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

6,900
Open access books available

185,000
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

Lego Based Computer Communication for Business and Learning

Rapelang Marumo

Department of Mechanical Engineering, University of Botswana, Gaborone, Botswana

1. Introduction

First launched in 1998, LEGO® MINDSTORMS® for Schools (LMfS) and ROBOLAB™, each year have helped countless students grasp science, technology, engineering, and mathematics concepts with hands-on, naturally motivating building sets, programming software, and curriculum relevant activity materials. Hands-on experiences are vital to stimulate the interest of students. One of the key challenges of undergraduate engineering education is providing students with this experience that includes solid theoretical underpinnings and a clear connection to industrial practice. Nowhere is it truer than in engineering, process control, mechatronics and robotics. Just about a decade ago, LEGO introduced a new line of kits called MindStorms that focused on robotics. LEGO MindStorm education is the next generation in educational robotics, enabling students to discover science, technology, engineering and mathematics in a fun, encouraging hands – on way. At the heart of these kits is the microcomputer called the RCX brick. It is a completely programmable LEGO brick with three power outputs and three sensor inputs. Students want to know what awaits them when have completed their mathematics, computer studies and science courses. By combining the power of the LEGO building system with the LEGO MindStorm education technology, teams of students can design, build, program and test robots. Showing students interesting future possibilities will help motivate them and encourage them to continue, where many students might otherwise drop or change fields. LEGO MindStorm is the ideal equipment to do the trick. This unit accomplishes three goals; it creates interest in robotics, engineering and science and it also encourages creativity and team work among students. Working together on guided and open ended engineering projects, the team members develop creativity and problem solving skills along with other important mathematics and science knowledge. Students also become more skilled in communication, organization and research, which help prepare them for future success in higher levels of schooling and in the work place. Although the MindStorms kit is a wonderful way to teach students about robotics, it is limited to a dry environment, such as class room floor. For the past decade, the MindStorms have been used in computer science resources to teach concepts from object-oriented programming (Barnes D., 2002) to artificial intelligence (Klassner F., 2002). They have been used to introduce students to the exciting world of computers (Weisheit F., 2004). The LEGO robots are often said to increase the interest of students in computing courses and they make class work more fun and exciting.

1.1 Problem statement

Technology has advanced a great deal since the beginning of the 20th century. The workplace and especially the industry have had evident and visible use of modern technology in handling operations and conducting other works. Today, the main work a human being does is manufacturing or designing and programming a device to perform certain tasks.

From another point, if students have to make projects in the machine shop, they cannot make very complicated parts. With Lego bricks, they can prototype far more quickly, and the complexity of the part correspondingly increases. The same is true with the software such as LabVIEW that allows rapid software prototyping and students can develop much more complicated software without many of the difficulties associated with a syntactical language. By letting students create the experiment themselves, rather than giving them a canned experiment, one can get them more excited about and more involved in what they are learning.

1.2 Objective of the research work

Objective of the project are:

- To learn the basic interfacing of different software; LabVIEW and Mindstorm NXT in modeling a simple industrial tool to carry out and execute assigned duties.
- To apply the virtual instruments and the Lego mind storm in the workplace.
- To advance technology, especially in the developing countries, in making it known to the people that a human operated task can be performed by programmed equipment.
- To take the engineering technology to the people i.e. promoting engineering for the society.

In this work, the problem of developing a laboratory kit that allows students to go through all steps of synthesizing a control system, a dynamic system and a process plant is addressed. Students assembled the process they were controlling, including placing sensors and control valves, using a collection of process units, pipes and fittings that have simple quick release connectors. It must be noted that to accomplish these student learning goals, the laboratory kits should be an inexpensive flexible system that:

- Can easily be used for open ended projects
- Are inexpensive enough that multiple steps can be easily purchased
- Are portable
- Require only standard power and water so setups can be used outside traditional laboratory facility
- Can be used as a lecture demonstration or active learning exercise in a regular class session
- Are simple and safe enough to be used by unsupervised students for out of class assignments
- Allow for application to various other engineering classes in future

1.3 Scope and summary of the research work

With e-learning emerging from the shadow of its post-dot.com readjustments, an entire generation of new practitioners is being drawn to e-learning for the first time (Sehurutshi et., al 2009). The virtual instrument and Lego Mindstorm is a laboratory set up which is done to teach various people on how complicated projects can be made easier with the use of toys called Lego bricks. The NXT is actually a little computer and interface unit that you can

program to take sensor input from up to three inputs and react to that input by controlling the direction and speed of up to three motors. Combined with the technical parts, you can quickly build and program sophisticated, mobile, autonomous robots.

In this project, Lego bricks are built to perform different tasks. The NXT robot was programmed manually with different configurations. Each sensor was tested with a short program and it worked very well. Upon testing all sensors, (sound, light, ultrasonic and touch sensors) the software was used to develop exercises comprising of a combination of different sensors. The results of testing the brick manually and with the NXT software were the same.

The main components of the project are LabVIEW and Mindstorm NXT softwares. It is with these softwares that our focus is on. The NXT is actually a little computer and interface unit that that can be programmed to take sensor input from up to three inputs and react to that input by controlling the direction and speed of up to three motors. Combined with the technical parts, sophisticated program can be built quickly to control mobile, autonomous robots. It is an autonomous Lego microcomputer that can be programmed using a PC. The NXT serves as the brain of Lego Mindstorms inventions. It uses sensors to take input from its environment, process data, and signals output motors to turn on and off. Users first build their robot using the NXT and Lego elements. They then create a program for their invention using NXT code, a simple yet powerful programming language. Next, the program can be downloaded to the NXT using a special infrared transmitter. The creations can now interact with the environment, and fully independent.

1.4 Related work

In 1989 Martin created the MIT Robot Design course following from Flowers' Introduction to Design course that was offered in the Mechanical Engineering department. The work on this project culminated into a textbook (Martin F., 1989). Students learn about the basics of building robots from kits and the course ends with a contest. Yanco in 2001 has adopted this course using the Bot-ball game as the tournament at the end of the term (Martin F., 2000). Mataric in 1998 has developed an award winning course called Introduction to Robotics (Yanco H., 2001) which takes a hands-on approach to the field of robotics. Students use both Handy board microcontroller and Lego MindStorms system. Another introductory course on robotics that uses Lego MindStorms is the Building Intelligent Robots course taught by Dean at Brown in 2001 (Martaric M., 1998). A few people have developed courses using hands-on robotics that do not focus on teaching robotics as the main subject. Littman's course on Programming Under Uncertainty (Dean T., 2001) teaches about a variety of methods for programming as its title says, under uncertainty, including Markov Decision Process and POMDP's and variety of machine learning techniques like reinforcement learning and genetic algorithms. Students in this course used Lego robots to demonstrate their knowledge to the methodologies studied. The course ended in a project, where some of the students developed their own application for their robots, from line-following task to making breakfast. One of these, described in (Littman M., 1999), carried out on-line reinforcement learning to complete a task analogous to the pole-balancing-an indication of what is possible at the upper limit of the MindStorms capabilities. Since 2001, Klassner has been teaching introductory artificial intelligence using Lego MindStorms and Russel & Norvig followed in 2003 (Baum D., 2000b and Klassner F., 2001) . Klassner's students make extensive use of the robots and are supported by tools developed by Klassner and his colleagues (Klassner F., and Anderson S 2003). These extend the capabilities of the RCX, making use of the infra-red communication built into the unit (normally used for

downloading programs) to allow off-board control. The description of the robot blocks maybe as shown in Table 1.

First Block	Introduction the basics of MindStorms, such as the RCX, RCX code, motors and sensors. Construction of a robot that uses a single touch sensor to avoid objects
Second Block	Introduction to the NQC (Not Quite C) and the RCX command center (RCXCC) Program first robot using the RCXCC and modification to use two touch sensors to go around obstacles
Third Block	Construction of robot that is capable of following a black line on a white paper using RCXCC
Fourth Block	Introduction to the datalog on the RCX Introduction to the first major project, students must map the reflectivity of the bottom of a prepared box
Fifth Block	Completion of remote sensing project
Sixth Block	Introduction to the final project; the creation of a robot animal, an environment, and a non-Lego component of the robot
Seventh Block	Presentation of the results

Table 1. Description of robot blocks.

1.5 The heart of Lego Mindstorm

The RCX is capable of outputting a maximum of 700mA and 9V through each of its power outputs. This translates into 6.3 Watts, which is very small amount of power. Because of this limited supply, LEGO manufactured motors are the most efficient to use for the ROV because they are already rated for such a low power level. LEGO MINDSTORMS for school is a modular concept for teaching groups of children from pre-University level in a classroom or after-school club environment. It is made up of core construction and add-on resource sets, a software program, and curriculum relevant activity packs.

1.6 The next technology now

LEGO MindStorms education features an advanced 32-bit computer controlled NXT brick, interactive servo motors, sound, ultrasonic and other sensors, Bluetooth communication and multiple downloading capabilities. The icon-based LEGO MindStorm education NXT software is built on the Lab View software from National Instruments, an industry standard with applications in many engineering and research fields. The LEGO MINDSTORMS for school software, called ROBOLAB, is developed specifically for teachers or adults working with groups of children. The program is icon-based, which means that youngsters can visualize the instructions they are stringing together for their robot. It provides basic, intermediate, and advance programming techniques including data logging.

1.7 Introduction to LEGO Mindstorms for schools & ROBOLAB overview

ROBOLAB is a powerful programming language that can be used by students at all levels. It was developed by the partnership between Tufts University, LEGO Education and National

Instruments. The ROBOLAB software is based on Lab VIEW, the software that was used to control the Sojourner rover on Mars in 1997. In Lab VIEW, programs are called virtual instruments (VI). In the professional edition of Lab VIEW, the software uses a computer as a virtual instrument. The Lab VIEW program is then used to acquire and analyze data. ROBOLAB programs are saved as .vi files. There are different programming levels within ROBOLAB. The Pilot and Inventor sections of ROBOLAB are based on the Lab VIEW programming language. It is an icon-driven language based on a logical sequencing if images and is basically independent of written language.

The Pilot section of ROBOLAB is made up of a series of templates that have a format associated with them. It is an easy way to introduce the logical sequence to students. The user can not modify the templates and therefore the program will always work. The program will always run each time and do the commands as listed in the sequence presented. The Inventor section of ROBOLAB uses the same command icons as in Pilot. There are more command icons as the user moves into higher levels of the program. Inventor is less structured allowing for greater potential for the software. The Investigator section allows users to collect and analyze data and create projects. It includes all the features of the ROBOLAB Programmer. Project management features are built into ROBOLAB Investigator. They include all the components needed to investigate questions with the RCX inventions. This includes writing programs, uploading data, viewing data, computing with data and documentation of that data. This software runs on either PC or Mac. The PC version requires Windows 98. There are several new features in this version:

USB support

Piano Player - updated musical support

Extended help features

Digital camera interface

New RCX firmware

Web support

Auto wiring of icons

Pilot - level capability for uploading a program from RCX to computer

The RCX Brick (Robotic Command Explorer)

As already mentioned, the heart of the system is the RCX, i.e. an autonomous LEGO microcomputer that can be programmed with either a PC or Mac computer. The RCX can be considered the "brain" of the system. It uses sensors to take input from its environment, to process data and to signal output motors and lamps to turn on and off. Students build their device using the RCX and the LEGO pieces in the building sets. They then create a program for their device using the ROBOLAB programming language. They then download their program to the RCX using the infrared transmitter. Their device can now interact with the environment, totally autonomous of the computer. There are 3 output ports - A, B, and C. These are connection points for LEGO motors, lights and other sensors. There are 3 input ports - 1, 2 and 3. These are connection points for sensors such as: touch sensor that turns the motors on and off when activated. Light sensor that allows RCX to differentiate between light and dark. Temperature sensor that distinguishes between varying degrees of temperature. Additional sensors are available for data logging.

The RCX brick has an infrared eye that allows the brick to communicate with the computer via the infrared transmitter (IR). It must be aimed towards the IR transmitter to function. Firmware is the RCX operating system level software. It must be downloaded to the RCX before the RCX can receive and execute a program from the computer. The firmware

remains in the RCX memory until the batteries are removed. If batteries are changed quickly, this will not affect the memory. An external power supply can also be used to save on batteries.

Fig. 1. LECO MindStorm.

The Lego MindStorms can yell & spin, catch & run, record & play.

Fig. 2. MindStorm NXT.

What it comes down to is that, with the combination of the mechanical flexibility of the Legos and the connectivity and ease of programming of the brick, students are no longer limited by execution and even those without masters in mechatronics can take crazy ideas and make them into a reality. These applications can all be downloaded from the MINDSTORMS website, and they represent just a small sampling of the crazy inventions that these Lego users are devising when they are limited only by their imagination.

2. Research procedures

The MINDSTORMS® equipment is located in the Instrumentation and Control laboratory room 2069/2 of Mechanical engineering Department. This project reflects the results of a review of literature about engineering classes taught at other Universities and proposed courses for the University of Botswana, Faculty of Engineering & Technology in the Department of Mechanical Engineering. Many Universities have incorporated MindStorms into introductory engineering classes because of their ease of use, relative cost effectiveness and ability to give hands – on experience with engineering projects while encouraging creative processes.

2.1 Control paradigm

In manual operation, the RCX delivers a constant 9V to each of its power outputs, and the remote operated vehicle (ROV)'s motors are controlled by switching the current flow with Double Pole Double Throw (DPDT) rockers switches. When the rocker switch is in the neutral position, no current flows. When it is switched forward, the current flows and the motor turns forward. When the DPDT is switched backward, the poles are reversed, resulting in the current flowing backwards, making the motor to turn backwards. Additionally, in each motor's control circuit is a manual override Double Poles single Throw (DPST) switch to give RCX direct control of the motor. With this switch, autonomous control can be turned on and off for motor individually, so the operator has the option of the RCX is controlling one function while another function controlled manually. One application of this ability would be the (ROV) hovering. The RCX would monitor the pressure and adjust the vertical thrusters while the operator could still manually drive in the horizontal plane.

The general control box design is a container for the RCX that is held with two hands on either side. The top has two horizontal motor control switch sets in reach of the thumbs in addition to the three switches. Each subroutine switch hooks directly into one of the sensor inputs and so when it is tripped it shorts out that input. In the ROBOLAB program a shorted input will return the maximum value of the sensor programmed to be hooked up to that input. When the maximum value is reached, the program can be triggered to do the task. This makes it possible for a manual switch to run a subroutine. The front of the box has the connection to the tether as a well as the vertical motor control switch set which is in reach of the index fingers. Additionally, the box has an AC adaptor input on its left side.

2.1.1 Control algorithm

As mentioned earlier, the LEGO RCX brick provides the computer interface. Control systems were constructed using ROBOLAB investigator software which is an adaptation of the National Instrument's LabVIEW software. The sensor is read and its raw reading is converted to 0 to 100% range. This signal is sent to a discrete (Proportional Integral Derivative) PID velocity algorithm (Riggs, James B, 2001), which outputs a change in valve position. The signal from the controller can be positive or negative depending on which way the motor should go. The programs integrate both LabVIEW and ROBOLAB languages.

2.1.2 Data logging program

This program's function is to collect data from a sensor connected to the RCX and to display that data in real time. Once the data collection is completed, the user may save it to a text file

in order for it to be used for analysis. Typical examples that students performed include step test (sudden increase in control valve position) and height of the liquid in the tank recorded by he data logging program of the Siemens programmable controllers (PLC). The program also allows for user control of a flow control valve so that, a change in valve position can be logged along with the resulting sensor reading. The data could be transferred to control station (Cooper, Doug, 2002) or a spreadsheet and a First Order Plus Dead Time model for the system can be obtained.

Another feature of this program is its calibration element. The user may enter a maximum and minimum calibration value to coincide with the raw value of 0 and 1023. Raw numbers are the values recorded by the RCX when receiving a signal from a sensor. The user can get the computer to output the actual raw values by altering calibration values in a range of 0 to 100% of a given attribute (i.e. full scale defection, span or reading) or the actual height of a liquid in a vessel.

2.1.3 Control program

The program collects data from a sensor, enters the value into a PID equation, which outputs a value to represent necessary valve position change. Like the data logging program, all of this occurs within a while loop which runs until the user presses the stop button. The front panel consists of a chart, which displays the calibrated sensor value and another chart, which displays the motor outputs. The front panel also has fields which enable the user to control setpoint as well as the tuning parameters of the controller.

2.2 LEGO education resources (eLab example)

The LEGO Education eLAB series introduces students to concepts of energy; allowing them to take the next natural step in the development of their scientific and technological knowledge and experience.

With a unique combination of LEGO® System sets, a motor, solar panel and capacitor, students can investigate how to generate, store and use sources of energy. The eLAB Renewable Energy Set allows students to build and test realistic models of wind and water mills, and a solar-powered Ferris wheel.

LEGO Education offers a scheme of work to use with the Renewable Energy Set, consisting of 49 activities designed to help students explore, investigate and solve problems.

Find out how to use the eLAB series together with the intelligent LEGO brick, the RCX, which features in the LEGO® MINDSTORMS® for Schools series.

2.2.1 Science & technology

Science and Technology solutions from LEGO Education help teachers to capitalize on students' natural curiosity and enthusiasm about the world around them. Tools and design tasks for exploration and experimentation are provided, so that teachers can introduce the basics and let students try out their own ideas. Students build structures and simple machines and mechanisms, and test forces and motion. They experiment with the functions of wheels and axles, gears, levers and pulleys. They investigate the effects of friction and gravity on speed, and explore sources of energy. The series starts with a range of LEGO® DUPLO® construction sets for year 2 dynamics (MMB 222 course) students and continues with LEGO® System and eLAB sets that introduce motorized power and other forms of energy such as wind and solar power, for students in subsequent years.

3. Lego MindStorm as teaching tool for mechatronics, measurement & instrumentation courses

Information and Automation Technology will be part of the future careers of virtually all students in Mechanical Engineering, Faculty of Engineering & Technology (FET of the university of Botswana (Marumo., 2007). Traditionally students are exposed to a course in a higher order computer language, which is perceived by many students as unexciting and tedious. To introduce students to automation in a far more challenging and exciting manner, LEGO Mindstorms will be used in the following courses: measurement and instrumentation and mechatronic, robotics, process control and systems & control. The LEGO Mindstorms kits contain a programmable brick which essentially is a micro controller built in a plastic LEGO package with three connectors for actuators (motors) and three sensor inputs such as switches, optical analog sensors, angle encoders etc. The micro controller has an infrared serial link through which programs can be uploaded and executed. The brick also has buttons to start, stop, and select programs and to check the status of actuators and sensors in real time. The Mindstorms brick can be programmed in a variety of languages. LEGO supplies its proprietary pictorial ladder-style language, there is NQC (Not Quite C), which is a C-derivative, and advanced users can even use C/C++ that runs on a dedicated operating system called BrickOS. Robolab is a pictorial programming environment, where actuators, sensors and control structures are joined together has shown to be very suitable for students with no procedural language background. The reason for selecting this language was the widespread use of Lab View in industry and the basic philosophy behind it. Users select components and attach modifiers to change their properties. Components are linked using a wiring tool very similar to building electrical circuits.

3.1 Mechatronics, measurements & instrumentation courses

The objectives of the LEGO Mindstorms projects are:
- To introduce the concepts of Automation to undergraduate students in B Eng Degree programmes
- Develop communication skills among students
- Exercise critical thinking and engineering design principles
- Promote enthusiasm for studying engineering
- Develop skills of graphical programming
- Enforce precision automation concepts
- Promote creativity and engaged learning
- Experience the feeling of accomplishment by completing a group design

The mechatronic design projects assigned in mechatronics course involve students' teams designing and developing mobile robotic platforms using Lego components. The activities will not only inspire students' creativity but also enhance their strength in engineering design course

3.2 Pilot programming

The RCX programming software has two options or phases. These are Pilot and Inventor. Pilot is the introductory section of the ROBOLAB software. A "click and choose" interface is utilized within a template. There are four levels of Pilot and these serve as an introduction to

ROBOLAB programming. Level 1 is the easiest and Level 4 has more flexibility. Each level builds on the previous one. There are limitations within this section but it is a simple way for users to become familiar with the software. The Pilot Level 1 template is a simple task that introduces students to the programming features of ROBOLAB. Each level of Pilot provides additional flexibility and options. Students will always be successful in their programming within Pilot.

4. Dynamics course

The design of a low-cost balancing robot is accomplished by using a popular development platform plint pack for the components and Lego for the construction material. The balancing robot is intended to improve the learning experience of students or hobbyists who are interested in control of mechanical components by presenting hands-on experiments and controller design examples that anyone would be able to follow easily and cost effectively.

4.1 System description

The system consists of motors, wheels, and two pairs of spur gears that connect the motors and the wheels. The spur gear was added because the motor shaft was not long enough to support the distance required by the optical encoder. The optical encoder requires some space between the striped disk and the optical sensors to read black and white. The top stage consists of a microcontroller and an H-bridge. The microcontroller is the brain of the robot and is responsible for sampling and processing all the data collected by the sensors. The H-bridge is used with the microcontroller to control the directions of the motors. There is one more sensor attached to a vertical strut between the two stages. In general it is called an Inertial Measurement Unit, (IMU), and this one contains two accelerometers and one gyro to determine the robot's attitude in one angular direction.

4.2 System modeling

To make the project simple and manageable, only a two-dimensional movement in which the system moves in one plane will be considered. The two motors are assumed to move together symmetrically when an identical input is given. Although this is unlikely the case, it is a good approximation for motors which are manufactured as the same product. A control voltage is applied to the motors, which results in displacement of the wheels and change in angular attitude of the body. A mathematical model for the system is obtained by observing the free body diagram.

4.3 Actuator assembly dynamics

There are two fundamental equations describing the characteristics of a DC motor. The first one relates torque (τ) on the shaft proportional to the armature current (l) and the second expresses the back emf voltage (e) proportional to the shaft's rotational velocity (ω_{enc}), where ω_{enc} represent the velocity measured by the encoder mounted on the same frame as the DC motor. This velocity represents the shaft velocity with respect to the DC motor itself.

$$\tau = Ki \tag{1}$$

$$e = K_e \omega_{enc} \tag{2}$$

4.4 Body dynamics

The rotational motion of the pendulum body about its center of gravity can be described by

$$I\ddot{\theta} = -2\tau + -2Hl\cos\theta + 2Vl\sin\theta \tag{3}$$

where I is the moment of inertia of the pendulum body about its center of gravity. The horizontal force, vertical force and torque are applied as pairs on both sides and each pair is assumed to be equal because only two-dimensional motion was considered.

The horizontal motion is given by:

$$2H = M\frac{d^2}{dt^2}(x + l\sin\theta) \tag{4}$$

The linearised equations are:

$$I\ddot{\theta} = -2\tau - 2Hl + 2Vl\theta \tag{5}$$

$$2H = M(\ddot{x} + l\ddot{\theta}) \tag{6}$$

$$2V = Mg \tag{7}$$

4.5 System dynamics

The objective here is to find suitable ordinary differential equations that fully describe the system in terms of input v, the states x, θ and their derivatives $\dot{x}, \dot{\theta}$. The following formulae describes how to characterize the essence of the system in two differential equations.

$$\tau = \left(\frac{J}{r} + rm + \frac{rM}{2}\right)\ddot{x} + \frac{rMl}{2}\ddot{\theta} \tag{8}$$

$$\frac{R}{K}\left(\frac{J}{r} + rm + \frac{rM}{2}\right)\ddot{x} + \frac{R\tau Ml}{2K}\ddot{\theta} + \frac{K}{r}\dot{x} - K\dot{\theta} = \upsilon \tag{9}$$

$$\left(I + rMl + ml^2\right)\ddot{\theta} + 2\left(\frac{J}{r} + rm + \frac{rM + lM}{2}\right)\ddot{x} - Mgl\theta = 0 \tag{10}$$

Finally the overall equation yields:

$$\left(I + rMl + Ml^2\right)\ddot{\theta} + 2\left(\frac{J}{r} + rm + \frac{rM + lM}{2}\right)\ddot{x} - Mgl\theta = 0 \tag{11}$$

These equations describe the motion of the system and constitute the two-dimensional linearized mathematical model of the system. Finally this set of equations can be expressed in a more convenient state space form:

$$\begin{bmatrix} \dot{x} \\ \dot{\theta} \\ \ddot{x} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & A_{32} & A_{33} & A_{34} \\ 0 & A_{42} & A_{43} & A_{44} \end{bmatrix} \begin{bmatrix} x \\ \theta \\ \dot{x} \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ B3 \\ B4 \end{bmatrix} \upsilon \tag{12}$$

The unknown properties, J, M, m, r, K and R should be obtained by direct or experimental measurements.

4.6 Sensors

In here, inertia measurement unit (IMU) is developed by an open source approach from the Kalman filter. The graphs below illustrate how the Kalman filter can be useful in fusing single-axis angular attitudes collected separately. Ideally a jig would be required to provide an absolute reference angle at any given moment, but since the data will only be used as an observational reference and not in the core algorithm itself, it is reasonable to do without a costly jig and roughly compare data from the two sources, accelerometer unit and gyro unit and ultimately with the Kalman filtering results
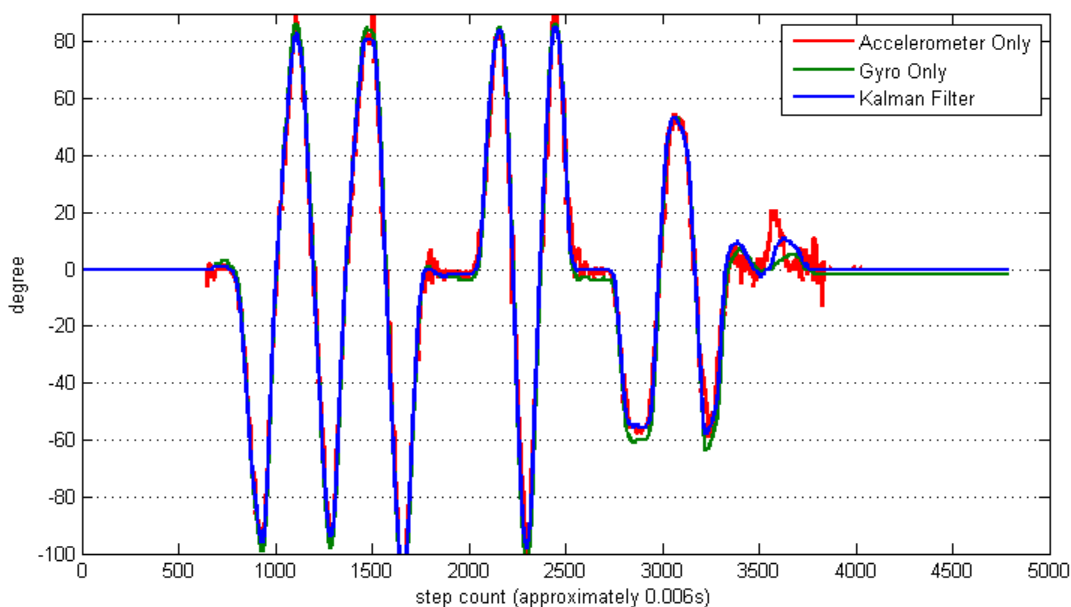
Fig. 3. Steady motion using Kalman filter.

Figure 3 represents a response to a steady movement. The Y-axis indicates the angle in degrees and the X-axis is the step count or sampling count from the microcontroller. Usually when implementing a microcontroller a task is programmed to be executed in a routine basis, but the experiment may also be operated in an endless loop without any delay time allowing the Euler Integration to produce higher quality results. It can be observed that both units reflect the angle quite similarly. The experiment was not long enough to catch the drift of the gyro but overall all data seemed to be reliable. So it was necessary to provide a harsh environment to intentionally invoke degradation in data. The next experiment was performed in a hard shaking circumstance.

Figure 4 shows interesting results. The IMU board was not only rotated radically but also shaken in order to generate lateral acceleration. The lateral acceleration introduces noise to the system so that by using accelerometers alone one cannot measure where about the angular attitude is. At the end of the shaking process the gyro reveals an offset compared to the angle provided by the accelerometer. In this case the accelerometer is more reliable and the offset can be considered as an integration propagation error or a continuous drift angle.
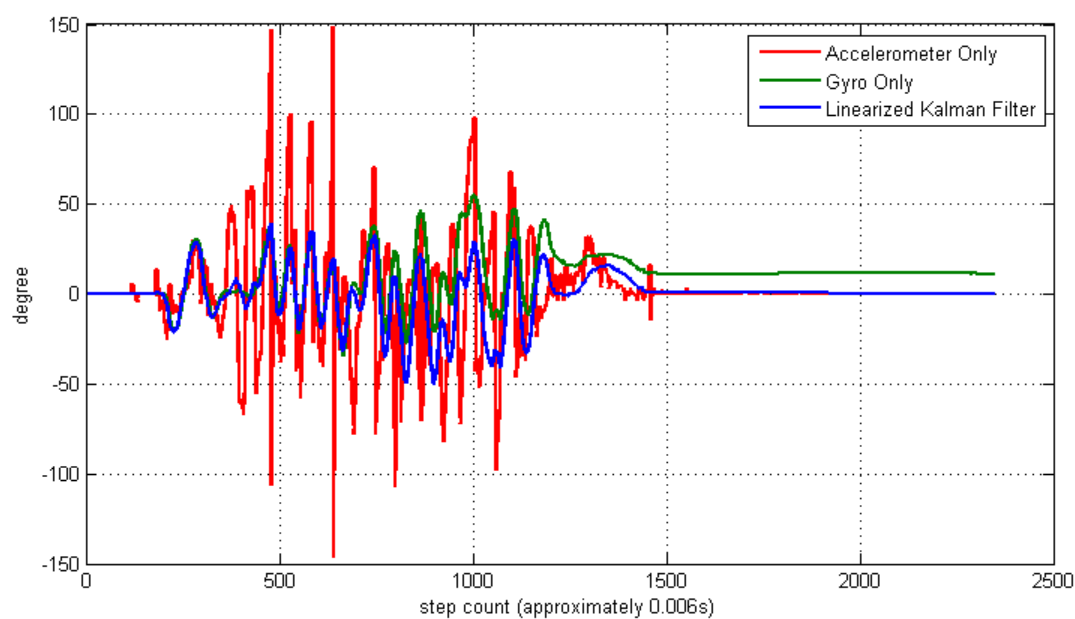
Fig. 4. Unsteady motion measurement using Kalman filter.

It is very interesting to observe that the Kalman filter is following well the patterns of the gyro in a very noisy environment, whereas, at the end of the shaking process the sensor is stable; it pertains to the more accurate accelerometer data.

5. Digital control for the motor position

One of the easiest ways may be through implementing a controller for the DC Motor. A digital controller for position reference will transform the DC motor to behave like a servo. The code for the controller was generated using the MATLAB Real time Toolbox Embedded encoder. To confirm the DC motor model, a simple proportional control was designed and implemented.
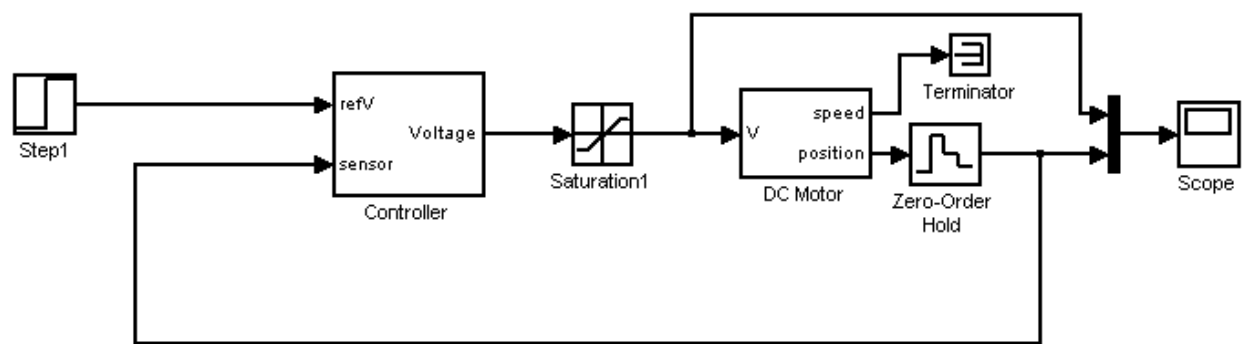
Fig. 5. Block diagram for motor position control.

Figures 6 and 7 illustrate the simulation result versus the real data. The rise time is shorter in reality. This is probably because of the sensor dynamics that was not considered in simulation. In reality the encoder has large quantization error. When conducting this experiment 64 lines were used by the encoder. This means 5.6 degrees was the maximum

precision the sensors could have which is quite poor. After this experiment the sensitivity was increased by a factor of 2 to achieve greater accuracy in position and velocity values. The gain value used in this experiment was 2, and the step input was 90 degrees.
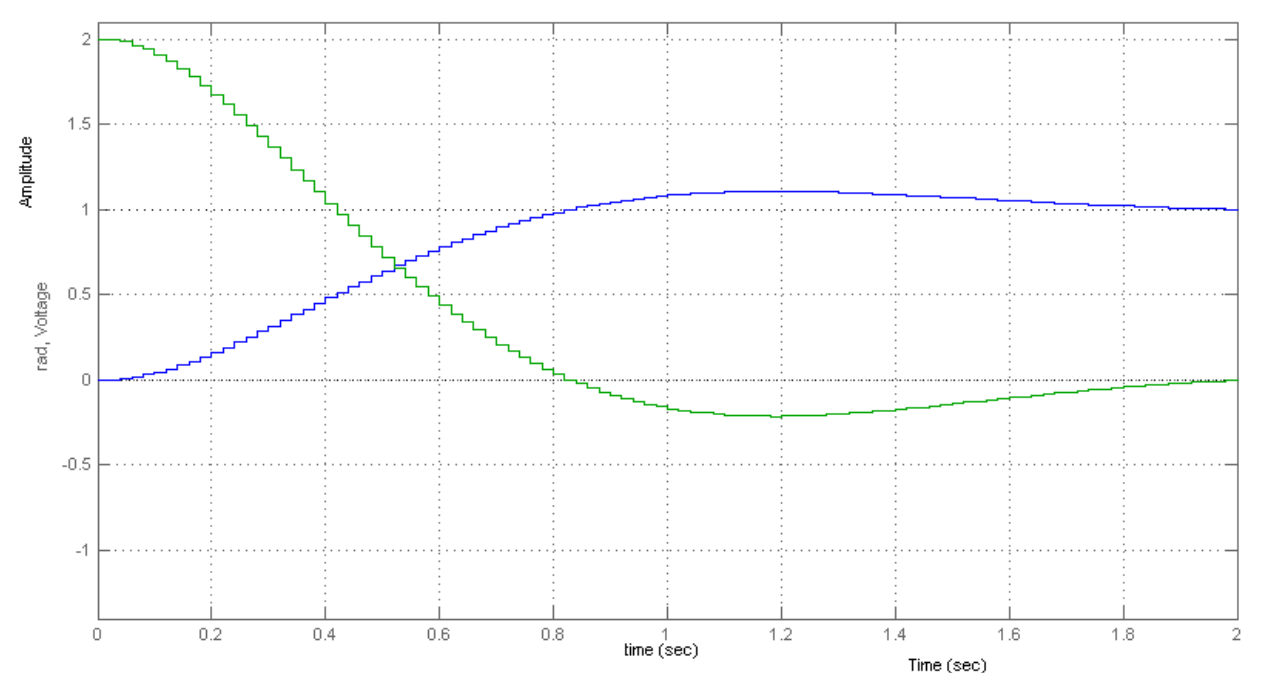
Fig. 6. Simulated results versus real data - continous.
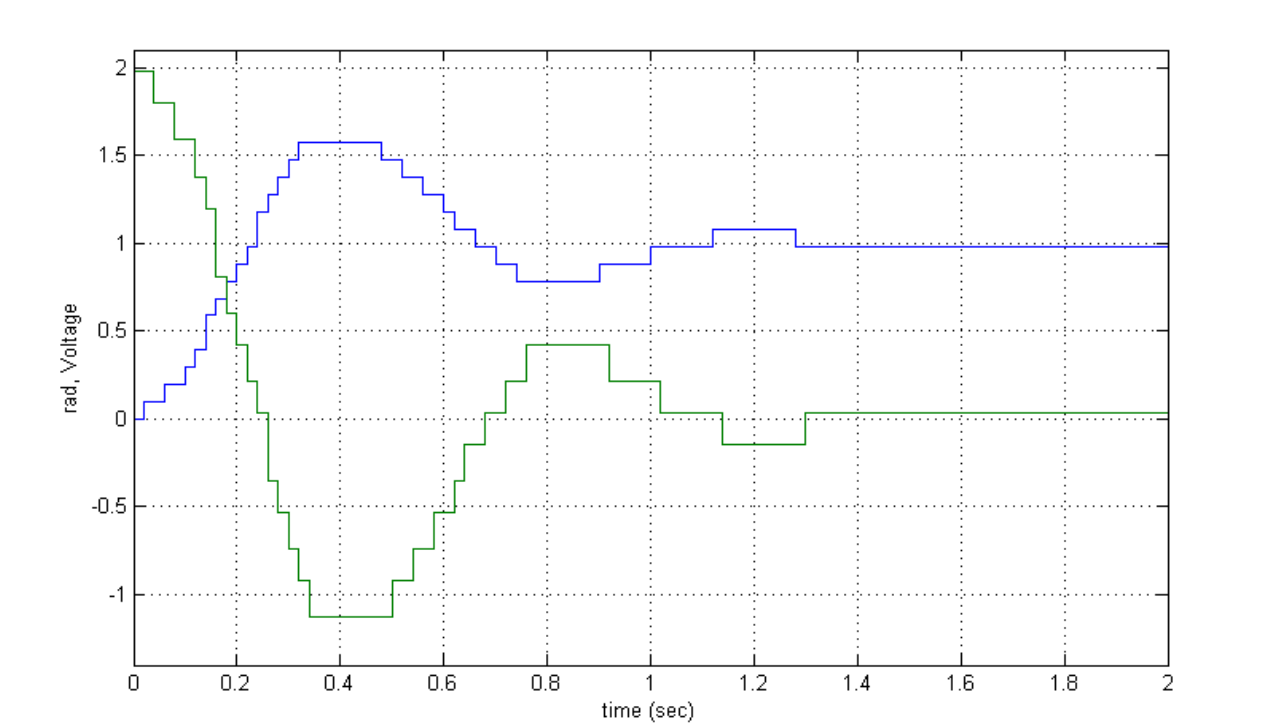
Fig. 7. Simulated results versu real data - digital.

6. Digital control root locus

In this experiment Control System Toolbox in Matlab, more especially 'sisotool' was used to design a Root Locus Controller by applying a gain and placing poles and zeros. Several designs were devised that looked robust in theory, but performed not as well. The difference of theory and practice became prominent as the solution required greater control effort. While the control design technique was intended for linear systems, the effect of saturation was becoming more outstanding in the real world.
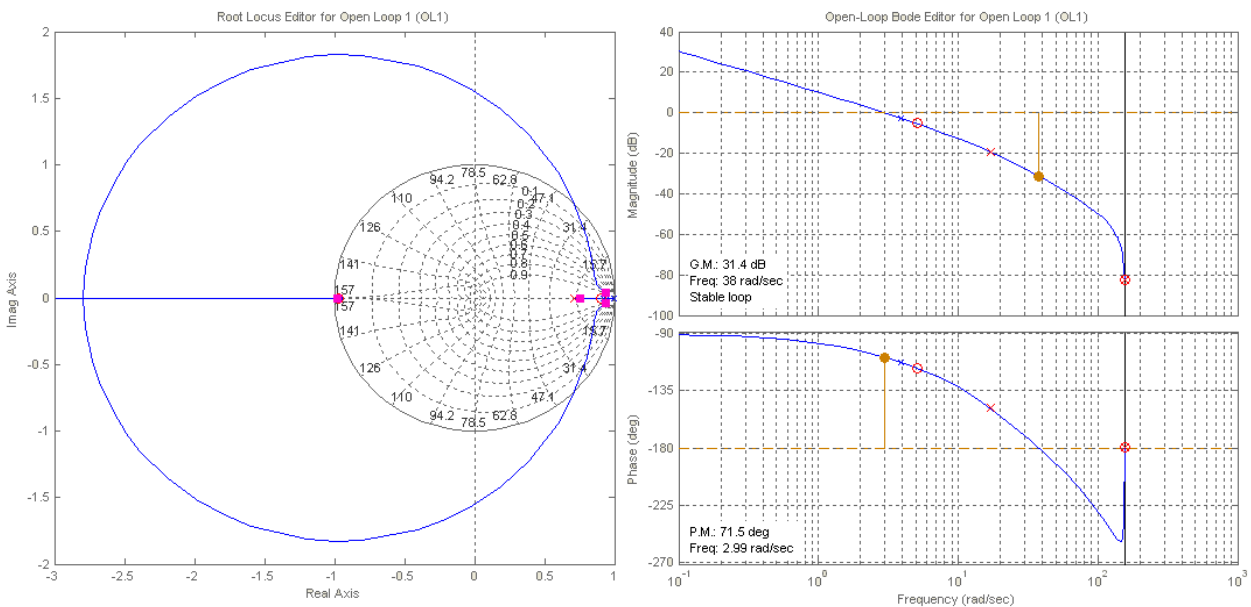
Fig. 8. Root locus plot of motor position control.

One of the working designs is presented. The control effort was designed to be kept below the saturation point; however the same quantization error that occurred in Proportional Control experiment was also observable here as well.

7. Hands-on artificial intelligence education

Educational researchers have long suggested that instruction utilizing a variety of delivery modes helps students with differing learning styles to better understand the studied material. Few, if any of us, would think about teaching our introductory programming courses solely via textbook readings and lecture. It is firmly believed that, to learn, students must "do." Thus, a variety of assignments and "labs" are provided that allow students to participate in both reflective practice in the classroom, and active practice in the computer lab. Thomas et al. (Thomas L., Ratclife M., Wodobury J., and Jarman E.,) have studied the success rate in traditional computer science courses by students exhibiting each of the

different learning styles described in the work of (Richard Felder) and suggested that instruction targeted across styles greatly improves the performance of students from all styles, but in particular from those styles less prominent in the domain of computer science.

8. Conclusion

Lego Mindstorms kits have been be used as a tool to introduce automation concepts at the University of Botswana's undergraduate level courses in engineering. Robolab, which is a programming environment similar to Lab View, helped to enhance programming skills and strength for students in measurement & instrumentation course MMB 314. The acquired skill can then be used to programme very small components such as 8-pin PIC16F84 micro controller used by mechatronic students in course MMB 416. It is therefore evident to conclude that by tapping into the competitive human spirit and the fun of playing with Lego Minstorm robotic development will help to enrich the emphasis on the merit of using automation technology in precision instrumentation and mechatronics.

9. Acknowledgements

It is recommended that the Department avails the necessary softwares in the rightful places to increase access to students when the need arises. There are currently four computers loaded with both NXT Mindstorm and LabVIEW softwares. The same computers are fully utilized when such softwares like Gambit, MATLAB, LabView and CAMaster are being used and when there are iterations going on, it is impossible use the computer for at least 24 hours. One thing which has puzzled us a lot is the fact that the PCs in the CAD/ cam center have loads of softwares and they usually will complain of low virtual memory. This really makes working difficult because amongst other things, the chances of these computers executing tasks successfully are very low.

It is further requested that the Department should avail all the items like; PLCs, DAQ plug in boards and transducers for signal conditioning- these will enable successful working environment with such VIs like LabVIEW.

10. References

Arai, T. & Kragic, D. (1999). Variability of Wind and Wind Power, In: *Wind Power*, S.M. Muyeen, (Ed.), 289-321, Scyio, ISBN 978-953-7619-81-7, Vukovar, Croatia

Barnes, D., Teaching Introductory Java through Lego Mindstorms Models. Proceedings of the 33rd SIGCSE technical Symposium on Computer Science education (SIGCSE 2002), Covington, Kentucky, 2002, pages 147-151

Baum, D., ed. 2000b. Extreme Mindstorms: An advanced guide to Lego Mindstorms: A press, New York

Cooper, Doug., 2002. Control Station Software for Process Control Analysis, Tuning and Training, Storrs, CT

Dean, T., 2001. CS148: Building intelligent without representation. Artificial intelligence 47:139-59.

Felder, R., Learning style strategies home page,
http://www.ncsu.edu/felder~public/ILSdir/styles.htm
http://sciyo.com/articles/show/title/wind-power-integrating-wind-turbine-generators-wtg-s-with-energy-storage

Klassner, F., 2001. Introduction to artificial intelligence.
http://www.csc.vill.edu/`klassner/csc4500

Klassner, F., A case study of Lego Mindstorms Suitability for Artificial Intelligence and Robot Course at College the Level Proceedings of the 33rd SIGCSE technical Symposium on Computer Science education (SIGCSE 2002), Covington, Kentucky, 2002, pages 8-12

Klassner, F., and Anderson, S., 2003. Lego Mindstorms: Not just for K-12 anymore. IEEE robotics and automation (in press)

Lego mindstorms education NXT user guide manual

Li, B.; Xu, Y. & Choi, J. (1996). Applying Machine Learning Techniques, *Proceedings of ASME 2010 4th International Conference on Energy Sustainability*, pp. 14-17, ISBN 842-6508-23-3, Phoenix, Arizona, USA, May 17-22, 2010

Lima, P.; Bonarini, A. & Mataric, M. (2004). *Application of Machine Learning,* InTech, ISBN 978-953-7619-34-3, Vienna, Austria

Littman, M., 1999. CPS196: Programming under uncertainty.
http://www.cs.duke.edu/`mlittman/courses/cps196

Martaric, M., 1998. CS 445 Introduction to robotics, a Lego-kit-based hands-on lab course.
http://www-scf.usc.edu/-csci445/

Martin, F., 1989. 6.270: The MIT Lego robot design project competition.
http://fredm.www.media.mit.edu/people/fredm/projects6270

Martin, F., 2000 Robotic explorations: A hands on introduction to engineering. Prentice Hall.

R. Marumo (2007), A proposal to Introduce LEGO® MindstormsTM Teaching Aids into University of Botswana's Engineering Programmes. Proceedings of the Botswana Institution of Engineers conference. Gaborone. Botswana, on CD, 17th – 19th October.

Riggs, James B., 2001. Chemical Process Control, Ferret Publishing, Lubbock TX, pp 221

Sehurutshi R, Marumo R and Wangombe K (2009), E-learning Platform for Education Innovation: A Case for Botswana. E-learning Journal. Volume 6 November (issue 4), pp. 351 – 362.

Siegwart, R. (2001). Indirect Manipulation of a Sphere on a Flat Disk Using Force Information. *International Journal of Advanced Robotic Systems,* Vol.6, No.4, (December 2009), pp. 12-16, ISSN 1729-8806

Thomas, L., Ratclife, M., Wodobury, J., and Jarman, E., learning styles and performance in introductory programming sequence. Proceedings on the 33rd SIGCS technical Symposium on Computer Science Education, pp 33-77.

Van der Linden, S. (June 2010). Integrating Wind Turbine Generators (WTG's) with Energy Storage, In: *Wind Power,* 17.06.2010, Available from

Weisheit, F., Using practical toys, modified for technical learning: A class aimed at increasing children's interest level in computer science. ACM Crossroads, Volume 10, Issue 6, 2004.

Yanco, H., 2001. 91.450: Robotics I. http://www.cs.uml.edu/`holly/91.450
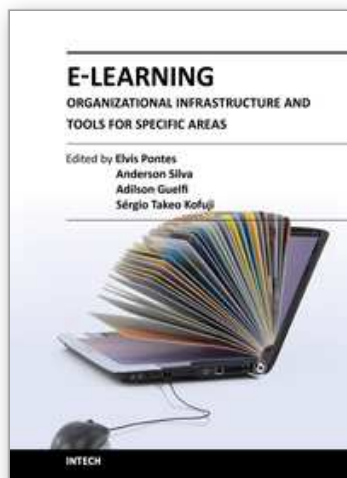
E-Learning-Organizational Infrastructure and Tools for Specific Areas

Edited by Prof. Adilson Guelfi

Technology development, mainly for telecommunications and computer systems, was a key factor for the interactivity and, thus, for the expansion of e-learning. This book is divided into two parts, presenting some proposals to deal with e-learning challenges, opening up a way of learning about and discussing new methodologies to increase the interaction level of classes and implementing technical tools for helping students to make better use of e-learning resources. In the first part, the reader may find chapters mentioning the required infrastructure for e-learning models and processes, organizational practices, suggestions, implementation of methods for assessing results, and case studies focused on pedagogical aspects that can be applied generically in different environments. The second part is related to tools that can be adopted by users such as graphical tools for engineering, mobile phone networks, and techniques to build robots, among others. Moreover, part two includes some chapters dedicated specifically to e-learning areas like engineering and architecture.

INTECH

open science | open minds