the conclusions and future work are described in Section VI.

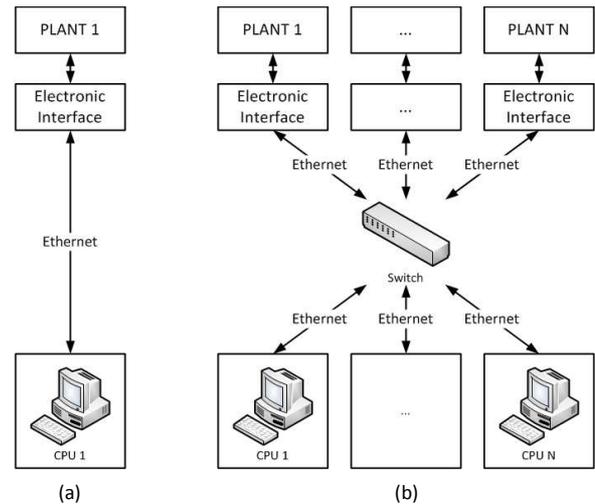


Fig. 2. Alternatives for Ethernet connection between CPUs and plants under control. (a) Direct connection. (b) Distributed connection.

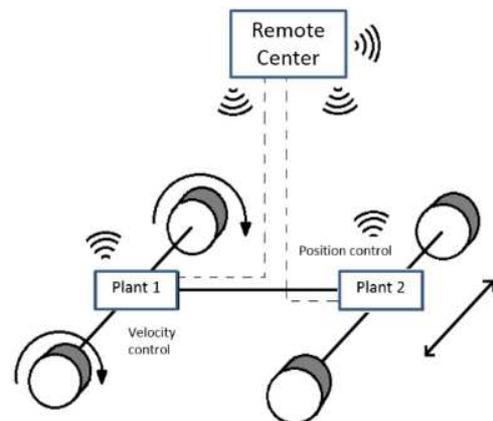


Fig. 3. Example of application of the CACSD proposal to steering and driving an electrical vehicle.

II. MECHATRONIC PLATFORM

Control engineering labs typically use commercial plant demonstrations, such as the inverted pendulum [13], the pendubot [15], the helicopter [16], the quadrotor [17], and the like. This allows the design effort to be focused on the algorithm solutions, which are generally complex. However, the authors consider that in training undergraduate students, their previously acquired knowledge and their ability to apply low-cost resources should be integrated and strengthened.

With any of the configurations shown in Fig. 2, the basic plant to be controlled comprises an electronic interface connected to an Ethernet network and a motion actuator. The first is based on an Arduino One [18] evaluation board, a low-cost open-source electronic platform. The motion actuator is a commercial PMDC gear motor with magnetic encoder. Fig. 4 shows a general block diagram of the proposed hardware architecture, which includes all the elements needed to control the electrical motor, both in angular velocity and position.

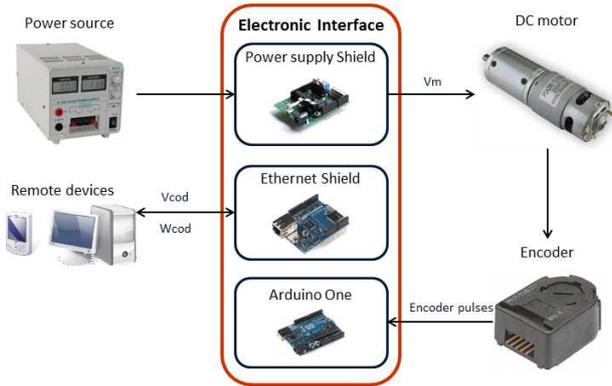


Fig. 4. General block diagram of the remote controller and the plant under study.

The electronic interface is composed of two low-cost open-source boards: Arduino One and the Ethernet shield, and a power supply shield designed ad-hoc for this application. These are briefly explained below.

A. *Arduino One* is a microcontroller board, based on the ATmega328 [18]. It is responsible for controlling communications between the remote device and the mechatronic platform, through the Ethernet interface, and for sampling the pulses from the magnetic incremental encoder.

B. *Ethernet shield*. This allows the Arduino One board to be connected to a 10Base-T (10Mbps) or 100Base-T (100Mbps) Ethernet network. It provides a standard RJ-45 Ethernet jack, and supports both TCP and UDP protocols.

C. *Power supply shield*. This has three parts:

- The power supply and system protection provides a 5V DC regulated output for powering the Arduino One and the Ethernet shield.
- The power stage generates a PWM (Pulse-Width Modulated) signal for driving the PMDC motor.
- The encoder input interface adapts the encoder output levels to the Arduino needed logic levels through two pull-up resistors.

This configuration offers a low-cost electronic interface that can apply different motion controllers to the electric motor. This simple mechatronic platform can test various control techniques, such as classical PID, servo-systems based on full-state feedback, observers, fuzzy and neural controllers, etc.

III. SOFTWARE TOOLS INTEGRATION

For CACSD, several specific software tools are available, such as Scilab [18], [19], [21], Slicot [22], [22], [24], or Niconet [24], [25], [27], but the most widely used in the academic and professional world is Matlab/Simulink [28]. Working with Matlab/Simulink facilitates the identification, modeling and simulation of the plant under control. Both the discrete-time linear components (including the deterministic channel delay) and the non-linear ones (saturation, dead zone and input and output quantification effects) can be easily evaluated. From the control point of view, a lot of resources are available in their libraries (toolboxes) for linear (root locus, bode, observers, optimal control, predictive control,

etc.) and non-linear (Lyapunov based control, fuzzy control, neural control, etc.) studies. Also, the Real-Time Workshop toolbox allows real-time executable code to be generated from the graphical solution validated with Simulink, facilitating its implementation on a real target. This capability does not appear to be available in other software alternatives.

The CACSD computer tools proposed include:

- The operating system Linux CNC [12], a software system for machine tools incorporating the real-time application interface (RTAI) for Linux [13].
- Matlab/Simulink and the specific toolboxes and blocksets related to control engineering: systems identification, control systems and real-time workshop (RTW).

In control engineering, being able to create real-time applications from Simulink and run them on dedicated target hardware is a key aspect. To do this, two elements are necessary: a driver block, implemented as an S-function, which allows the communication with external hardware (Arduino board), Fig. 5, and RTW to automatically generate source code and real-time applications from the Simulink model.

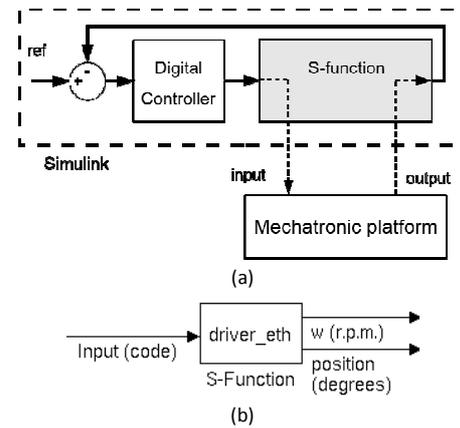


Fig. 5. (a) Diagram of the interaction between the Simulink model and the mechatronic platform through an S-function block. (b) Variables involved in the specific S-function “*driver_eth*”.

The S-function used for the communication (“*driver_eth*” in Fig. 5) has one input and two outputs. The input is for the information coming from the control algorithm to be sent to the platform. The first output provides the actual motor angular velocity (rpm), and a simple correction factor is used to change the units (rpm, rad/s, and degrees/s). The second output provides the motor position (in degrees or radians).

The “*driver_eth*” S-function is also responsible for controlling the Ethernet communication to the mechatronic platform, sending the input code and reading the encoder signals. The position resolution is one count, being the number of counts per revolution equal to the number of encoder pulses multiplied by four. From the motor position, obtained counting these encoder pulses, its angular velocity is also computed in the S-function. To allow control of different plants, the IP address of the Arduino Ethernet shield as well as the feedback sampling time can be easily configured through the S-function arguments.

IV. FROM IDENTIFICATION TO CONTROLLER IMPLEMENTATION

Fig. 6 shows the various stages the students follow to obtain the final driving control solutions. Below, some practical aspects of the lab activity are detailed.

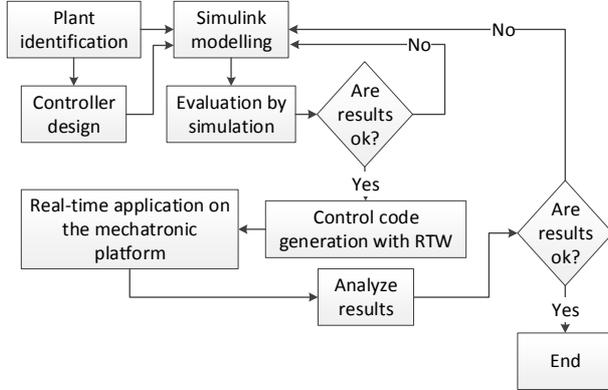


Fig. 6. Block diagram of the design process to be used within the CACSD.

A. Plant Model Identification

Before tackling the control design, the behavior of the plant must be analyzed, identifying its input, output, and internal states and the relationship between them [29].

Here, the intended purpose is the control of a dual mechatronic plant based on a PMDC motor, Fig. 7, emulating the steering and driving motion actuator of an electric vehicle, Fig. 3. This requires the design of velocity and position tracking systems, implemented in a digital device, and connected to each motion actuator electronic interface through an Ethernet link. As a first approach, the design and validation of any of the motion actuators can be done independently, as shown in Fig. 7.

The parameters of the PMDC gear motor and the encoder can be found in [29]. Since the plant model has to describe the actual behavior of all elements linked to the controller - not only the DC gear motor with the encoder but also the communication channel and the electronic interface - the real plant is identified after registering a collection of input-output data in open-loop testing.

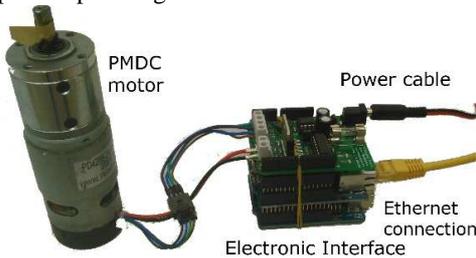


Fig. 7. Elements of the individual plant studied in the experiment.

The plant input is a digital code in the range $[-255, +255]$, and the output can be the motor angular velocity in revolutions per minute (rpm) or the motor axis position in degrees.

Due to the wired Ethernet link between the controller and the plant under study, the channel delay can be considered as a deterministic time-delay. The experimental tests confirm that the delay variability as well as the packet dropout are negligible and do not disturb the control loop stability.

In order to carry out the plant open-loop test, the selected

sample time is $T_s=5\text{ms}$, chosen as a compromise between the time-delay and the transitory response. The complete identified model is shown in Fig. 8. The various elements are described below, and their characteristic values are detailed in Table I.

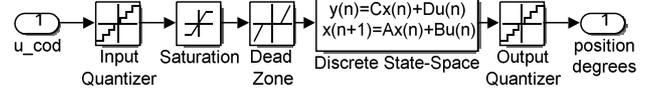


Fig. 8. Linear and non-linear elements of the mechatronic platform model.

Parameter	Value	Units
Dead zone	$[-14, 14]$	code
Input saturation level	$[-255, 255]$	code
Input quantizer	1	code
Output quantizer	11.278 or 0.3355	rpm/count degree/count
Alpha coefficient (α)	0.212	-
Beta coefficient (β)	0.847	-
Static gain	1.355	rpm/code
Time constant	30.2	ms

The model linear components are represented in Fig. 9. Parameters α and β are obtained using the Matlab Identification Toolbox, comparing “*arx*” and “*oe*” structures [29] and choosing the one with lower loose function. T_s is the sample time and K_e is a conversion factor from rpm to degrees/s. Finally, the model state variables are: position (x_1) in degrees, velocity (x_2) in rpm and delayed input code (x_3).

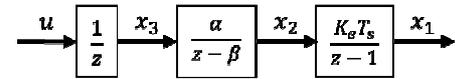


Fig. 9. Linear components of the mechatronic plant model.

The plant was modeled in the discrete time state space; the characteristic matrices for the steering actuator are shown in Eq. (1):

$$\mathbf{A} = \begin{bmatrix} 1 & K_e T_s & 0 \\ 0 & \beta & \alpha \\ 0 & 0 & 0 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad \mathbf{C} = [1 \ 0 \ 0] \quad \mathbf{D} = [0] \quad (1)$$

The nonlinear elements shown in Fig. 8 are: the *input quantizer* modeling the integer value (8 bits) related to the electronic interface input; the *dead zone* modeling the minimum input code that produces the PMDC motion; the *saturation zone* modeling the maximum allowed input code; and the *output quantizer* modeling the encoder counting quantification effect in the output value in rpm (velocity) or degrees (position).

B. Control Systems Design

Next, the velocity and position servo-systems are designed, including a Luenberger observer, which, as the noise related to states and output is not relevant, complies with the objective. Both the position and velocity controllers, Fig. 10, are based on the same control law [31], [32]:

$$u_k = -\mathbf{K}_r \cdot \mathbf{x}_k + \mathbf{K}_i \cdot v_k \quad (2)$$

where u_k is the control vector, \mathbf{x}_k is the plant state vector (with two components in the velocity control system and three

in the position one), \mathbf{K}_r is the state feedback gain matrix, \mathbf{v}_k is the state vector related to error integration and \mathbf{K}_i is the integral gain matrix. As an example, in the position servo-system the mentioned variables size is as follows: $\mathbf{u}_k \in \mathcal{R}^1$, $\mathbf{x}_k \in \mathcal{R}^3$, $\mathbf{v}_k \in \mathcal{R}^3$, $\mathbf{K}_r \in \mathcal{R}^{1 \times 3}$ and $\mathbf{K}_i \in \mathcal{R}^{1 \times 1}$. The gain matrices are calculated by pole placement using any of the methods described in [31], or by applying the available Matlab functions (“place” or “acker”).

C. Simulated and Experimental Evaluation

Once the controller has been designed, for example to obtain an overdamped response, the plant model is replaced with the Simulink driver. Fig. 10 shows the block diagram corresponding to the steering control; a similar one is designed for the drive control.

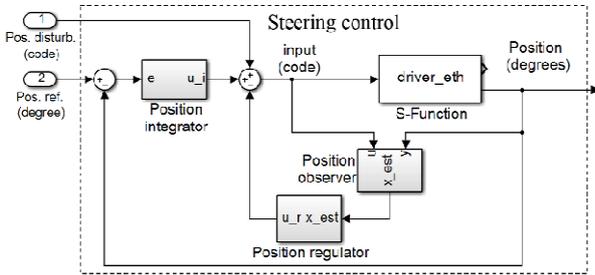


Fig. 10. Block diagram of the position (steering) controller in Matlab/Simulink ready for building the real-time application using RTW.

This diagram is then ready for building the real-time application using the Real-time Workshop (RTW) tool in Matlab. This application is run on the remote PC to test the closed-loop control with the two mechatronic platforms simultaneously.

Next, the results of the position and velocity servo-systems designed for emulating an electric vehicle steering and driving motion actuators, Fig. 3 and Fig. 7, are shown. Fig. 11(a) shows the results obtained from the simulated position control tests, and Fig. 11(b) from the experimental, as well as the system response with external disturbances. As can be seen in the figure, the designed controller achieves null steady-state position error and can compensate for position disturbances. It can also be seen that correct plant modeling minimizes differences between simulated and experimental execution.

Once the mechatronic plants have been independently validated, an improvement allows a better approach to the real electric car prototype. This consists of designing a dependence function between the angular velocity ω_d of the driving actuator and the steering angle θ_s . The student selects the reference driving value ω_0 from the Linux terminal at the remote PC. This angular velocity matches the actually applied to the motion actuator when the steering angle is $\theta_s=0^\circ$. However, the current traction speed is reduced if the steering value deviates from zero according to the expression:

$$\omega_d = \omega_0 \left(1 - \frac{|\theta_s|}{90} \right) \quad (3)$$

This way, ω_d is canceled if the steering reference is perpendicular to the vehicle axis, Fig. 3.

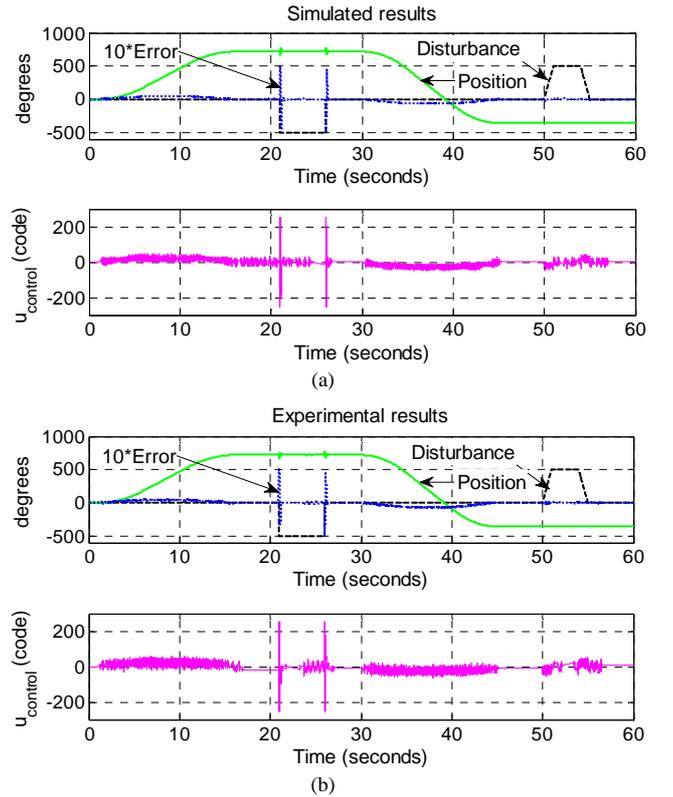


Fig. 11. (a) Simulated and (b) experimental results of the position servo-system shown in Fig. 10. The signals shown are: output, control input, disturbance and tracking error, as labelled.

V. RESULTS

This lab experiment was implemented in three academic years: 2012-13, 2013-14 and 2014-15, as the practical part of a control course. Two hours per week were given to theory lectures, and two hours to working on the lab, for a total of 30 hours of lab work over the semester. The evaluation was based on students’ course grades and their responses to a satisfaction survey. These results were compared with those obtained in the previous three years (2008-2011) in which the lab element was mainly based on simulation practices with virtual plants.

To pass the course, students must demonstrate the following competences:

- Knowledge of multiple disciplines related to the electronic interface (digital and analog electronics, power electronics, mechatronics, etc.), the motion actuator (PMDC gear motor with encoder) and the Ethernet link.
- Choice of the appropriate sample time according to the process dynamics.
- Mathematical modeling from the tests with the real plant, linear and non-linear components (saturation, dead zone and quantization effects) identification in a space state model.
- Design of two tracking systems from specific requirements, for a model electrical vehicle driving (angular velocity) and steering (angular position). In order to achieve this goal, the students have to demonstrate their knowledge of control theory [31], [32] in the state space (servo-system,

observers).

- Validation of the designed controller using CACSD tools (Matlab/Simulink), analyzing the transitory and permanent response of the different variables of interest: tracking error, plant input and output, and internal states estimation.
- Evaluation and understanding of the non-linear effects (dead zone, delay, saturation and quantizers) in the complete plant model.
- Generation of real-time code from a Simulink diagram (control solution and interface with plant).
- Real-time running of the remote controller working in a control loop.
- Critical analysis of the results obtained, and comparison between the simulated and experimental ones.
- Proposal of possible improvements to the design.
- Qualitative analysis (advantages and disadvantages) of the movement (steering and driving) control solution designed, compared to classical PID solutions.

To measure students' grasp of these competences, various assessment methods (written projects, interviews, experiments supervision, etc.) were used. These assessment results are given in Table II.

TABLE II

AVERAGE RATES OF THE STUDENTS ALONG THREE ACADEMIC YEARS FOR THE PREVIOUS AND THE PROPOSED METHODOLOGIES

	Previous Methodology	Proposed Methodology
Outstanding (9 -10)	2.73 %	5.97 %
Good (7 – 8.99)	16.71 %	39.7 %
Pass (5-6.99)	24.63 %	39.8 %
Fail (0-4.99)	21.01 %	8.15 %
Did not take the exam	34.92 %	6.38 %

Fig. 12 shows the percentages of students who passed, failed or did not take the control syllabus final exam, both for students from 2008 to 2011, when the previous methodology was in use; and for students from 2011 to 2015, when the methodology described here was implemented. On average there were about 90 students each year.

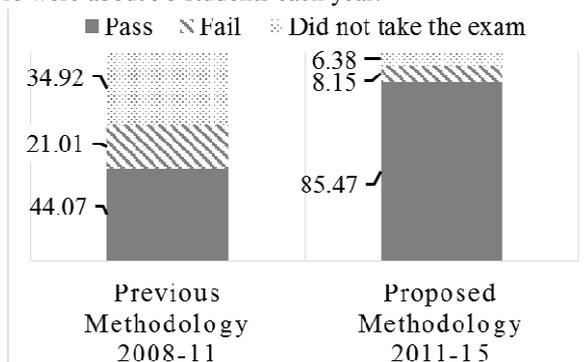


Fig. 12. Percentages for the academic years 2008-11 (previous methodology) and 2011-15 (methodology described here) of students who passed, failed and did not take the final exam.

It can be seen from Fig. 12 that under the proposed methodology the number of students passing the course almost doubled, and that the number dropping the course was drastically reduced.

Finally, students' level of satisfaction was evaluated by

means of surveys managed, provided and analyzed by the University Quality Department. These surveys include different categories to be qualitatively assessed (on a 0-5 scale, where 0 means strongly disagree and 5 strongly agree), related to various aspects such as the role of the professor in the class, the student's academic progress or the effort required to reach the predefined aims. All these aspects are very subjective and highly connected to the student's personal motivation for learning electronic control engineering.

The most significant questions in the survey are:

- Q1. The organization of the course's subject matter is appropriate for learning it.
- Q2. The professor establishes a link between this course and other disciplines.
- Q3. The professor arranges activities to promote participatory learning.
- Q4. The professor promotes activities that give an opportunity to use the theoretical knowledge in practical problems.
- Q5. In general, my degree of satisfaction with the professor's teaching is...
- Q6. The course is relevant and linked to the professional world.
- Q7. There is a coherence between the course's objectives, its contents, the proposed activities and the evaluation methods.
- Q8. The professor evaluates various aspects of the knowledge acquired, such as the concepts, collaborative work, the degree of involvement, personal progress, etc.
- Q9. The professor encourages student participation in the course activities.
- Q10. How would you evaluate your learning experience in this course? (Unsatisfactory / moderately satisfactory / satisfactory).

These items were selected for their relevance to the objectives of this work: Q2 and Q6 are related to multidisciplinary; Q3, Q8 and Q9 are associated with collaborative learning and finally Q4 and Q7 correspond to the relationship between theoretical and practical knowledge.

The students' answers to questions Q1 to Q9 are shown in Table III, which gives the average and standard deviation in students' answers for both the old and new methodologies.

TABLE III
SATISFACTION SURVEYS ANSWERS COMPARISON.

Question	Previous methodology (2008 to 2011)		Proposed methodology (2012 to 2015)	
	Average	Std. deviation	Average	Std. deviation
Q1	3.57	0.98	4.34	0.55
Q2	3.29	0.95	3.93	0.91
Q3	3.29	0.95	3.98	0.99
Q4	3.33	1.03	4.11	0.75
Q5	3.71	0.76	4.22	0.79
Q6	4.14	0.9	4.21	0.95
Q7	3.14	0.9	4.21	0.91
Q8	2.71	0.76	3.57	1.13
Q9	3.86	0.69	4.56	0.50

Fig. 13 plots the averages given in columns 2 and 4 of Table III, showing that students' perception of various aspects of the course has significantly improved.

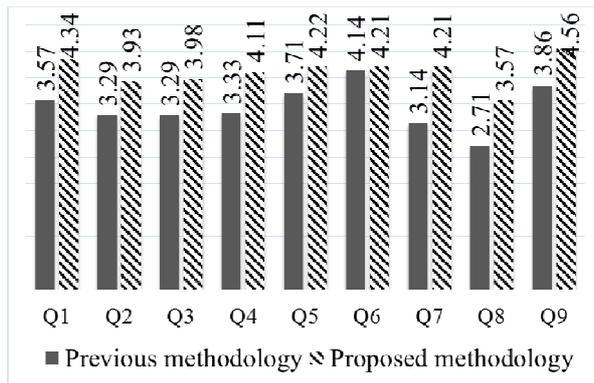


Fig. 13. Average responses to the questions in the satisfaction survey (on a 0-5 scale) for the previous and the new methodology.

For question Q10 (general satisfaction level), Fig. 14 plots the percentage of students considering their learning experience to be unsatisfactory, moderately satisfactory or satisfactory under the previous or new methodology, again showing that the objectives and expectations of the new methodology proposal have been achieved.

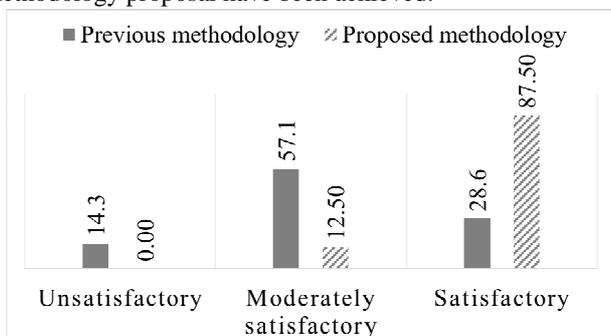


Fig. 14. Percentage response to Q10 asking students' level of satisfaction with the course under the previous methodology and the new one.

VI. CONCLUSIONS

This paper describes an experience of teaching electronic control engineering, focused on the practical aspects of plant identification (including linear and nonlinear elements) and modeling, and on the control design and real-time implementation of a mechatronics platform.

In order to motivate students, the lab topic chosen was to remotely control the steering and driving of an electric car prototype, based on two motion actuators connected via Ethernet to a digital controller. In addition, the lab was designed to integrate the knowledge students had acquired throughout their degree, in the areas of mathematics, physics, electronics, control, communications, and the like, thus further promoting student motivation.

From the control engineering point of view, two servo-systems (angular velocity and position) were designed, with observers, based on the state-space model of the mechatronics plant under study. The Matlab/Simulink resources used, especially the Real-Time Workshop tool, facilitate the identification, control design and implementation tasks, as well as help students to understand the difference between simulated and experimental results.

The results of this effort to establish a project-based

learning methodology in the electronic control syllabus over three academic years were compared to three previous years that applied a methodology mainly based on practicals in which simulations were carried out with virtual plants; the improvement in students' grades and their significantly better perception of this engineering course confirm the educational benefits.

REFERENCES

- [1] P. Moore, A. Thomas, F. Xhafa, L. Barolli, "Context and the virtual campus: Collaboration in tertiary education," in *5th Int. Conf. Intelligent Networking and Collaborative Systems*, Xi'an, INCoS 2013, pp. 266-272.
- [2] S. Vaidya, A. Paranjape, "MOOCs - changing the way of education," in *2014 IEEE Int. Conf. MOOC, Innovation and Technology in Education*, Patiala, (MITE), 2014, pp. 362-365.
- [3] L. Davidovitch, A. Parush, A. Shtub. "Simulation-based Learning in Engineering Education: Performance and Transfer in Learning Project Management," in *J. Eng. Education*, vol. 95, no. 4, pp: 289-299, Oct. 2006.
- [4] Feisel, L. D.; Rosa, A. J., "The Role of the Laboratory in Undergraduate Engineering Education," in *J. Eng. Education*, vol. 94, no. 1, pp. 121-130, Jan 2005.
- [5] T. Vinay, "Computer aided laboratory in digital control" in *IEEE Int. Conf. MultiMedia Eng. Education*, Melbourne, 1996, pp. 253-259.
- [6] W. Dixon, D. Dawson, B. Costic, M. de Queiroz, "A matlab-based control systems laboratory experience for undergraduate students: toward standardization and shared resources," in *IEEE Trans. Edu.*, vol. 45, no. 3, pp. 218-226, Aug 2002.
- [7] A. Barrios, S. Panche, M. Duque, V. H. Grisales, F. Prieto, J. L. Villa, P. Chevrel, M. Canu, "A multi-user remote academic laboratory system," in *J. Computers & Education*, vol. 62, pp. 111 - 122, Mar 2013.
- [8] E. Fabregas, G. Farias, S. Dormido-Canto, S. Dormido, F. Esquembre, "Developing a remote laboratory for engineering education," in *J. Computers & Education*, vol. 57, no. 2 pp. 1686-1697, Sep 2011
- [9] M. Gunasekaran, R. Potluri, "Low-cost undergraduate control systems experiments using microcontroller-based control of a dc motor," in *IEEE Trans. Edu.*, vol. 55, no. 4, pp. 508-516, Nov. 2012.
- [10] Y. Kim, "Control systems lab using a lego mindstorms nxt motor system," in *IEEE Trans. Edu.*, vol. 54, no. 3, pp. 452-461, Aug. 2011.
- [11] M. Abdulwahed and Z.K. Nagy, "Applying Kolb's Experiential Learning Cycle for Laboratory Education," in *J. Engineering Education*, vol. 98, no. 3, pp. 283-294, Jul 2009.
- [12] Linux CNC [Online] Available: <http://www.linuxcnc.org/>, [Accessed: Jun-2015].
- [13] Real-Time Application Interface for Linux, [Online] Available: <https://www.rtai.org/>, [Accessed: Jun-2015].
- [14] S. Boonto, H. Werner, "Closed-loop system identification of LPV input-output models: application to an arm-driven inverted pendulum," in *47th IEEE Conf. Decision and Control*, Baltimore, CDC 2008, pp. 2606-2611.
- [15] R. O'Flaherty, R. Sanfelice, A. Teel, "Hybrid control strategy for robust global swing-up of the pendubot," in *American Control Conf.*, Seattle, 2008, pp. 1424-1429.
- [16] Y. Zhai, M. Nounou, H. Nounou, Y. Al-Hamidi, "Model predictive control of a 3-dof helicopter system using successive linearization," in *Int. J. Engineering, Science and Technology*, vol. 2, no. 10, pp. 9-19, 2010.
- [17] L. Dai, R. N. Jazar, "Nonlinear Approaches in Engineering Applications," in Springer-Verlag New York, 2012.
- [18] Arduino One, [Online] Available: <http://www.arduino.cc/>, [Accessed: Jun-2015].
- [19] Scilab [Online] Available: <http://www.scilab.org/>, [Accessed: Jun-2015].
- [20] Desprez, F.; Fleury, E.; Gomez, C.; Steer, S.; Ubeda, S., "Bringing metacomputing to Scilab," in *Proc. 1999 IEEE Int. Symp. Computer Aided Control System Design*, Kohala Coast, 1999, HI, pp.398-403.
- [21] Liao Wenjiang; Nanping Dong; Tongshun Fan, "Application of Scilab in teaching of engineering numerical computations," in *2009 IEEE Int. Workshop Open-source Software for Scientific Computation*, Guiyang, OSSC 2009, pp.88-90.
- [22] Slicot: [Online] Available: <http://slicot.org/>, [Accessed: Jun-2015].

- [23] Sima, V., "SLICOT-Based Computational Tools for CACSD," in *Proc. 2005 IEEE Int. Symp. Intelligent Control, Mediterrean Conference on Control and Automation*, Limassol, 2005, pp. 170-175.
- [24] Sima, V.; Benner, P., "Experimental evaluation of new SLICOT solvers for linear matrix equations based on the matrix sign function," in *2008 IEEE Int. Conf. Computer-Aided Control Systems*, San Antonio, TX, CACSD 2008, pp.601-606.
- [25] Niconet [Online] Available: <http://www.icm.tu-bs.de/NICONET/> [Accessed: Jun-2015].
- [26] Van Dooren, P., "The basics of developing numerical algorithms," in *IEEE Control. Syst. Mag.*, vol. 24, no.1, pp.18-27, Feb. 2004
- [27] Reichlin, L.F., "Free computer-aided control system design (CACSD) tools for GNU octave," in *2013 IEEE Conf. Computer Aided Control System Design*, Hyderabad, (CACSD) 2013, pp.334-339.
- [28] Schmid, C. "Computer-Aided Control System Engineering Tools," in *Control Systems, Robotics, and Automation. Encyclopedia of Life Support Systems (EOLSS)*, Developed under the Auspices of the UNESCO, Eolss Publishers, Paris, France, [<http://www.eolss.net>]
- [29] Transmotec Planetary gear DC Motor (PD4266) and encoder datasheet [Online] Available: <http://download.transmotec.com/eng/dc-motors/planetary-gear/Transmotec-DC-Motors-PD-0.6W-35W-eng-D.pdf> [Accessed: Jun-2015].
- [30] L. Ljung (Ed.), "System Identification: Theory for the User," (2nd Ed.), Prentice Hall PTR, Upper Saddle River, NJ, USA, 1999.
- [31] K. Ogata, "Discrete-time Control Systems", Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1987.
- [32] Ken Dutton, Steve Thompson, Bill Barraclough. "The Art of Control Engineering" (1st edition.). Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA. 1997.

Cristina Losada received her B.S. degree in telecommunications engineering in 2004, and her Ph.D. in intelligent systems advanced electronic in 2010, both from the University of Alcalá, Spain. Since 2011 she has been a Ph.D. Assistant Lecturer in the Electronics Department with the UAH. Her research interests are focused on computer vision, human behavior analysis, intelligent spaces and intelligent sensor systems.

Felipe Espinosa received the M.S. degree from the Technical University of Madrid (UPM), Spain, and the Ph.D. degree from the University of Alcalá (UAH), Spain, in Telecommunication, in 1991 and 1999 respectively. He became an Associate Professor in 2000 in the Electronics Department with the UAH, regularly involved in electronic control and automation subjects in several postgraduate programs. His current research interests include network control systems, event-based control, wireless network systems and sensorial integration applied to intelligent transportation systems and intelligent environments.

Carlos Santos received the B.S. degree in telecommunications engineering in 2010 and the M.Sc. in electrical engineering in 2011, both from the University of Alcalá, Spain. He is currently a Ph.D. student in the Electrical Engineering Department at the same university. His research interest focuses on the field of fusion algorithms, trajectory generation for navigation in mobile robotics and varying-time sampling control techniques.

Manuel Gálvez was born in Madrid (Spain) in 1976. He received his degree in electronics engineering from the University of Alcalá, Alcalá de Henares, Madrid, Spain, in 2003, and de technical engineering degree in industrial electronics from the University of Alcalá, Alcalá de Henares, Madrid, Spain, in 2000. From 2002 to 2014 he was an Assistant Professor with the Electronics Department,

University of Alcalá, Madrid, Spain. His research interests include the areas of control electronics and new MPPTs for photovoltaic systems.

Emilio J. Bueno (S'05–M'06) was born in Madrid, Spain, in 1972. He received the M.S. and Ph.D. degrees in electronics engineering from the Universidad de Alcalá, Madrid, Spain, in 1999 and 2005, respectively. Since 2009, he has been an Associate Professor with the Department of Electronics, University of Alcalá (Spain), and member of the research group "Electronics Engineering Applied to the Renewable Energies." From 2010 to 2013 he was a Vice-Dean of the Polytechnic School of the University of Alcalá, in charge of Electrical Engineering studies. His research focuses on linear control of grid converters and drives, power quality, distributed generation systems and medium voltage converter topologies.

Marta Marrón received the Ph.D. in intelligent systems advanced electronics in 2008 from the University of Alcalá, Spain. From 1996 to 2001 she was a researcher in the Department of Electronics in the same university, from 2001 to 2008 an Assistant Professor, and since 2009 an Associate Professor. She is a member of the GEINTRA research group. Her research interests include multisensor indoor localization (intelligent spaces), scene understanding, human-behavior analysis, computer vision, probabilistic algorithms, embedded and electronics control systems and robotics in general, and personal mobile robots in particular applied to assistive technologies. She has authored more than 90 publications in international journals, book chapters, and conference proceedings.

Francisco J. Rodríguez (S'99–M'00) received the B.Sc. degree in technical telecommunication engineering from the University of Alcalá, Alcalá de Henares, Spain, in 1985, the M.Sc. degree in telecommunication from the Technical University of Madrid, Madrid, Spain, in 1990, and the Ph.D. degree in electronics engineering from the University of Alcalá in 1997. He worked in the private electronic industry for two years. Since 1986, he has been a Lecturer with the Department of Electronics, University of Alcalá, where he is now a Professor. He is the author of more than 142 refereed publications in international journals, book chapters, and conference proceedings. He has directed more than 45 investigation projects funded by public institutions and private industry. His research interests include the areas of control electronics, real-time processing, and embedded systems applied to power electronic systems