We are IntechOpen,
the world's leading publisher of
Open Access books
Built by scientists, for scientists

6,900
Open access books available

186,000
International authors and editors

200M
Downloads

154
Countries delivered to

Our authors are among the

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS

BOOK
CITATION
INDEX

INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

4

Gaining Control Knowledge Through an Applied Mobile Robotics Course

Lluís Pacheco, Ningsu Luo, Inès Ferrer, Xavier Cufí and Roger Arbusé
University of Girona
Spain

1. Introduction

In this chapter the use of an open mobile robot platform as an innovative educational tool in order to promote and integrate control science knowledge is presented. The idea of including applied interdisciplinary concepts is an important objective in engineering education. Future work in Electrical and Computer Engineering education is pointing towards gaining the ability to understand the technical details of a wide variety of disciplines (Antsaklis et al., 1999, Murray et al., 2003). Moreover, experimental developments have helped to breathe life into theoretical concepts found in text books and have thereby greatly changed the educational experience of students (Hristu-Varsakelis and Brockett, 2002). Students react positively to realism, since the models used for such experiments are in general accurately described by some relatively simple differential equations. Within this framework, the challenge of using mobile robots becomes evident. Thus, realistic platforms that incorporate engineering standards and realistic constraints increase student skills and experience through engineering practices.

In this sense, mobile robot platforms can be used as educational tools to promote and integrate different curriculum subjects. Control system world and computer science share a mutual interest in robotics. Therefore, in the educational community, the Robotics Education Lab becomes a resource for supporting courses with an academic curriculum in a broad range of subjects. Some important institutions have developed various teaching activities by introducing the mobile robots as a necessary academic tool in a broad sense in order to extend the student's previously acquired knowledge (Carnegie Mellon University, 2011, Qu and Wu, 2006). In this context, many universities take advantage of mobile robot competitions in engineering education. This allows real world problem projects to be tackled, and fundamental concepts by increasing motivation and retention purposes to be reinforced. Thus, for example, FIRST (For Inspiration and Recognition of Science and Technology) mobile robot contest attracts young people to careers in engineering, technology and science. Robotics competition encourages students to apply knowledge gained throughout their engineering degree, it also offers all students a chance to serve as members of interdisciplinary engineering teams, and introduces both freshmen and sophomores to engineering concepts. Moreover, university curriculum is reinforced by the knowledge gained throughout applied experiences that embrace a wide spectrum of subjects (Wilczynski and Flowers, 2006). The educational and research objectives can also be achieved through the use of configurable, small, low-cost such as LEGO mobile robot kits (Valera, 2007).

In this Chapter, the program and acquired experience of an optional course named "Applied Mobile Robotics" is outlined. The main aim of the this educational course is to integrate different subjects such as electronics, programming, architecture, perception systems, communications, control and trajectory planning by using the educational open mobile robot platform PRIM (Pacheco et al., 2006). The course is addressed to the students of Computer Engineering around the time of their final academic year. As a practical approach, the majority of educational activities of the course are developed in our university labs. Section 1 introduces the teaching framework from a global point of view. Section 2 indicates which community of students is suitable for this teaching activity within the context of our university framework. The robotic platform used and the course program are presented from a wide scope within the multidisciplinary context. Section 3 presents the program related to the control educational content of the course in detail. Attention is paid particularly to describing the different experiments designed in order to fulfill the process of gaining knowledge. Section 4 briefly introduces the student opinions and future work.

2. Educational structure

Spanish educational framework arises from the present European convergence program (European Commission, 2008). In this way, the main challenge of Spain in 2010 was to design and adapt the university degrees studies. The objective was to improve the quality of Education processes by implementing new learning methodologies (Amante et al., 2007). In this work, knowledge integration is proposed by using an open mobile robot. The course is addressed to Computer Engineering students that can take this course to consolidate their curriculum studies. It is an optional subject within the degree of Computer Engineering at the Polytechnic School of University of Girona. However, the course "*Applied Mobile Robotics*" started as special subject in the summer of 2005 (Pacheco et al., 2009). It was proposed by a group of faculty members belonging to the teaching areas of Control and Computer Engineering and sharing a common background that involves the research on mobile robots. Hence, its purpose consists to consolidate, to integrate and to complement different teaching aspects such as electronics, programming, control and perception systems from a practical point of view by using an available robotic platform.

This section outlines the Computer Engineering curriculum and the course schedule. Especially attention is paid to relationships between different curriculum subjects and the "*Applied Mobile Robotics*" course. The mobile robot platform used is also described.

2.1 Course schedule and student curriculum

Students, who want to pursue the course, have previously acquired the basic skills in some fundamental areas such as electronics, programming, control and perception systems. In this way, an open mobile robot is used for linking multiple curriculum subjects from a practical point of view. Significant subjects of the actual university degree curriculum are depicted in Fig. 1.

The subjects of Fig. 1, which have blue background, have relationship with the "*Applied Mobile Robotics*" course. The main related subjects cover the following educational parts:

- *Programming* covers aspects such as: programming methodology and technology (first year); algorithms, data structure and programming projects (second year); user interfaces, multimedia, fundamentals of computing, programming languages and artificial intelligence (third year).

Computer Engineering Degree

First Year

| Math (210 h) | Computer Science (240 h) | Physics (90 h) | Programming (60 h) |

Second Year

| Statistics (60 h) | Programming (140 h) | Computer Science (150 h) | Operative Systems (50 h) |

Third Year

| Networks (100 h) | Programming (200 h) | Computer Science (100 h) | Operative Systems (50 h) |

Fourth Year

Optional subjects

| Applied Mobile Robotics (50 h) | Other optional subjects (400 h) | Final Degree Project (150 h) |
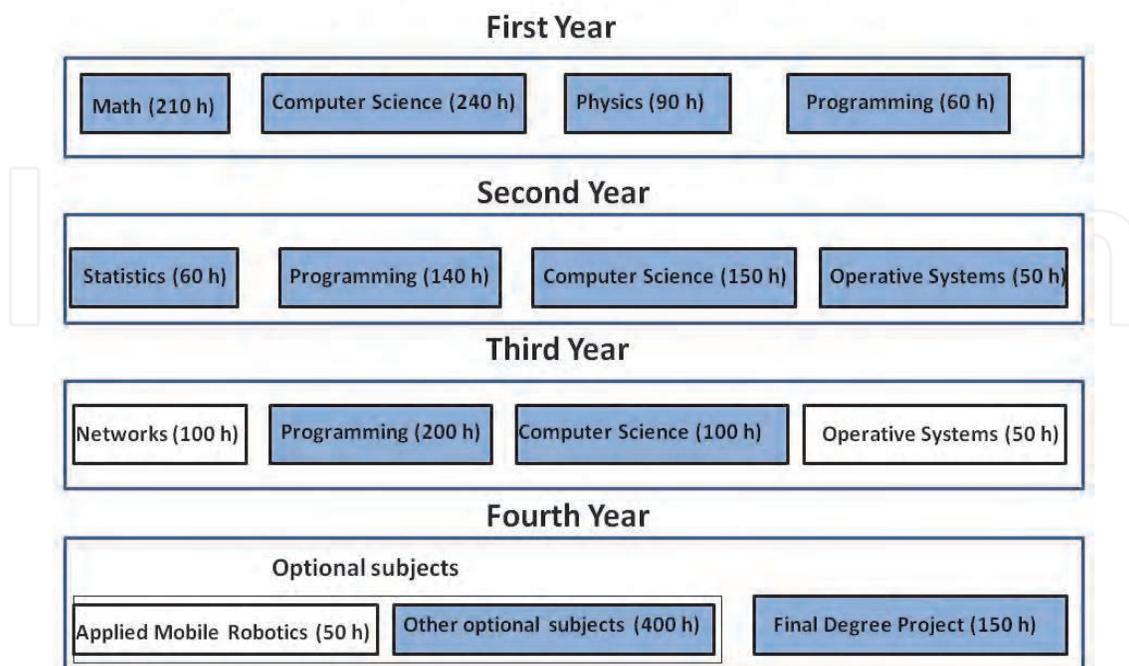
Fig. 1. Computer Engineering degree curriculum.

- *Computer Science* includes issues such as: computer structure and technology (first year); digital systems and microcomputers (second year); architecture of computers, industrial computer science and robotics (third year).

It is noted that topics such as computer vision and control, which are also highly related with the proposed course, are included within the optional subjects. Furthermore, some students are attracted to perform their "*final degree project*" by continuing and developing issues related with some practical aspects of the mobile robotics course. The students who pass the course successfully are awarded 5 credits on their degree which corresponds to 50 hours of teaching activities. It consists of theoretical and laboratory sessions, related to the following different topics: PSE (power system electronics), DPED (design & programming of electronic devices), MCS (modeling and control systems), TPC (trajectory planning and control) and CVS (computer vision systems). The theoretical parts are presented during 11 classroom hours, which are scheduled as follows:

1. Course and platform presentation. Introduction of power drivers (PSE).
2. Hardware description. Digital systems and PLD (programmable logic devices) (Wakerly, 1992). Theoretical introduction of DPED practices.
3. Hardware description. Digital systems and microprocessors (Barnett, 1994). Theoretical introduction of DPED practices.
4. Introducing the MCS block. System identification (Ljung, 1989). Discrete and continuous systems (Kuo, 1996).
5. Developing the MCS block. System reduction, observers and filters (Aström and Wittenmark, 1988).
6. Introducing the TPC block. Configuration space and trajectory following (Schilling, 1990). Fuzzy rules.
7. TPC & MCS integration. Trajectory following. Design of PID controllers.

8. TPC & MCS integration. Trajectory following. Introducing MPC (model predictive control) (Maciejowski, 2002).
9. Introducing the CVS block (Horn, 1998).
10. Practicing within CVS block, color and filters.
11. Final classroom, overall demos.

Laboratory parts are developed during 39 hours that are scheduled as follows:

1. DPED practices with PLD: 4 bits magnitude comparison, PWM generator, design of counters (encoders speed and position).
2. DPED practices with microcomputers: Serial port, I/O ports, A/D converters, PWM signal.
3. MCS practices with the robot PRIM and using MATLAB: excitation of the input system by using PRBS (pseudo random binary signals), the system model using the toolbox IDENT of MATLAB. Discrete and continuous models.
4. MCS practices using SIMULINK environment: reducing the system model and PID speed control.
5. MCS practices using the SISO tools of MATLAB: reduced SISO systems and PID speed control design. Testing of the controllers within SIMULINK environment.
6. MCS practices using the PRIM platform: on-robot testing of the obtained set of speed controllers.
7. DPED practices using PC and the robotic platform: introduction of the high level languages. Communication with the robot by using sockets that include the on-robot available functions (actuators and sensors).
8. TPC practices using the robotic platform: heuristic navigation using sonar sensors.
9. TPC practices using the robotic platform: the odometer system, trajectory following using PID and discontinuous control laws.
10. Time gap for finishing the previously proposed TPC practices.
11. MCS and TPC practices: introduction of MPC software simulation tools, trajectory following with different prediction horizons and trajectories.
12. CVS practices: introducing the software tools, binary images and segmentation.
13. CVS practices: color, filters and tracking of an object.

2.2 The educational open mobile platform

The robot structure, shown in Fig. 2.a, is made from aluminium. It consists of 4 levels where the different parts are placed. On the first level there are two differential driven wheels, controlled by two DC motors, and a third omni-directional wheel that gives a third contact point with the floor. On the second level there is an embedded PC computer, and on the third level specific hardware and the sonar sensors are placed. On the fourth level the machine vision system is placed. The system can be powered by 12V dc batteries or by an external power source through a 220V ac, and a switch selects either mode of operation. The 12V provided by the batteries or the external source of 220V are transformed into the wide range of dc voltages needed by the system. Moreover, the use of a 220V power supply allows an unlimited use of the platform during the teaching activities. The robot has the following sensorial system:

• Two encoders connected to the rotation axis of each dc motor.
• An array of sonars composed by 8 ultrasounds sensors.
• A machine vision system consisting of a monocular camera.

The meaningful hardware consists of the following electronic boards:

- The dc motor power drivers based on a MOSFET bridge that controls the energy supplied to the actuators.
- A set of PCB (printed circuits boards) based on PLD act as an interface between the embedded PC system, the encoders, and the dc motors. The interface between the PLD boards and the PC is carried out by the parallel port.
- A μc processor board controls the sonar sensors. Communication between this board and the embedded PC is made through a serial port. This board is also in charge of a radio control module that enables the tele-operation of the robot. The embedded PC is the core of the basic system, and it is where the high level decisions are made.

The PLD boards generate 23 khz PWM (pulse width modulation) signals for each motor and the consequent timing protection during the command changes. This protection system provides a delay during power connection, and at the moment of change of the rotation motor direction. A hardware ramp is also implemented in order to facilitate a better transition between command changes. The value of the speed is encoded in a byte, it can be generated from 0 to 127 advancing or reversing speed commands that are sent to the PLD boards through the parallel port. The PLD boards also measure the pulses provided by the encoders, during an adjustable period of time, giving the PC the speed of each wheel at every 25ms. The absolute position of each encoder is also measured by two absolute counters used in order to measure the position and orientation of the robot by the odometer system. The shaft encoders provide 500 counts/rev since encoders are placed at the motor axes; it means that the encoders provide 43,000 counts for each turn of the wheel due to the gear reduction. Moreover, the μc has control of the sonar sensors, so for each sensor a distance measure is obtained. The ultrasound sensor range is comprised of between 3cm and 5m. The data provided by these boards are gathered through the serial port in the central computer based on a VIA C3 EGBA 733/800 MHz CPU running under LINUX Debian OS. The rate of communication with these boards is 9600 b/s. Fig. 2.b shows the electronic and sensorial system blocks.
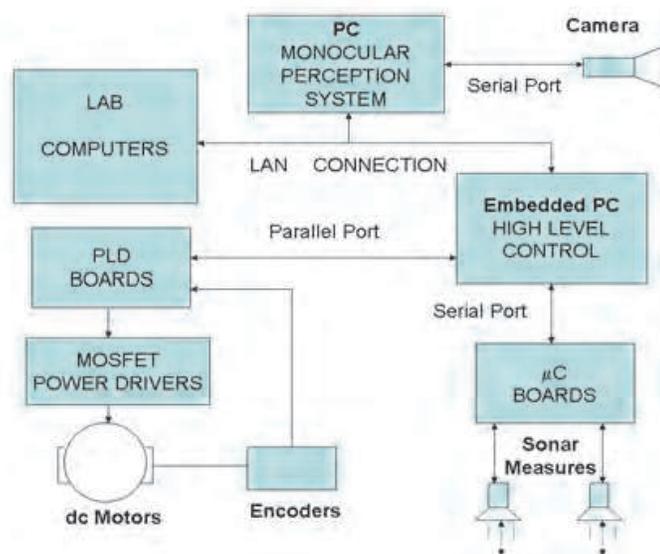

Fig. 2. (a) The open mobile educational platform PRIM.
(b) Sensorial and electronic system blocks.

The flexibility of the system allows different hardware configurations as a function of the desired application and consequently the ability to run different programs on the μc or PLD boards. The open platform philosophy is reinforced by the use of the similar μc and PLD boards that are used as teaching tools at our school. Furthermore, reinforcement of the teaching activities can be achieved through the integration of different subjects. The system's flexibility is increased with the possibility of connecting it to other computer systems through a local LAN. Hence, in this research a network server has been implemented, which allows the different lab groups to make their remote connection with the mobile robot during different practices. The use of a PC as a high level device is shown from two different points of view, consisting of PC high level programming and the TCP/IP protocol net knowledge that allow on-robot LINUX created functions to be run by the use of the laboratory LAN. Thus, lab personal computers become a flexible and easy tool that allows communication with the on-robot hardware real-time devices through the use of WIFI connections.

3. Gaining control knowledge

In this section it is presented the MCS and TPC learning activities that introduce the basic control theory. The control knowledge is mainly acquired through the following lab practices:

- Experimental identification. The model of the system.
- Designing and testing speed PID controllers.
- Trajectory following, controlling the robot position.
- Trajectory following by using MPC techniques as other suitable advanced control method.

It is noted that the compulsory curriculum of Computer Engineering has a lack of control science theory, while obviously students show experience in their programming skills. In this context, the control science is introduced from a general point of view. Moreover, students have a self-guide lab practices manual that includes the important theoretical aspects.

3.1 Experimental model and system identification

The model is obtained through the approach of a set of lineal transfer functions that include the nonlinearities of the whole system. The parametric identification process is based on black box models (Norton, 1986; Ljung, 1989). The nonholonomic system dealt with in this work is considered initially to be a MIMO (multiple input multiple output) system, as shown in Fig. 3, due to the dynamic influence between two DC motors. This MIMO system is composed of a set of SISO (single input single output) subsystems with coupled connection.
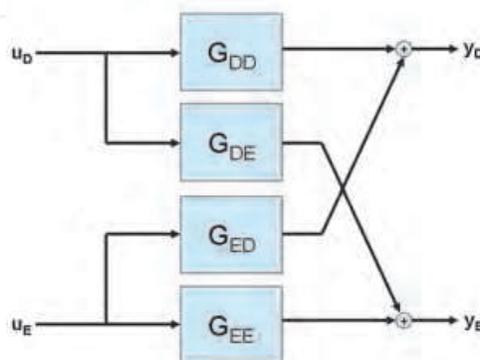

Fig. 3. The MIMO system structure

First lab control practical begins with experimental parametric model identification. Then, the model is obtained by sending different PRBS to the robot for the low, medium and high speeds, see Fig.4. Students select the adequate speed model and load the experimental data onto the MATLAB environment. Each data group has 5 different arrays containing time, right motor reference velocity, right motor measured velocity, left motor reference velocity, left motor measured velocity, respectively. Then, they obtain the time response of the inputs and outputs. The system identification is carried out by using the MATLAB toolbox *"ident"*, shown in Fig. 5, selecting a second-order ARX model and importing the necessary workspace data that should be identified.
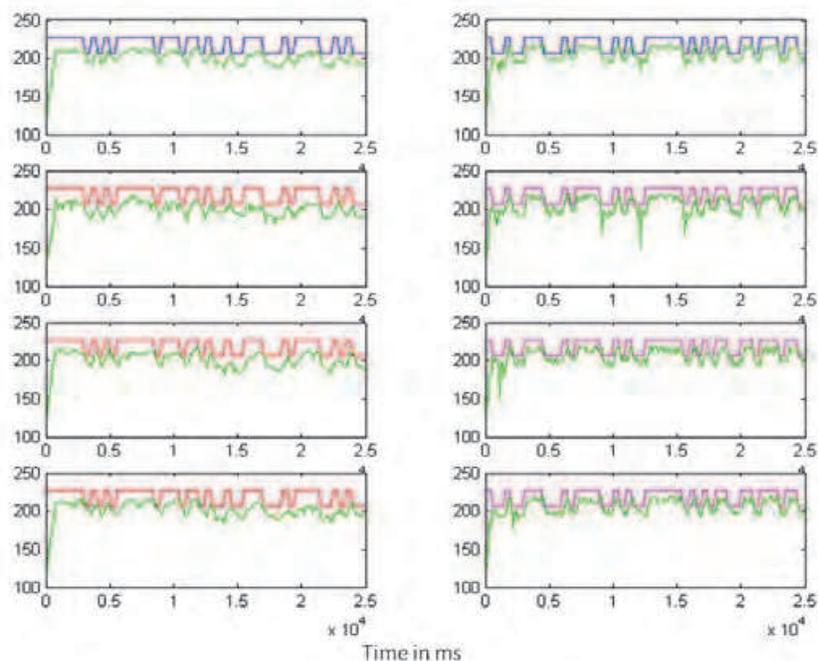
Fig. 4. Speed system outputs corresponding to the PRBS input signals
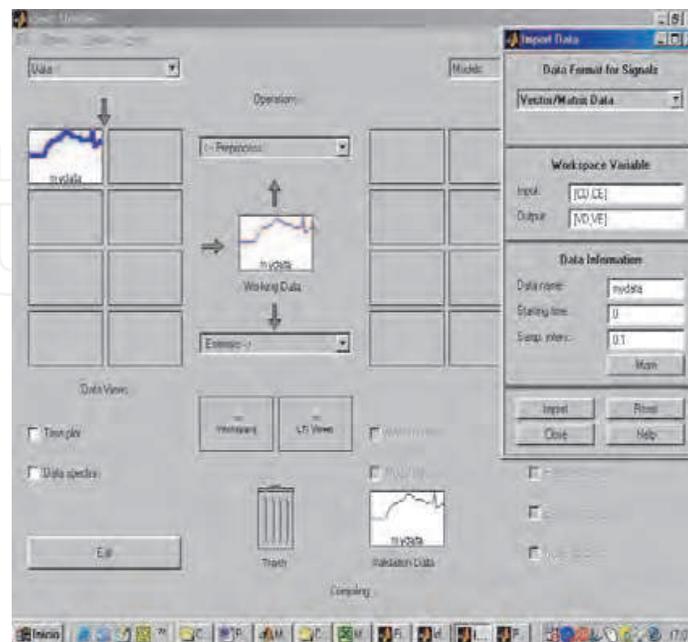
Fig. 5. System identification toolbox of MATLAB.

Tendency suppression and data filtering are suggested. The complete MIMO (multiple input multiple output) discrete-time model obtained will be used by students for validating the effectiveness of control. Table 1 shows a possible set of continuous transfer functions obtained for the three different speed lineal models used.

Linear Transfer Function	High velocities	Medium velocities	Low velocities
G_{DD}	$\dfrac{0.20s^2 - 3.15s + 9.42}{s^2 + 6.55s + 9.88}$	$\dfrac{0.20s^2 + 3.10s + 8.44}{s^2 + 6.17s + 9.14}$	$\dfrac{0.16s^2 + 2.26s + 5.42}{s^2 + 5.21s + 6.57}$
G_{ED}	$\dfrac{-0.04s^2 - 0.60s - 0.32}{s^2 + 6.55s + 9.88}$	$\dfrac{-0.02s^2 - 0.31s - 0.03}{s^2 + 6.17s + 9.14}$	$\dfrac{-0.02s^2 - 0.20s + 0.41}{s^2 + 5.21s + 6.57}$
G_{DE}	$\dfrac{-0.01s^2 - 0.08s - 0.36}{s^2 + 6.55s + 9.88}$	$\dfrac{0.01s^2 + 0.13s + 0.20}{s^2 + 6.17s + 9.14}$	$\dfrac{-0.01s^2 - 0.08s - 0.17}{s^2 + 5.21s + 6.57}$
G_{EE}	$\dfrac{0.31s^2 + 4.47s + 8.97}{s^2 + 6.55s + 9.88}$	$\dfrac{0.29s^2 + 4.11s + 8.40}{s^2 + 6.17s + 9.14}$	$\dfrac{0.25s^2 + 3.50s + 6.31}{s^2 + 5.21s + 6.57}$

Table 1. The second order WMR models

The second practice consists in finding a simplified robot model. The continuous-time model is introduced into the SIMULINK environment, as shown in Fig. 6. Several studies concerning about the importance of coupling terms are carried out by students through the analysis of stationary gains, while the order reduction is done by searching for dominant poles. The frequency response and BODE analysis can provide a dominant pole existence, which can reduce the model order of the system. Students learn from the obtained results that the MIMO model can be approximated by two SISO models.
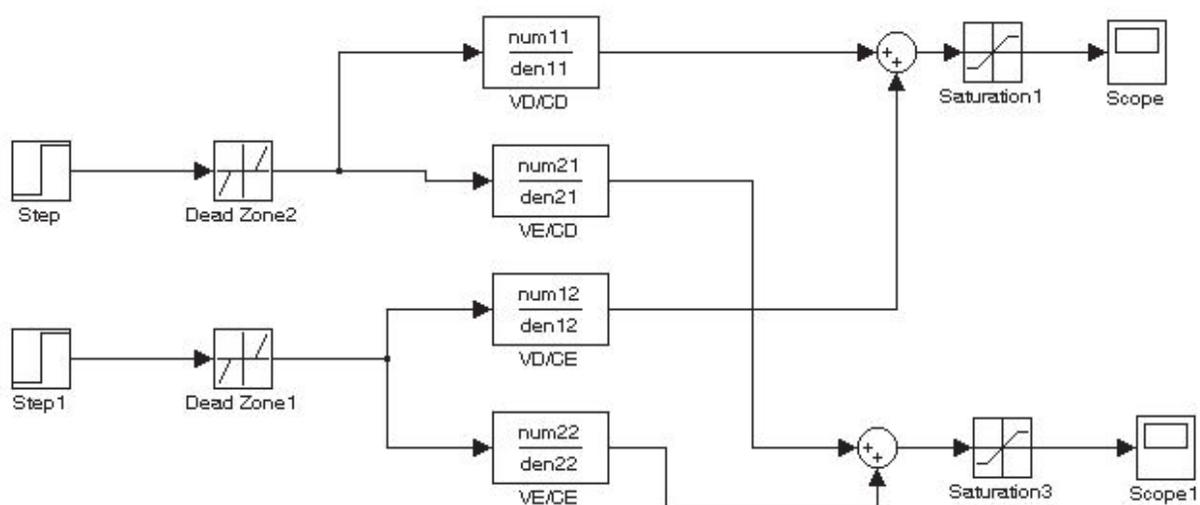

Fig. 6. MIMO robot model in SIMULINK environment.

Finally, the reduced SISO systems should be validated by considering the static gain and simulating reduced and complete robot models. Table 2 shows an example of the first order

transfer functions obtained. Students can validate the results obtained by sending open loop speed commands to the robot.

Linear Transfer Function	High velocities	Medium velocities	Low velocities
G_{DD}	$\dfrac{0.95}{0.42s+1}$	$\dfrac{0.92}{0.41s+1}$	$\dfrac{0.82}{0.46s+1}$
G_{EE}	$\dfrac{0.91}{0.24s+1}$	$\dfrac{0.92}{0.27s+1}$	$\dfrac{0.96}{0.33s+1}$

Table 2. The reduced WMR model

3.2 Designing and testing the speed controllers

The third practice consists of obtaining a controller based on the simplified SISO models. The students will try to make the controller design by using pole placement techniques and the MATLAB design facilities. For instance, students can use the MATLAB "sisotool", as shown in Fig. 7, to design the controller by first importing the transfer functions corresponding to both robot wheels. Then, they depict the open loop frequency response without a compensator. Afterwards, students learn to add a PI controller to the system in order to achieve the desired response. The control performance is verified by using the complete MIMO model of the robot in the presence of several perturbations.
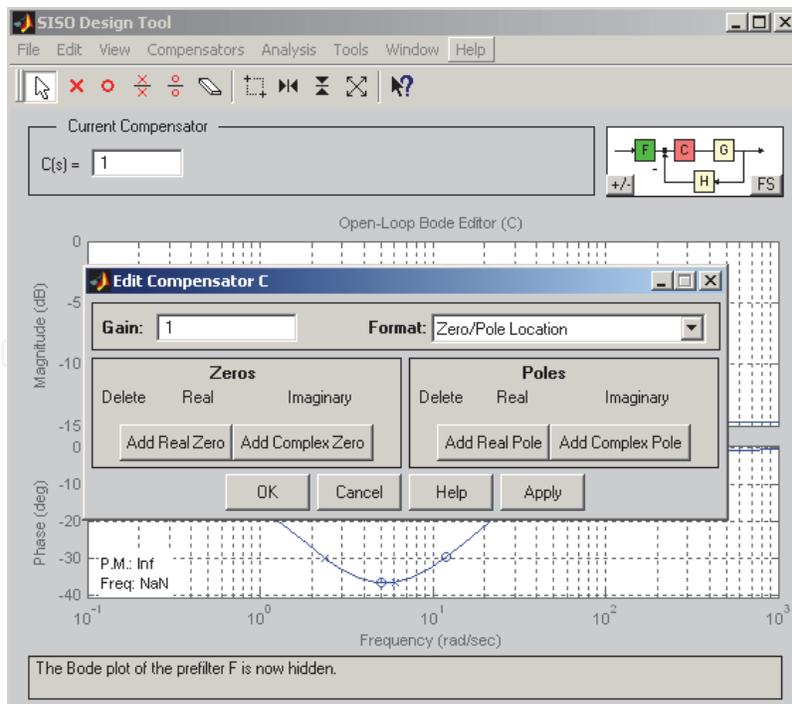
Fig. 7. "sisotool" design window.

On-robot speed control is an objective of the last MCS practice. Students transform firstly the controller from continuous to discrete time. Then, they use the robot remote network

server to send the controller parameters to the robot in order to test the speed control performance.

3.3 Trajectory following using PID speed controllers

Once PID speed controllers are obtained, TPC block practices are used for performing trajectory tracking controllers. Student goal is following trajectories by using high level languages and the existing communication functions that allow sending speed commands to the robot and to receive the odometer data. The coordinates (x, y, θ) can be expressed:

$$
\begin{aligned}
x(k+1) &= x(k) + dS \cos(\theta(k) + d\theta) \\
y(k+1) &= y(k) + dS \sin(\theta(k) + d\theta) \\
\theta(k+1) &= \theta(k) + d\theta
\end{aligned}
\tag{1}
$$

Fig. 8.a shows two consecutive robot coordinates. Fig. 8.b describes the positioning of robot as a function of the radius of left and right wheels (R_e, R_d), and the angular incremental positioning (θ_e, θ_d), with E being the distance between two wheels and dS the incremental displacement of the robot. The position and angular incremental displacements are expressed as:

$$
dS = \frac{R_d d\theta_d + R_e d\theta_e}{2} \qquad d\theta = \frac{R_d d\theta_d - R_e d\theta_e}{E}
\tag{2}
$$

The incremental position of the robot can be obtained by the odometer system through the available encoder information from (1) and (2).
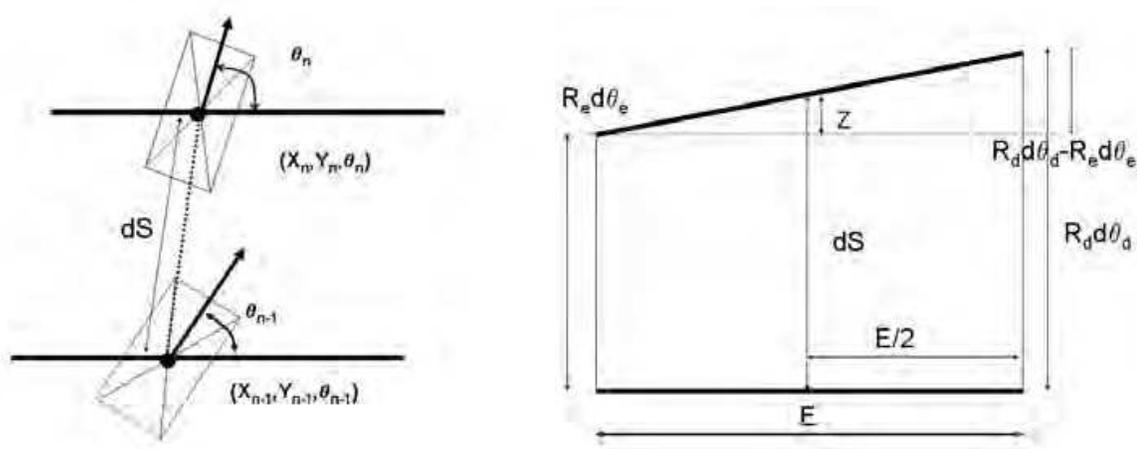
Fig. 8. (a) Robot coordinates at two consecutive instants of time.
(b) Robot position as function of the angular displacement of each wheel.

When students are able to obtain the robots coordinates, the path planning approach consists of following a sequence of straight lines. Thus, the trajectory tracking can be performed through either straight lines or turning actions (Reeds and Shepp, 1990). Trajectory tracking using discontinuous control laws can be implemented by using heuristic concepts related to experimental dynamic knowledge of the system. Therefore, when the error of path distance of the robot is greater than a heuristic threshold, the control law can

minimize the distance to the path by performing a turning action. Otherwise the desired point distance should be minimized (Pacheco and Luo, 2006). Orientation and desired point distances can be considered as important data for achieving proper path-tracking. Their deviation errors are used in the control law as necessary information for going forward or turning. Students have to compute Cartesian distances between the robot and the trajectory to be followed. Fig. 9 depicts these concepts for two cases.
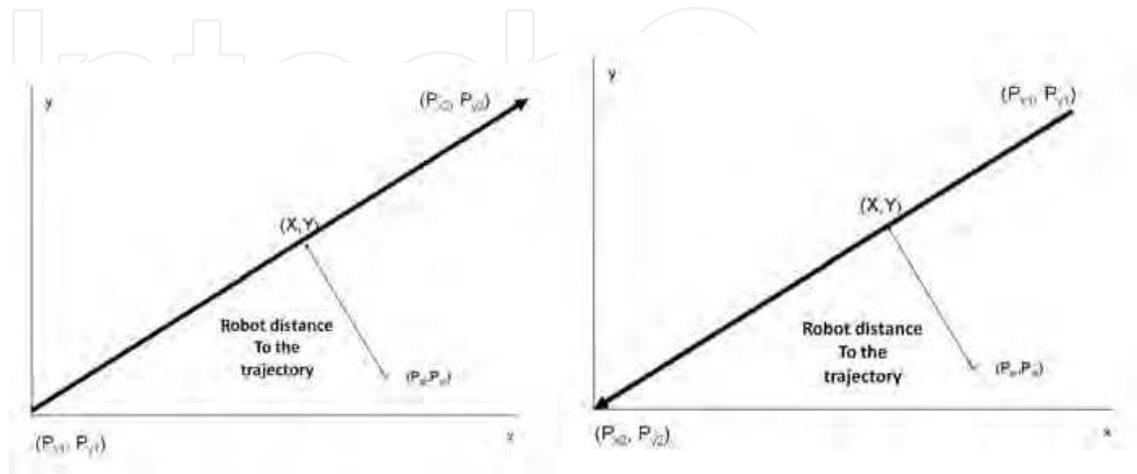

Fig. 9. (a) Straight line trajectory from (P_{x1}, P_{y1}) to (P_{x2}, P_{y2}).
(b) Straight line trajectory from (P_{x2}, P_{y2}) to (P_{x1}, P_{y1}).

The proposed control law is based on sending the same speed to both wheels when the robot distance to the trajectory is less than a heuristic threshold and different speed to each wheel in order to perform a turning action when such distance is greater. As can be observed in Fig. 9, the turning action must be different as a consequence of the robot position, up or down the trajectory, or the trajectory sense. Fig. 10 shows a trajectory following experience with two straight lines that have a 90º angle.
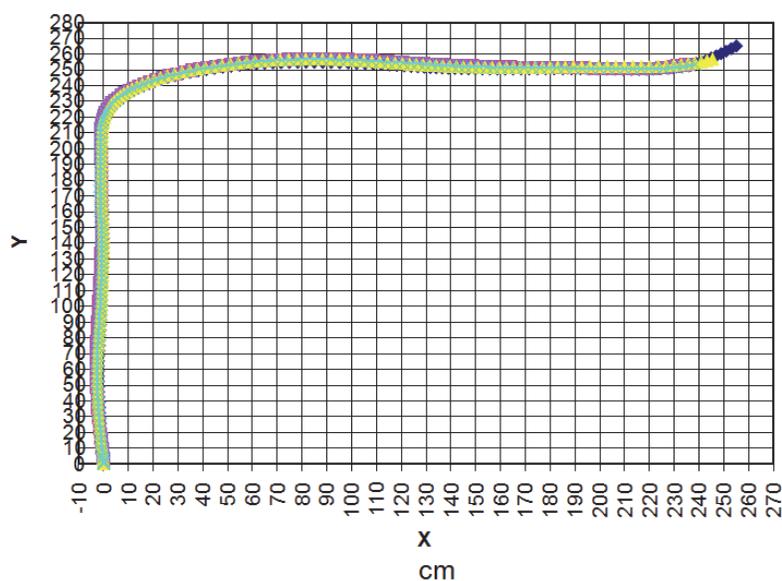
Fig. 10. Example of trajectory following experiences, using discontinuous control laws and PID speed controllers.

3.4 Trajectory following using MPC techniques

In this subsection the LMPC (local model predictive control) techniques based on the dynamics models obtained in the previous subsection are presented. LMPC strategies are simulated in the laboratory by using the methods and software developed by those of our teaching staff participating in the course (Pacheco and Luo, 2011).

3.4.1 The LMPC formulation

The main objective of highly precise motion tracking consists in minimizing the error between the robot and the desired path. Global path-planning becomes unfeasible since the sensorial system of some robots is just local. In this way, LMPC is proposed in order to use the available local perception data in the navigation strategies. Concretely, LMPC is based on minimizing a cost function related to the objectives for generating the optimal WMR inputs. Define the cost function as follows:

$$
J(n,m) = \min_{\left\{U(k+i|k)_{i=0}^{i=m-1}\right\}} \left\{
\begin{array}{l}
\left[X(k+n|k)-X_{ld}\right]^T P\left[X(k+n|k)-X_{ld}\right] \\
+\sum_{i=1}^{n-1}\left[X(k+i|k)-\overline{X_{ld}X_{l0}}\right]^T Q\left[X(k+i|k)-\overline{X_{ld}X_{l0}}\right] \\
+\sum_{i=1}^{n-1}\left[\theta(k+i|k)-\theta_{ld}\right]^T R\left[\theta(k+i|k)-\theta_{ld}\right] \\
+\sum_{i=0}^{m-1} U^T(k+i|k)SU(k+i|k)
\end{array}
\right\} \tag{3}
$$

The first term of (3) refers to the attainment of the local desired coordinates, $X_{ld}=(x_d,y_d)$, where (x_d, y_d) denote the desired Cartesian coordinates. $X(k+n/k)$ represents the terminal value of the predicted output after the horizon of prediction n. The second one can be considered as an orientation term and is related to the distance between the predicted robot positions and the trajectory segment given by a straight line between the initial robot Cartesian coordinates $X_{l0}=(x_{l0}, y_{l0})$ from where the last perception was done and the desired local position, X_{ld}, to be achieved within the perceived field of view. This line orientation is denoted by θ_{ld} and denotes the desired orientation towards the local objective. $X(k+i/k)$ and $\theta(k+i/k)$ $(i=1,\ldots n-1)$ represents the predicted Cartesian and orientation values within the prediction horizon. The third term is the predicted orientation error. The last one is related to the power signals assigned to each DC motor and are denoted as U. The parameters P, Q, R and S are weighting parameters that express the importance of each term. The control horizon is designed by the parameter m. The system constraints are also considered:

$$
\left\{
\begin{array}{ll}
G_0 < |U(k)| \le G_1 & \alpha \in (0,1] \\
|X(K+n/k)-X_{ld}| \le \alpha|X(k)-X_{ld}| & \\
\text{or } |\theta(k+n/k)-\theta_{ld}| \le \alpha|\theta(k)-\theta_{ld}| &
\end{array}
\right\} \tag{4}
$$

where $X(k)$ and $\theta(k)$ denote the current WMR coordinates and orientation, $X(k+n/k)$ and $\theta(k+n/k)$ denote the final predicted coordinates and orientation, respectively. The limitation

of the input signal is taken into account in the first constraint, where G_0 and G_1 respectively denote the dead zone and saturation of the DC motors. The second and third terms are contractive constraints (Wang, 2007), which result in the convergence of coordinates or orientation to the objective, and should be accomplished at each control step.

3.4.2 The algorithms and simulated results

By using the basic ideas introduced in the previous subsection, the LMPC algorithms have the following steps:

1. Read the current position;
2. Minimize the cost function and to obtain a series of optimal input signals;
3. Choose the first obtained input signal as the command signal;
4. Go back to the step 1 in the next sampling period.

In order to reduce the set of possibilities, when optimal solution is searched for, some constraints over the DC motor inputs are taken into account:

- The signal increment is kept fixed within the prediction horizon.
- The input signals remain constant during the remaining interval of time.

The above considerations will result in the reduction of the computation time and the smooth behavior of the robot during the prediction horizon (Maciejowski, 2002). Thus, the set of available input is reduced to one value, as it is shown in Fig. 11.
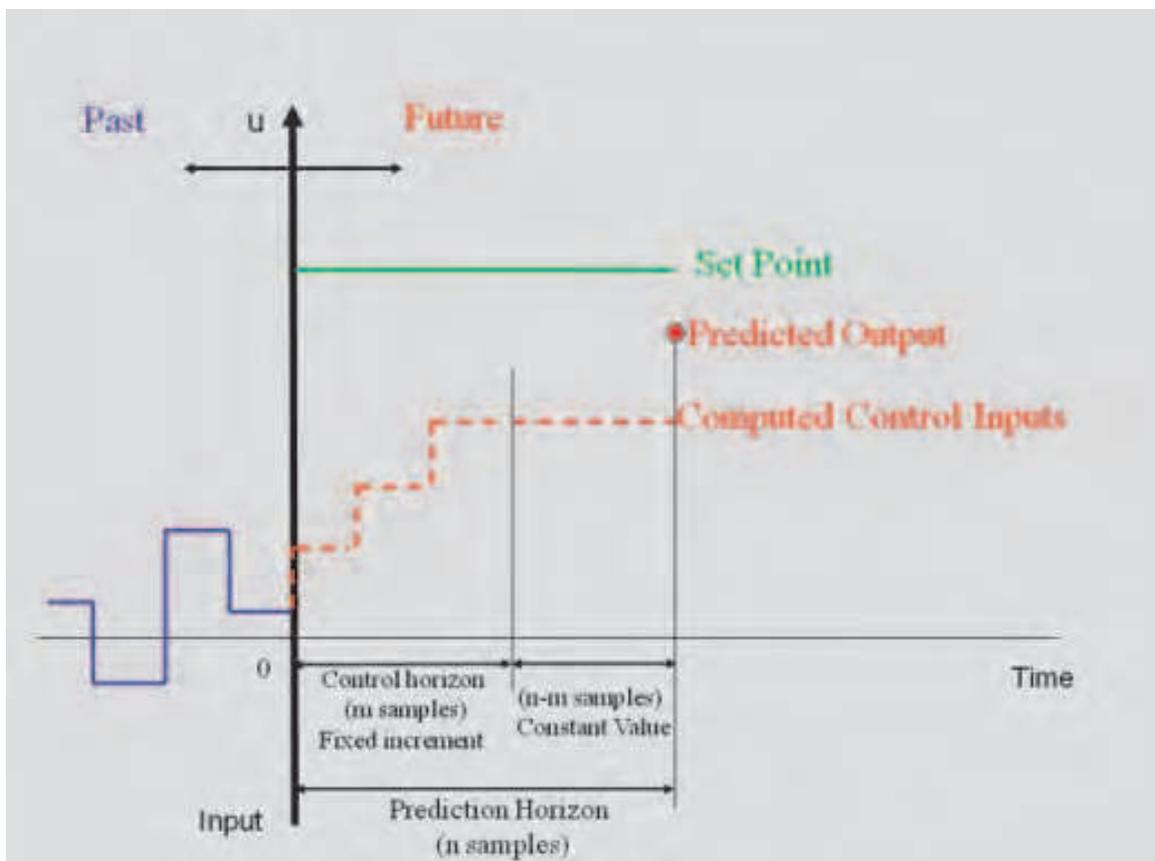
Fig. 11. LMPC strategy with fixed increment of the input during the control horizon and constant value for the remaining time

The LMPC simulation software provides to students a set of files and facilities to draw the results, and consequently the students can represent the files containing the cost function values, for different available set of inputs, corresponding to different moments of the simulation. Simulation environment allows testing the control law with different set up. Therefore, different control and prediction horizons can be tested. The control law parameters of equation (3) can be analyzed and modified according to the different horizons of prediction. Fig. 12 depicts the results obtained, m=3 and n=5, when a trajectory composed of 3 straight lines is followed. The lines are defined by the coordinates (0, 0), (0, 100), (50, 200) and (250, 200) where (x, y) are given in cm. It is noted that the sample time is 0.1s, so n=5 means 0.5s.
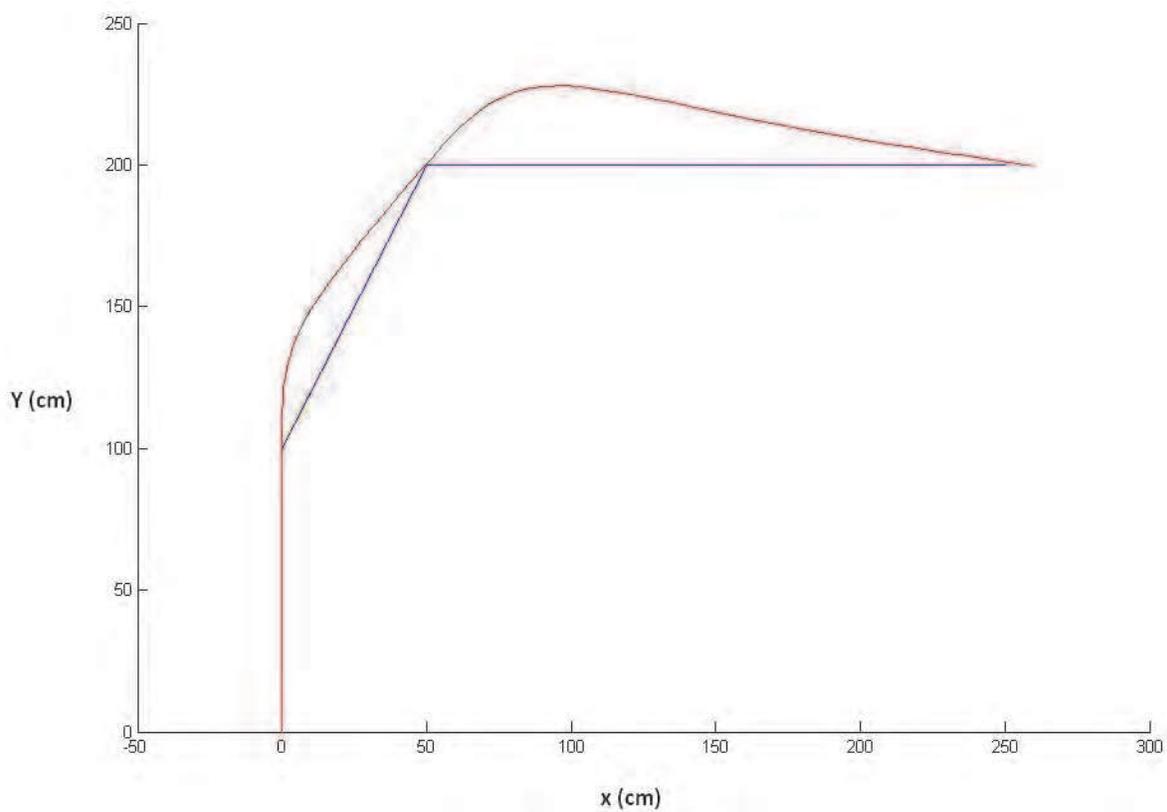
Fig. 12. Simulation results obtained (m=3, n=5, P=1, Q=1, R=1 and S=0).

Fig. 13 show the simulation results when the control horizon is set to 4 and the prediction horizon to 8.
The results obtained in Fig. 13 show that the robot is stopped near (50, 200) coordinates due to contractive constraints, and consequently the final coordinates (250, 200) cannot be achieved. The reason is that each new straight line trajectory is commanded when the final coordinate of trajectory is reached considering a tolerance threshold but in the above case the coordinates (50, 200) will not be reached even considering the tolerance threshold. The problem can be solved by modifying the parameter weights of the control law as it is shown in Fig. 14.
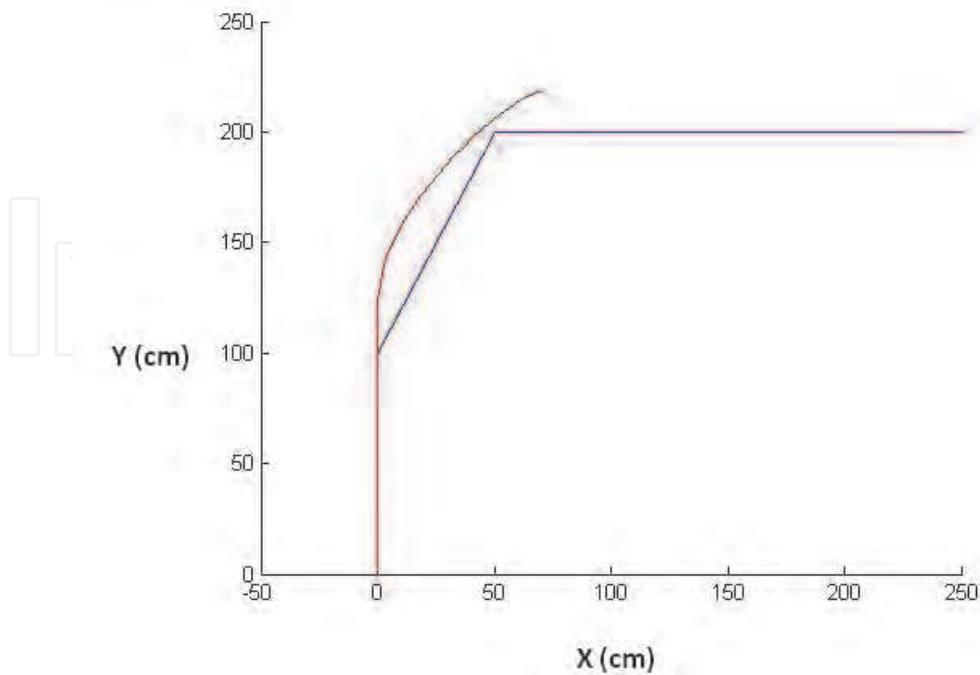
Fig. 13. Simulation results obtained (m=4, n=8, P=1, Q=1, R=1 and S=0).
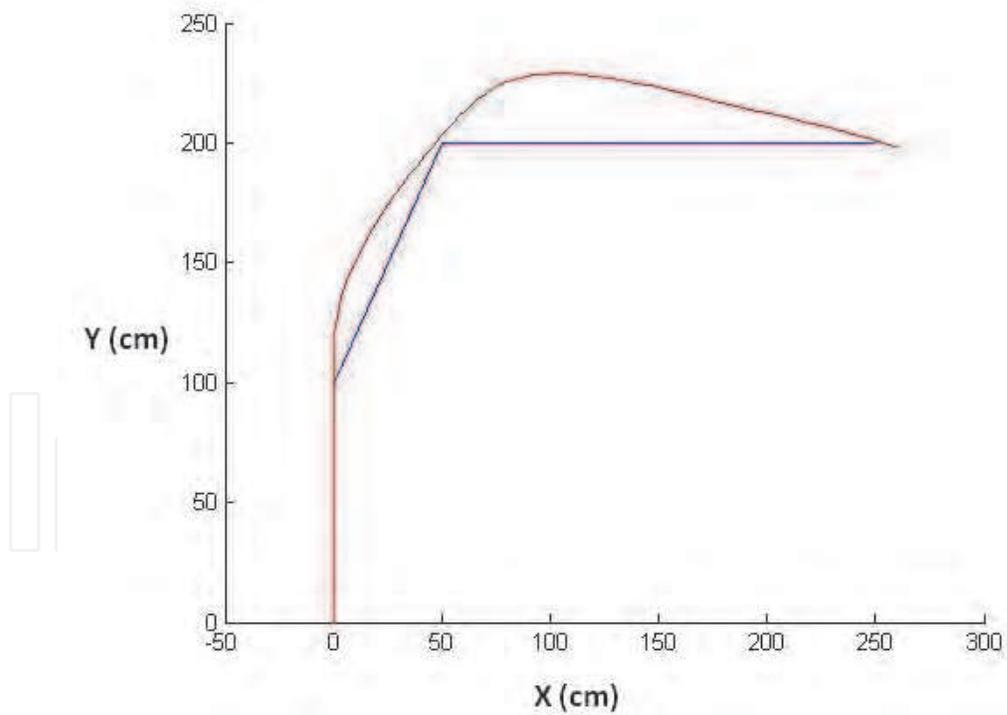
Fig. 14. Simulation results obtained (m=4, n=8, P=1, Q=2, R=1 and S=0).

4. Conclusions

The teaching experience gained from this course has shown the usefulness of the mobile robot as an important experimental platform for computer engineering education. The

course framework allows a different level of knowledge learning according to the student skills. Various control techniques have been provided to students as a main issue in order to improve their knowledge. They can acquire a higher level of skills by integrating different subjects related to electronics, control system, computer science, communication, etc. Moreover, students have the opportunity to be familiar with basic control techniques as well as some advanced methods as MPC. It should be mentioned that the developed lab practices related to the theoretic counterparts have increased the motivation of the students and have achieved the multidisciplinary knowledge consolidation of the proposed objective of control education. In this context, according to the student's feedback on the course, the average opinion of 30 students of the course is 4 over 5 according to the different survey questions that the university provides at the end of the course. They underline the interesting and attractive educational and practical aspects as the relevant characteristics of the course. One of the suggestions is to increase the number of hours in order to perform better the theoretical analysis, lab practices and algorithm implementation. Furthermore, some students decide to do their compulsory final degree project by improving the robot performance in some practical aspects; e.g., nowadays the design of factorial experiments to perform the MPC cost function tuning. In the future, the teaching staff from different research areas will do their utmost to promote the development and consolidation of the course so that the quality of teaching can be further improved. In this process, the student's opinions are important enough to encourage this work. Therefore, the path for achieving renewed progress in robotics is through the integration of several fields of knowledge, such as computing, communications, and control sciences, so as to perform a higher level reasoning and use decision tools with strong theoretical base.

5. Acknowledgment

6. References

Amante, B., Romero, C., Piñuela, J.A. (2007). Aceptación de la metodología del aprendizaje colaborativo en diferentes ciclos de las carreras técnicas. Cuadernos de innovación educativa de las Enseñanzas Técnicas Universitarias. Vol. 1, No. 1, 2007, 65-74.

Antsaklis, P., Basar, T., Decarlo, R., McClamroch, N. H., Spong M., and Yurkovich, S. (1999). Report on NSF/CSS workshop on new directions in control engineering education. *IEEE Control Systems Magazine*, Vol. 19, No. 5, (Oct. 1999) 53-58.

Aström K.J and Wittenmark B., (1988). Computed Controlled Systems. Theory and Design. *Prentice-Hall International Ed.*, New Jersey, 1988.

Barnett, R. H. (1994). The 8051 Family of Microcontrollers. *Prentice Hall International Ed.*, New Jersey, 1994.

Carnegie Mellon University. *The Robotics Institute: Robotics Education in the School of Computer Science*. In URL: http://www.ri.cmu.edu, 2011.

European Commission. *Higher Education in Europe, inside the police areas of European Commission*, in URL:
http://ec.europa.eu/education/policies/higher/higher_en.html, 2008.

Horn, B. K. P. (1998). Robot Vision. *MIT press*, Ed. McGraw–Hill, ISBN 0-262-08159-8, London (England)

Hristu-Varsakelis, D., and Brockett, R. W. (2002). Experimenting with Hybrid Control. *IEEE Control Systems Magazine*, Vol. 22, No. 1, (Feb. 2002), 82-95.

Kuo, C. (1996). Automatic Control Systems. *Prentice-Hall International Editors*, New Jersey, 1996.

Ljung, L. (1989). System Identification: Theory for the User. Prentice-Hall Inter. Ed., New Jersey, 1989.

Maciejowski, J.M. (2002). Predictive Control with Constraints, *Ed. Prentice Hall*, ISBN 0-201-39823-0, Essex (England)

Murray, R. M., Aström, K. J., Boyd, S. P., Brockett, R. W., and Stein, G. (2003). Future Directions in Control in an Information-Rich World. *IEEE Control Systems Magazine*, Vol. 23, No. 2, (April 2003) 20-33.

Norton, J. (1986). An Introduction to Identification. *Academic Press*, ISBN 0125217307, London and New York, 1986

Pacheco, L., Batlle, J., Cufí, X., Arbusé, R. (2006). PRIM an Open Mobile Platform. Motivation, Present and Future Trends. *Inter. Conf. on Automation, Quality and Testing, Robotics*. Vol. 2, 231-236, 2006.

Pacheco, L. and Luo, N. (2006). Mobile robot experimental modelling and control strategies using sensor fusion. *Control Engineering and Applied Informatics*. Vol. 8, No.3, (2006), 47-55.

Pacheco, L., Luo, N., Arbusé, R., Ferrer, I., and Cufí, X. (2009). Interdisciplinary Knowledge Integration Through an Applied Mobile Robotics Course. *International Journal of Engineering Education*, Vol. 25, No. 4, (July 2009), 830-840.

Pacheco, L., Luo, N. (2011) Mobile robot local trajectory tracking with dynamic model predictive control techniques, *International Journal of Innovative Computing, Information and Control*, Vol.7, No.6, (June 2011), ISSN 1349-4198, 3457-3483

Qu, Z. and Wu XH. (2006). A new curriculum on planning and cooperative control of autonomous mobile robots. *International Journal of Engineering Education*, Vol. 22, No 4, (July, 2006), 804-814.

Reeds, J. A and Shepp, L. A. (1990). Optimal paths for a car that goes forwards and backwards. *Pacific Journal of Mathematics*, vol. 145, 1990.

Schilling, R. J. (1990). Fundamentals of Robotics. *Prentice-Hall International Ed.*, New Jersey, 1990.

Valera, A., Weiss, M., Valles, M., Diez, J. L. (2007). Control of mobile robots using mobile technologies. *International Journal of Engineering Education*, Vol. 23, No 3, (March 2007), 491-498.

Wakerly, J.F. (1992). Digital Design Principles and Practices. *Prentice-Hall International Ed.*, New Jersey, 1992.

Wan, J. (2007) Computational reliable approaches of contractive MPC for discrete-time systems, *PhD Thesis, University of Girona*

Wilczynski, V. and Flowers, W. (2006). FIRST Robotics Competition: University curriculum applications of mobile robots. *International Journal of Engineering Education*, Vol. 22, No. 4, (July, 2006), 792-803.
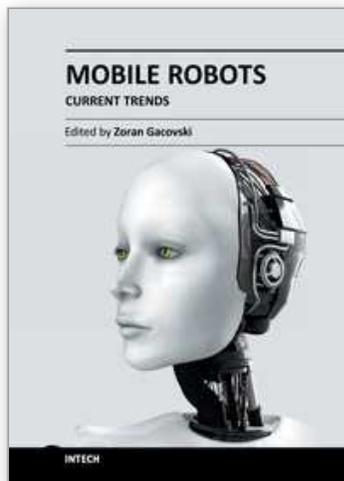
Mobile Robots - Current Trends

Edited by Dr. Zoran Gacovski

This book consists of 18 chapters divided in four sections: Robots for Educational Purposes, Health-Care and Medical Robots, Hardware - State of the Art, and Localization and Navigation. In the first section, there are four chapters covering autonomous mobile robot Emmy III, KCLBOT - mobile nonholonomic robot, and general overview of educational mobile robots. In the second section, the following themes are covered: walking support robots, control system for wheelchairs, leg-wheel mechanism as a mobile platform, micro mobile robot for abdominal use, and the influence of the robot size in the psychological treatment. In the third section, there are chapters about I2C bus system, vertical displacement service robots, quadruped robots - kinematics and dynamics model and Epi.q (hybrid) robots. Finally, in the last section, the following topics are covered: skid-steered vehicles, robotic exploration (new place recognition), omnidirectional mobile robots, ball-wheel mobile robots, and planetary wheeled mobile robots.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

INTECH
open science | open minds