

Introduction to Autonomous Robotics with Lego Mindstorms

H. Levent Akin, Çetin Meriçli, Tekin Meriçli, and Ertan Doğrultan

*Department of Computer Engineering, Boğaziçi University,
Bebek, Istanbul, 34342, Turkey*

E-mail: {akin,cetin.merichli,tekin.merichli,ertan.dogrultan}@boun.edu.tr

This paper describes an introductory robotics course where the Lego Mindstorms NXT kits are used as the robot platform. The aims, scope, and the contents of the course are presented and the design of the laboratory sessions is explained in detail. Finally, the term project of the course, for which the students were expected to design an autonomous robot to navigate in a maze to reach its goal and climb up a pole to complete the task, is elaborated from the students' point of view.

Keywords: Robotics Education, Climbing Robots, Lego NXT

1. Introduction

A good comprehension of the autonomous robotics field heavily relies on solid background knowledge in a great variety of fields like mathematics, probability, physics, mechanics, electronics, and computer science. A good undergraduate level introductory robotics course should cover as many fundamental topics as possible, yet should be easy to understand. Additionally each covered topic should be backed up by an appropriate hands-on laboratory session for providing a concrete experience on the application of the covered theoretical material.

Lego Mindstorms NXT kits provide an excellent hardware platform for rapid prototyping of sophisticated robotic hardware and makes it very easy to implement examples of the *Sense – Plan – Act* paradigm with its simple sensor and motor interface. Teaching robotics to computer science undergraduate students is a challenging problem since they are used to develop software for ready to use hardware and hardly know about the uncertainty associated with every physical part of a typical robotic system. Although a carefully crafted software system highly utilizes the physical capabilities of the platform, a good robotic system should balance the utilization of both

hardware and software capabilities of the available robotic platform if we have the opportunity to modify the hardware as well.

This paper describes an introductory robotics course offered in the Department of Computer Engineering at Boğaziçi University, Turkey. The aims, scope, and the contents of the course are presented and the design of the laboratory sessions is explained in detail. An evaluation of the course from the student point of view is also presented.

2. Course Description

Based on our previous teaching experiences in graduate level introductory robotics courses, we started with the following main groups of topics that we consider important and would like to cover in the course:

- Uncertainty modeling
- Basics of navigation
- Controller architectures
- Advanced topics

Expanding the mentioned groups, we decided to cover following specific topics listed below:

- Uncertainty in sensing and action, and methods for dealing with it
- Self localization
- Path planning and obstacle avoidance
- Behavior based and reactive controller architectures
- Control theory
- Multi-robot systems, learning, and mapping

Covered topics are supported with an optional textbook¹ and a set of reference books on robotics,²⁻⁴ AI,⁵ and Lego Mindstorms.⁶⁻⁸

The course is a lab-driven course with two hours of lab sessions per each one hour of lecture. The lab sessions associated with the covered topics can be listed as follows:

- **Introduction:** The main purpose of this lab is to become familiar with the Lego parts and the development environment.
- **Building First Robot:** Building a simple robot and implementing a set of simple controllers using the Lego Mindstorms software.
- **Moving to Java:** Converting firmwares of NXT bricks for Java compatibility and reimplementing previously implemented controllers using Java.

- **Sensors and Actuators:** The students are expected to calibrate the light, ultrasound, sound, and touch sensors, and also the motors by fitting a model on the relation between the measured property and the sensory reading. A similar procedure is repeated for establishing the relation between the motor speeds applied and the linear and rotational motion observed.
- **Locomotion:** Application of UMBmark odometry calibration procedure⁹ on differential drive robots and reporting of the observed errors as well as the level of improvement before and after calibration.
- **Control:** The students are asked to build a two wheeled self-balancing robot using a light sensor pointing downwards as its feedback mechanism. They then try to find appropriate coefficients for a complete PID control code for keeping the robot balanced.
- **Reactive Architectures:** The aim of this lab is to show the students that complex behaviors can be obtained by using very primitive sensory-motor couplings. The students are asked to implement the first four vehicles from Valentino Braitenberg's book¹⁰ and also to implement a vehicle of their own design.
- **Behavior Based Architectures:** The students build a pre-designed robot with an ultrasonic sensor on a turning turret and then they are asked to develop a behavior scheme using subsumption architecture¹¹ for seeking a light source and heading the robot towards it.
- **Localization:** This lab mainly concentrates on developing appropriate observation and motion models for a complete Monte-Carlo Localization system.¹² The students use a wireless monitoring tool for observing the performance of the developed self localization system.
- **Path Planning:** The students implement the Bug1 algorithm¹³ for a robot that tries to avoid obstacles and run towards a light source.

2.1. Term Project

The term project includes the simplified versions of the main problems in the field of robotics, some of which are path planning, map making, perceiving the environment, and odometry. In addition, the mechanical design for a pole climbing robot is a part of the project to challenge the physical capabilities of the robot. Mainly, the approach to those problems and carrying

the theoretical knowledge of robotics to a real world set up is aimed.

The students are asked to design a robot for rescuing the *princess* from a *dungeon*. The robot needs to climb a pole after finding the princess in order to get out of the dungeon. The dungeon world is implemented as a rectangular area consisting of square cells and the objects of interest are color-coded. The area is surrounded by walls and each square can contain at most one type of object. The valid object set consists of:

- An obstacle (no specific color information)
- The pole (in blue)
- The princess (a yellow cylinder)

The students can use extra sensors but they are limited to using three motors and one Lego brick. The initial position of the robot is determined randomly and is same for all student groups. Figure 1 shows the picture of two models designed for the term project.

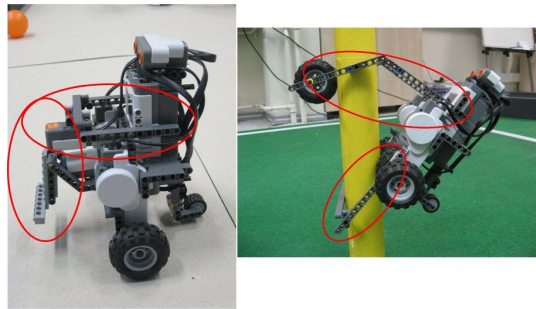


Fig. 1. Two models for the specified tasks.

3. Student Evaluation

This project was very beneficial for giving us the chance of designing and implementing algorithms for some of the main problems of robotics, such as perceiving the environment, map making, and calibrating and making use of odometry, and path planning. The limitation on the number of motors challenged us to come up with clever designs for pole climbing.

The boundaries of the map are known by the robot. However, the starting location or the locations of the obstacles on the map are not given, so the robot has to explore. Mainly the ultrasonic sensor and odometry information are used to navigate in the environment, hence odometry calibration

was necessary before starting the experiments on the map. The robot is supposed to know what the perceived cell contains. This is achieved by using the light sensors since the colors of the goal and the pole is specified in the description. The agent should know the cells it has visited before in order to prevent deadlocks and to be able to find the goal. Since the world is defined as discrete cells and the odometry readings supply reasonably deterministic displacement feedback, this requirement can be considered as less complex than a regular map making problem.

The robot is supposed to know the content of the cells in its perception. The light sensors are utilized for that purpose, since the colors of the goal and the pole is specified in the description and although the light sensors of Lego are only sensitive to intensity, intensity characteristics of different colors can be modeled and distinguished statistically.

The Lego brick that the robot carries is too heavy to lift; therefore, the robot can slide while climbing. In order to prevent that, feedback from the motors are considered and the force to be applied is determined accordingly using a control loop.

The software on the robot is mainly responsible for decision making part. A variety of possible misperceptions are considered and reasoning is done by taking those situations into account. The experience gained in the laboratory sessions of the course had an important role to implement an efficient framework since it is important to be familiar with the capabilities of both software and hardware parts of the robot, and to know what it is capable of doing as well as what it is not.

In robotics, one of the main problems of software implementation is debugging. What the robot perceives and what decisions it makes are usually not visible to the programmer. Therefore, a special software utilizing the wireless Bluetooth connection capability of NXT brick for debugging is implemented for this project to make it easier to progress.

The testing is done for different scenarios and the robot succeeds if the sensors are not deceived too much. The percentage of the misperceptions has a great effect on the performance of the robot. However, by using many different scenarios and making various observations, it is made sure that the robot does not just rely on its sensors or the odometry information. It also considers the probability of making a mistake in the perception level and tries to be sure about its perception by using different types of sensors with different inference strategies.

Since knowing the capacity and limitations of the hardware is important in directing the design process, the experience gained in the laboratory

sessions of the course had significant effects on the final design.

4. Conclusions

This paper presents an introductory robotics course from the instructor's point of view with its design goals and expectations from the course, and from the students' point of view with their evaluation of the term project which can be considered as a summary of the course. Students feedbacks indicate that although there is still room for improvement, the course has achieved the design goals which were to introduce the field of mobile autonomous robotics to the undergrad students, and giving them a challenging yet amusing team work experience.

Encouraged by the success of this course, we are planning to extend both laboratory sessions and the term project to cover more advanced topics like multi-robot systems or vision based navigation and mapping as a future work.

References

1. M. Mataric, *The Robotics Primer* (MIT Press, 2007).
2. G. Dudek and M. Jenkin, *Computational Principles of Mobile Robotics* (Cambridge University Press, 2000).
3. F. G. Martin, *Robotic Explorations: A Hands-on Introduction to Engineering* (Prentice Hall, 2001).
4. R. R. Murphy, *An Introduction to AI Robotics* (MIT Press, 2000).
5. S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach* (Prentice Hall, 2003).
6. M. Ferrari, *Building Robots with Lego Mindstorms* (Syngress Pub., 2002).
7. B. Bagnall, *Maximum Lego NXT: Building Robots with Java Brains* (Variant Press, 2007).
8. M. Gasperi, P. E. Hurbain and I. L. Hurbain, *Extreme NXT: Extending the LEGO MINDSTORMS NXT to the Next Level* (Apress, 2007).
9. J. Borenstein, L. Feng and C. J. Borenstein, Measurement and correction of systematic odometry errors in mobile robots, in *IEEE Transactions on Robotics and Automation*, 1996 pp. 869–880.
10. V. Braitenberg, *Vehicles: Experiments in Synthetic Psychology* (The MIT Press, February 1986).
11. R. Brooks, A robust layered control system for a mobile robot, in *Robotics and Automation, IEEE Journal of [legacy, pre - 1988]*, (1) 1986 pp. 14–23.
12. S. Thrun, W. Burgard and D. Fox, *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)* (The MIT Press, September 2005).
13. V. J. Lumelsky and A. A. Stepanov, Path-planning strategies for a point mobile automaton moving amidst unknown obstacles of arbitrary shape (Springer-Verlag New York, Inc., New York, NY, USA, 1990) pp. 363–390.