The CDIO initiative from an automatic control project course perspective

Martin Enqvist, Svante Gunnarsson, Miklael Norrlöf, Erik Wernholt, Anders Hansson

Division of Automatic Control
Department of Electrical Engineering
Linköpings universitet, SE-581 83 Linköping, Sweden
WWW: http://www.control.isy.liu.se
E-mail: maren@isy.liu.se, svante@isy.liu.se,
mino@isy.liu.se, erikw@isy.liu.se, hansson@isy.liu.se

AUTOMATIC CONTROL
COMMUNICATION SYSTEMS
LINKÖPING

Report no.: LiTH-ISY-R-2651
Submitted to 16th IFAC World Congress

Abstract

The CDIO (Conceive Design Implement Operate) Initiative is explained, and some of the results at the Applied Physics and Electrical Engineering program at Linköping University, Sweden, are presented. A project course in Automatic Control is used as an example. The projects within the course are carried out using the LIPS (Linköping interactive project steering) model. An example of a project, the golf playing industrial robot, and the results from this project are also covered.

Keywords: Control education, project management, robot programming, mathematical models, group work

THE CDIO INITIATIVE FROM AN AUTOMATIC CONTROL PROJECT COURSE PERSPECTIVE

M. Enqvist, S. Gunnarsson, M. Norrlöf,
E. Wernholt and A. Hansson

Dept. of Electrical Engineering, Linköping University,
SE-58183 Linköping, Sweden

Abstract: The CDIO (Conceive Design Implement Operate) Initiative is explained, and some of the results at the Applied Physics and Electrical Engineering program at Linköping University, Sweden, are presented. A project course in Automatic Control is used as an example. The projects within the course are carried out using the LIPS (Linköping interactive project steering) model. An example of a project, the golf playing industrial robot, and the results from this project are also covered.

Keywords: Control education, project management, robot programming, mathematical models, group work

1. INTRODUCTION

This paper presents results of a student project in Automatic Control that was carried out during the spring of 2004. The project was one of approximately 25 projects, organized in eleven different courses, that were carried out by students in the fourth year of the Applied Physics and Electrical Engineering program at Linköping University, Sweden. The introduction of the project courses is a result of the participation in the CDIO Initiative, which is an international collaboration between a number of universities with the aim to develop further engineering education. The paper is organized as follows. Section 2 gives a short description of the CDIO Initiative, its organization, and its main goals, and in Section 3 the Applied Physics and Electrical Engineering program is presented briefly. In all the project courses that have been introduced at Linköping University via the CDIO Initiative, the project work is carried out using a common and structured methodology. This methodology is defined by the project model LIPS, and it is presented in Section 4. The Automatic Control project course is one of the eleven project courses, and it is described in some detail in Section 5. The project that will be presented in this paper is the development of a golf playing industrial robot, and this project will be presented in Section 6. Finally, Section 7 contains the conclusions.

2. THE CDIO INITIATIVE

The CDIO Initiative started in 2000, and from the beginning it consisted of three universities from Sweden (Linköping University, Chalmers University of Technology, and the Royal Institute of Technology) and one university from the USA (Massachusetts Institute of Technology). During the development of the initiative, a number of universities have joined the CDIO Initiative. A list of the participating universities can be found on the web site `http://www.cdio.org`. The aims of the project are to emphasize the CDIO view of engineering education and to present a systematic procedure for developing an engineering program into a CDIO program. The activities within the CDIO Initiative are based on two documents, the CDIO Syllabus (*The CDIO Syllabus* 2004) and the CDIO Standards (*The CDIO Standards* 2004), respectively. The first document, the CDIO Syllabus, can be seen as a specification of the desired knowledge and skills of the students that graduate from the engineering education. The Syllabus is organized in the following four sections:

1. Technical knowledge and reasoning
2. Personal and professional skills and attributes
3. Interpersonal skills: Teamwork and communication
4. Conceiving, designing, implementing and operating systems in enterprise and societal context.

For each section there are subsections specifying in more detail the desired skills of an engineer. The main goal of the CDIO Initiative is to develop methods and activities that strengthen items 2-4 of the Syllabus.

The second document, the CDIO Standards, specifies the desired properties of an engineering program. The philosophy behind the CDIO Initiative is formulated in the first standard saying "Adoption of the principle that product and system lifecycle development and deployment - Conceiving, Designing, Implementing and Operating - are the context for engineering education".

3. THE APPLIED PHYSICS AND ELECTRICAL ENGINEERING PROGRAM

The Applied Physics and Electrical Engineering program is one of the largest engineering programs at Linköping University. It admits 180 students (150 in the regular program and 30 in the international version) each year. The program has a strong emphasis on mathematics, physics, and electrical engineering, and it is considered to be one of the most demanding engineering programs in Sweden. The main part of the first three years consists of mandatory courses and the main part of the fourth year is spent on a specialization within a selected area. In agreement with the Swedish system the nominal time of studies is 4.5 years, corresponding to 180 units, *i.e.* 40 units/year. 160 units are spent on courses and 20 units are spent on the Master's Thesis Project. The course part consists of approximately 115 units of mandatory courses (50 units mathematics), 25 units specialization and 20 units elective courses. Starting from the last semester of year three, the students choose one out of twelve specializations.

One of the main results of the participation in the CDIO Initiative is that a sequence of project courses has been introduced into the program. One aim of these courses is to cover items 2-4 in the CDIO Syllabus, which means emphasizing personal skills, interpersonal skills and the CDIO view of engineering. The sequence consists of an Introductory Course in year one, an Electronics project course in year three and a set of project courses in the fourth year, related to the specializations of the program. A further objective of introducing the project courses is to give the students training in project work using industry like methods. For that purpose, a project management model, LIPS (Svensson and Krysander 2004), has been developed. This project model is based on industrial project models, but it has been adapted for educational purposes.

The third stage in the sequence of project courses consists of a set of eleven courses with connections to the specializations in the program:

- Applied mathematics, project course
- Design and manufacturing of sensor chips
- Computational physics
- Mixed signal processing systems
- System design
- VLSI design project
- Image and graphics, project course
- Automatic control, project course
- Systems engineering, project course
- Biomedical engineering, project course
- Embedded systems simulation and verification

The courses are given by five different departments, and they vary between five and six units in size. All courses are given during the spring semester of year four. For the development of these courses, the board of the program formulated a set of specifications to be satisfied by the courses:

- Minimum of four students in each project group.
- At least four units spent on the project part.
- The project should be carried out using the project model LIPS.

The set of courses was given for the first time 2004. Around 210 students participated in the courses, and approximately 125 students belonged to the Applied Physics and Electrical Engineering program. 60 students were non-Swedish speaking and came from some of the international master's programs that are offered at Linköping University. The remaining group of students belonged to some of the other engineering programs.

4. THE LIPS PROJECT MODEL

The LIPS project model has been designed at Linköping University to support the CDIO concept and to introduce a professional project management approach into the academic environment.

LIPS is similar to modern industrial project models but adapted for usage in education or in small industrial projects. The model introduces the phases, definitions and tollgates necessary for running a project in an efficient way. The three steps of the model describe the project preparation and planning phase, the project execution phase and the project delivery and evaluation phase. The model also includes descriptions of activities, roles and communication flows in a project.

The different project documents are described and exemplified by electronic templates. Examples of documents are requirement specification, project plan, time plan, status report, meeting minutes, and project reflection document.

The use of milestones and tollgates is introduced. At defined tollgates the students are required to deliver documents etc. to get approval for entering the next phase in the project.

The model is scalable, and it can advantageously be used in a track of project courses with varying complexity. The model has been successfully used in more than 150 projects, and the experiences are very positive. As an example, the well-defined steps in the model automatically introduce continuous assessment. It also triggers processes that reveal if a project is delayed or if a member in a group does not contribute.

5. AUTOMATIC CONTROL PROJECT COURSE

5.1 Overall description

The automatic control project course is a 200 hours course where groups of at least six students do projects according to the LIPS project model (described in Section 4). Quoting the official course plan of the course, the aim is:

"The project should be conducted according to industrial standards and it should develop the students competence in the following areas: - How to analyze engineering problems - Research of knowledge - Application of knowledge obtained from previous courses - To find creative solutions - When applicable, the project work should consist of modeling, design, implementation and testing of a control system."

In the course 2004 there were six projects:

(1) Autonomous robot control (Dept. of EE)
(2) Golf playing industrial robot (Dept. of EE)
(3) Control of a missile (Saab Bofors Dynamics)
(4) Control of a fighter aircraft (MathCore)
(5) Target estimation for a UAV platform (Swedish Defense Research Agency)
(6) Occupant Spatial Sensing (Autoliv)

As can be seen from this list, two projects were carried out at the department, and four were carried out together with companies. In Section 6, the second project is going to be described in some detail.

5.2 Course organization

The course starts with a presentation of the available projects. The students then choose a project task they would like to carry out. Based on the students' choices, the examiner of the course assigns students to the different projects.

For each project there is a customer, sponsor, project manager, and a supervisor. For the industrial projects the customer is a person at the company, and for the internal project the customer is a faculty member from the department. The sponsor is a graduate student, and the task of the sponsor is to be the link between the customer and the project group. The role of the supervisor, who in most cases is a graduate student, is to support the group with technical issues. The project manager is one of the students in the project group. The sponsor together with the customer approves the original requirement specification. The end product is later evaluated against the requirement specification. If there are requirements that the group cannot meet, those requirements have to be negotiated with the sponsor and the customer. Formal meetings between the sponsor and the project group have to take place at tollgate two (TG2) and TG5 according to the LIPS project model. At the tollgate, the sponsor reviews the progress of the project and decides if the project is allowed to move into the next phase. Minutes from the two meetings are used by the examiner as inputs for the final assessment of the students.

Within the groups, each student gets his/her own area of responsibility. In addition to the project manager task, the areas of responsibilities are documents, quality, testing, customer relations, and design. The project manager should report weekly to the sponsor how the project develops.

6. THE GOLF PLAYING ROBOT

In this section, the results of the project "The golf playing robot" will be presented. In Section 6.1, the hardware platform that was given to the students is presented, and the aim of the project is also explained. In Sections 6.2 to 6.5 the technical details of the project and the achieved results are discussed.

6.1 Project platform

The students were given a hardware platform that they should use in the project. This platform is shown in Figure 1 and consists of a standard industrial robot, the ABB IRB1400 robot (ABB 1997*a*), a golf course built by the department, a tool with a golf putter and a vacuum device to pick up the ball.

The robot is programmed in the programming language RAPID 2.0 (ABB 1997*b*), and it is connected to the local area network in the laboratory. The project group was given a short course in robotics, programming RAPID, and safely using the robot.

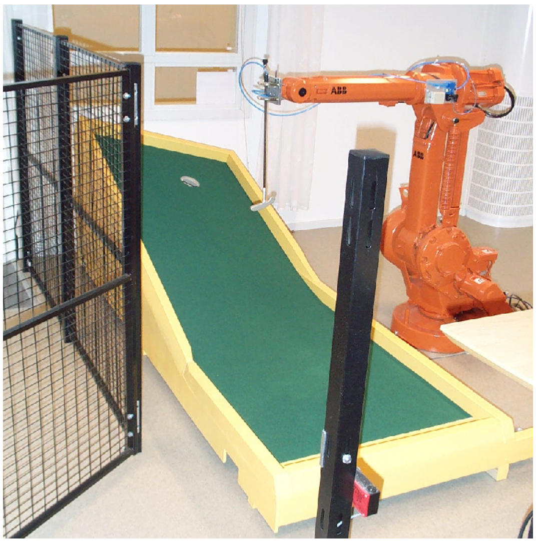
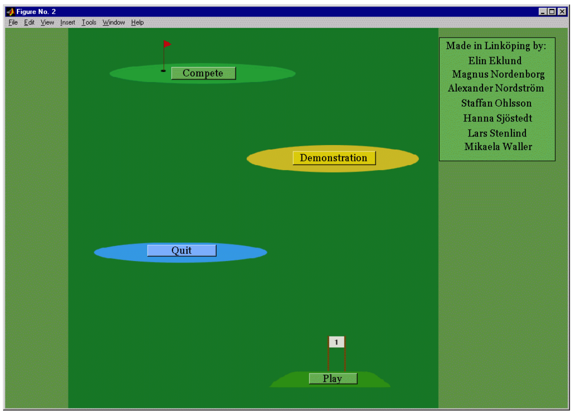


Fig. 1. The ABB IRB1400 robot and the golf course.

The goal of the project, as it was formulated to the students when the course started, was to create a golf playing robot that could be used to explain control and robotics to a wide range of people in both a pedagogical and an entertaining way. The resulting product should be interactive and have an easy-to-use graphical user interface (GUI).

6.2 Graphical user-interface

The graphical user interface (GUI) was implemented in Matlab using the Guide tool (MathWorks 2002). The software is divided into three modes which the user can reach from the system main window, shown in Figure 2. The three modes are, “competition mode”, “demonstration mode”, and “play mode”, and the GUI for the last two are shown in Figure 3 and Figure 4. In competition mode, it is possible for one or more players to compete against the robot in a game of six shots. In demonstration mode, the user can give an angle that the robot should hit the ball at. Using a model, to be described in Section 6.3, the software computes the velocity that gives the highest probability for a hole-in-one. In Figure 3 the angle 9° is shown together with the probability 0.95 for hole-in-one and the predicted ball path. In play mode, the user can choose angle and speed manually.



Made in Linköping by:
Elin Eklund
Magnus Nordenborg
Alexander Nordström
Staffan Ohlsson
Hanna Sjöstedt
Lars Stenlind
Mikaela Waller

Fig. 2. Main menu screen shot.

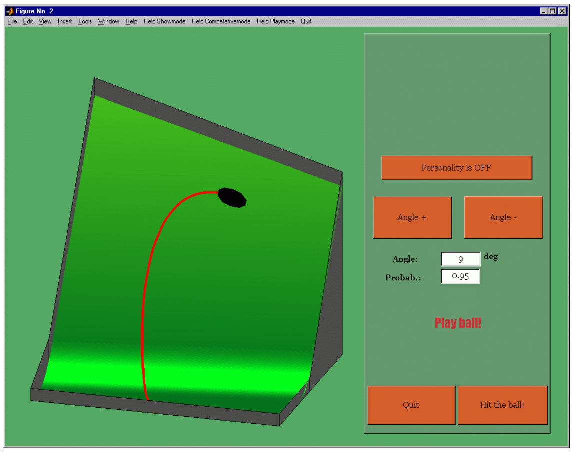


Figure No. 2
File Edit View Insert Tools Window Help Help Showmode Help Competetivemode Help Playmode Quit
Personality is OFF
Angle +
Angle -
Play ball!
Quit
Hit the ball!

Angle: 9 deg
Probab.: 0.95

Fig. 3. Demonstration mode. The user gives an angle and the software computes the speed that gives the highest probability for a hole-in-one shot and shows the predicted ball path.

6.3 Model

The model consists of a geometrical model of the golf course, a physical model of the dynamics of the ball, and a statistical model which is updated every time the robot hits the ball.

6.3.1. Geometrical model of the course The geometrical model of the golf course was found by using the robot to measure a large number of points on the course surface. The model is made out of three segments, two planes and a cubic-spline surface connecting the two planes. In Figure 5 the model is shown and the three different segments are indicated with numbers.

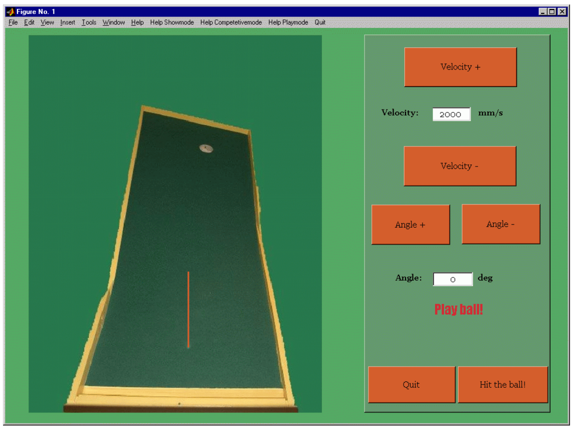


Figure No. 1
File Edit View Insert Tools Window Help Help Showmode Help Competetivemode Help Playmode Quit
Velocity +
Velocity -
Angle +
Angle -
Play ball!
Quit
Hit the ball!

Velocity: 2000 mm/s

Angle: 0 deg

Fig. 4. Play mode. The vertical line on the course shows the angle which the robot will hit the ball at. The user can choose angle and velocity for the robot by pushing the different buttons in the interface.

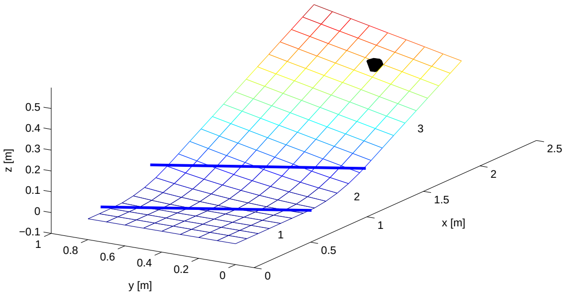


Fig. 5. The geometrical course model.

6.3.2. Model of the ball and the ball trajectory The model of the ball is based upon fundamental physical laws (Alonso and Finn 1980), but the rotational dynamics of the ball are not considered. The approximative equations describing the dynamics therefore become

$$\ddot{x} = -\frac{1}{m}\left(\mu_N F_N \mathrm{sign}(\dot{x}) + \mu_v \dot{x} + F_N \sin(\theta)\cos(\phi)\right),$$

and

$$\ddot{y} = -\frac{1}{m}\left(\mu_N F_N \mathrm{sign}(\dot{y}) + \mu_v \dot{y} + F_N \cos(\theta)\sin(\phi)\right),$$

where μ_N and μ_v are coefficients of friction which are proportional to the normal force F_N and the velocity, respectively. Furthermore θ and ϕ are Euler angles representing the orientation of the course plane in the position of the ball. The variable θ corresponds to rotation around the y-axis, while ϕ corresponds to rotation around the x-axis. The normal force is computed as

$$F_N = mg\cos(\theta)\cos(\phi),$$

where m is the mass of the ball. The initial position is known since the robot itself puts the ball on the course and the initial velocity is found as a constant times the programmed hit velocity. To take into account the ball bouncing on the walls of the course and going into the hole, events have been used in the simulation, leading to a hybrid system. The standard Matlab `ode45` solver, which also supports events, has been used to calculate the trajectory of the ball.

6.3.3. Statistical model The statistical model was created in a three step procedure. In step 1, all angles and velocities in a range −35° to 35° and 3200 to 4500 mm/s were tested by the robot. In the setup described in Section 6.1, the only input to the identification process is the value zero, if the ball has missed the hole, or the value one if the ball has gone into the hole. An exhaustive search algorithm was employed and all combinations of angles and velocities, in steps of 1° and 20 mm/s were tested. In a future project, a camera will also be used as input. This will make the search process much more rapid since the ball trajectory will be available to the identification process.

In step 2, the process continued by returning to areas in the angle velocity space where the robot has managed to hit the hole. Since there are uncertainties in the initial position of the ball, the probability for most combinations of angle and velocity to actually get hole-in-one is less than one. The results from step 2 is a matrix of probabilities for hole-in-one for each combination of angle and velocity.

The last step, step 3, is a continuous process that allows to update the probabilities based upon the result (zero or one) from a hit. In this way the robot has a learning capability which makes it possible to adjust for slowly varying parameters. If a parameter is changed rapidly, for example the ball is replaced with a new ball, the process should start from step 1 again since the statistical model in this case is no longer valid.

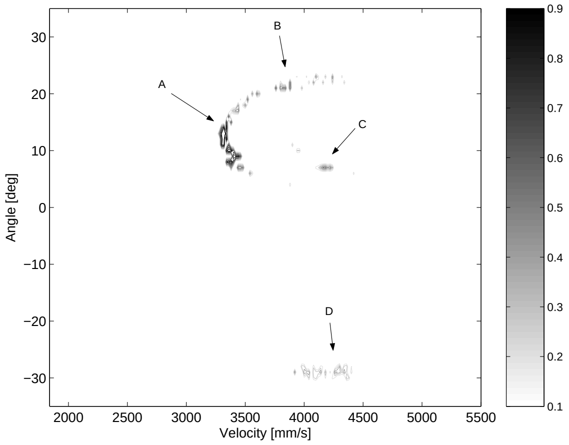


Fig. 6. Probability for hole-in-one for different angles and different hit velocities.

In Figure 6 the result of the statistical model after the project had finished is shown. A number of interesting features can be observed. In the region

A, the ball goes directly into the hole (as shown in Figure 3). Region B contains cases where the ball bounces on the left wall before going into the hole. In region C there are some samples where the ball bounces on the back wall before entering the hole. Finally, region D contains the cases where the ball bounces on the right wall before reaching the hole.

6.4 Robot program

The robot program communicates with Matlab using a simple file interface. The Matlab program sends information such as `action` (collect ball or hit ball), `speed` (hit speed in mm/s), `angle` (hit angle in degrees), and `personality` (on or off). The `personality` flag is used to make the demonstration a bit more interesting, with this option the robot shows *emotions* if the hit is a success or a failure and it also warms up before hitting the ball. After hitting or collecting the ball, the robot program sends `score` (missed or hit the hole) back to the Matlab program.

In the robot program, the golf course is defined as a work-object with an associated coordinate system. All the coordinates for, for example, picking up and hitting the ball are given in the work-object coordinate system. If the golf course is moved with respect to the robot, it is therefore only the work-object coordinate system that has to be calibrated in order to run the program. The hit procedure is done by moving the robot along a circular path. More work should be done in the optimization of the club path to maximize the available speed.

6.5 Future developments

In next year's project, the technical platform given to the students will be developed further. There will be an extension to the golf course where the ball will drop if it goes into the hole. This is illustrated in Figure 7. In this way, the game can continue for more than one shot. The new course also makes it necessary to have a camera to find the ball on the new platform. The camera can also be used in the modeling and identification step and will make it possible for the robot system to see if the course has been moved. The calibration process could at least in principle be done using the camera system. The model of the ball dynamics could be further developed. Although it is quite difficult it would be interesting to take the rotational dynamics of the ball into account. The step when the club hits the ball should be more thoroughly studied, and the ball bouncing on the walls is not covered very well by the current model.

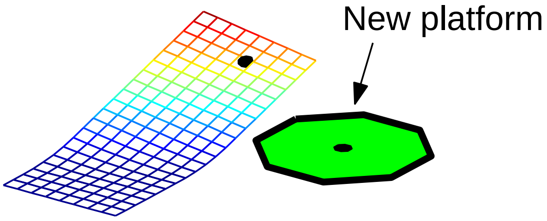


Fig. 7. Course design for the spring 2005 course.

7. CONCLUSIONS

The CDIO initiative has been described and a student project in Automatic Control has been presented. The task of the project has been to develop a control program and a user interface for a golf playing industrial robot. The project has been carried out using the project model LIPS in order to mimic the way an industrial project is carried out.

ACKNOWLEDGMENT

The CDIO Initiative has been sponsored by The Knut and Alice Wallenberg Foundation. The ABB IRB1400 robot has been available via the Center of Excellence, ISIS, sponsored by VINNOVA. Finally, the authors wish to thank the students in the Automatic Control project course and especially the students in the *golf playing robot* project, Lars Stenlind, Elin Eklund, Mikaela Waller, Alexander Nordström, Magnus Nordenborg, Hanna Sjöstedt, and Staffan Ohlsson, for providing the material on which the technical project description is based.

REFERENCES

ABB (1997*a*). *Product Manual IRB1400.* M97A ed.

ABB (1997*b*). *RAPID Reference Manual.* RobotWare 2.0 ed.

Alonso, M. and E.J. Finn (1980). *Fundamental university physics: Volume One, Mechanics and Thermodynamics.* 2nd ed.. Adison-Wesley. Reading, Massachusets.

MathWorks (2002). *Using MATLAB Graphics, Version 6.* MathWorks. 3 Apple Hill Drive, Natick, MA 01760-2098, USA.

Svensson, T. and C. Krysander (2004). *The LIPS project model. Ver 1.0.* Linköping University, Sweden.

The CDIO Standards (2004). `www.cdio.org`.

The CDIO Syllabus (2004). `www.cdio.org`.