Teaching AI using LEGO Mindstorms

Simon Parsons
Department of Computer and Information Science
Brooklyn College
City University of New York
Brooklyn, NY 11210
parsons@sci.brooklyn.cuny.edu

Elizabeth Sklar
Department of Computer Science
Columbia University
1214 Amsterdam Avenue
New York, NY 11025
sklar@cs.columbia.edu

Abstract

We have taught a number of artificial intelligence classes that include project work centred around the use of LEGO Mindstorms robots. These courses have been offered at three institutions that have quite different student populations, and all have been greeted enthusiastically by the students. This paper aims to share some of our experience, give pointers to material we are developing (and hope others will use), and describe some of our efforts to quantify the educational benefits of using robots in the classroom.

Introduction

Artificial intelligence can be a hard subject to teach at the undergraduate level. The great breadth of the subject means that many introductory courses skate over the surface of many of the areas, sacrificing depth for a broad overview of the many achievements in the field. Such an approach can be stimulating for students who engage with the subject, but for many the glittering range of AI gems can seem to be a disjoint set of unconnected topics. An alternative approach is to go into some of the foundational methods in depth, but this often leaves students feeling that "AI is just search" or "all we learnt was more discrete maths".

As a result, many faculty seem to prefer organising their courses around a common theme (as indeed we do), and a popular theme is that of agents. Partly this has to do with the current popularity of that field, and partly it has to do with the availability of a very good textbook (Russell & Norvig 2003) which takes exactly this stance. However, if one is going to go this route, why not make a slight detour and make the central theme that of *the* prototypical agents—embodied agents. That is, why not make the course about robots?

There are several reasons for doing this that we think are compelling.

- It is very easy to see why robots are agents according to most of the criteria that have been proposed (Franklin & Graesser 1997), including those used by Russell and Norvig (2003). They therefore make good concrete examples.
- Students are familiar with robots. It is therefore easy to come up with examples of robot behaviour that communicate sophisticated ideas in an accessible way.
- Students find robots intriguing. Using robots is therefore a way of capturing and holding interest in the subject.

Against these positive aspects are ranged the fact that historically robotics equipment has been sufficiently expensive that it was not possible to offer all undergraduate students hands-on experience of programming robots (thus undermining the value of organising the course around them), and the fact that, in general, using robot platforms involves lots of background work on construction and calibration that has little direct connection to the material the students are learning in an AI curriculum.

However, the advent of platforms such as the LEGO Mindstorms[1] has provided a way around these problems. Not only are the robots sufficiently cheap that even a limited budget can purchase enough for a class, but both the construction of robots and the calibration of sensors are within the capabilities of 11 year-old children.

This, then, has been our motivation in developing AI courses that focus around robotics on the LEGO Mindstorms platform and its RCX programmable brick. The aim of the paper is to briefly report our experience, to describe the current state of the course we have developed and planned future development, and to present our work evaluating the impact of using robotics in this way.

Our experience

History

Our effort began in Spring 2001, when Sklar taught a course "Introduction to Robotics" at Boston College[2]. The Computer Science Department at Boston College

[1] http://www.legomindstorms.com

[2] http://www.cs.columbia.edu/~sklar/teaching/spring2001/mc375/default.html

is in the School of Management, and there is no engineering school in the university, so part of the aim of the course was to introduce students to aspects of computer science that they would otherwise have had no contact with. The course included one lecture and one laboratory session a week, and the work centered around two contests in which students robots participated.

As a result of Sklar's involvement in RoboCupJunior[3]—the division of RoboCup[4] aimed at children from age 8 and up—the contests were based on RoboCupJunior challenges. One contest was a timed run through a maze (a black line on a white background), and the other was a game of soccer with a light-emitting (and thus easy to track) ball.

A year later Parsons brought robotics into the introductory AI class at Columbia University[5]. That course was built around Nilsson's "Artificial Intelligence: A new synthesis", which is not only agent-centric, but also covers reactive control architectures in detail early on. As a result, it is easy to incorporate material on approaches like the subsumption architecture (Brooks 1991) that provide a route to using AI methods in the robotics work even on a limited platform like the Mindstorms. While this course was constrained by timetabling to be mainly lecture-based[6] it ran broadly the same two contests as at Boston College (with an additional, extra credit project in which students choreographed their robots to dance to music).

The same basic pattern has been used by Parsons at the City University of New York (CUNY) every semester since[7], with the main changes being in the increased provision of resources to the students to support the robotics work, a steady refinement of the contests, and the modification of the syllabus. The syllabus is now largely based around Russell and Norvig's "Artificial Intelligence: A modern approach" (Russell & Norvig 2003), but includes the material on reactive control mentioned above.

Course structure

The current version of the course is best described by the material on the course web-page[8]. This gives the detailed syllabus for various offerings of the course, all of which use a subset of topics outlined in Figure 1 (We have developed curricular modules for each of these topics and vary which modules we include in each offering.) The web-pages also give the course schedule, homework, additional readings, lecture notes, and the detail of the robotics projects. Here we describe the organisation of the projects, concentrating on aspects that are less obvious from the web site.

- Introduction to AI
- What is an agent?
- Reactive control
- Introduction to robotics
- Perceptrons
- Machine vision
- Heuristic search
- Adversarial search
- Knowledge representation
- Rule-based expert systems
- Propositional logic
- Predicate logic
- Commonsense reasoning
- Means-ends planning
- Partial-order planning
- Reinforcement learning

Figure 1: Topics which have been covered on the introductory AI courses

The projects are group efforts and account for 25% of the term grade. We find that the optimum group size is two students (more and it is easy for one to become a passenger, or to get shut out), though limitations on the number of robots has often forced us to have three or more in a group. Each group is supplied with a kit that contains:

- Around 200 LEGO parts from which to build their robot (a subset of the 700 or so pieces that come as part of the standard LEGO Mindstorms kit); and
- A set of robot designs and sample code to run on those robots.

There is an initial lab session in which the students build and program a very simple robot in order to become familiar with the basic concepts, and then the lab is made available outside of class-time for the students to get together and practice running their robot under the same conditions in which it will be evaluated in the contests.

The robot designs and sample code are taken from Baum's "Definitive Guide to LEGO Mindstorms" (Baum 2000a) (a book that is thoroughly recommended to anyone thinking of using the LEGO Mindstorms), and are chosen to illustrate the various issues that the students will have to deal with when participating in the con-

[3] http://www.robocupjunior.org

[4] http://www.robocup.org

[5] http://www.cs.columbia.edu/~sp/4701-2.html

[6] Though it should be noted that this does not necessarily rule out regular practical work—the LEGO kits are handy enough that they can easily be used in a classroom, and with cheap laptops liberate robotics from the laboratory.

[7] Courses following the basic pattern described here were run for undergraduates in Fall 2002, Spring 2003, and Fall 2003. Similar courses have also been run for Master's students in Spring 2003, with projects based on robot simulations, and Fall 2003 where the LEGO robots were used, but the course featured more advanced material.

[8] http://www.sci.brooklyn.cuny.edu/~parsons/courses/

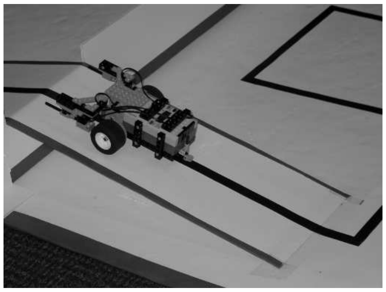


Figure 2: The first contest.

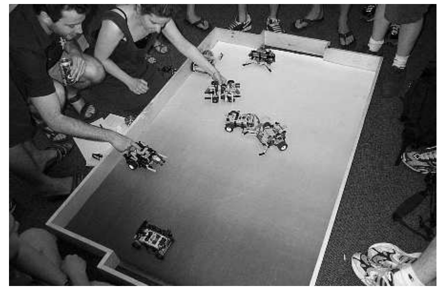


Figure 3: The second contest.

tests. The set of parts is sufficient to build all the robot designs, though none of these designs will suffice alone for the challenges. The idea is to try and balance giving the students some help with the mechanical design problem, since this is the aspect of the project least connected to the subject matter of the AI course, without solving the problem for them. Similarly the code covers many of the necessary aspects without coming close to solving the problems. We don't give the students the entire LEGO Mindstorms kit because:

- we know students will have to carry the pieces to and from campus (because there is nowhere the robots can be stored on campus), and reducing the kit to something that will fit in a medium-sized plastic food-storage box reduces the burden, especially for students who commute over an hour on the subway as many of the students at CUNY do; and
- we know pieces will get lost, and keeping some pieces in reserve (especially some of the smaller parts) makes it possible to absorb these losses without having to purchase new kits.

Our experience has been that this reduced set of parts does not greatly restrict the range of designs that our students manage to create.

We are in the process of fully documenting sample robot designs, programs, and a selection of different sets of parts appropriate for different projects. These include programs to illustrate specific agent architectures—such as the subsumption (Brooks 1991) and the belief/desire/intention (Bratman, Israel, & Pollack 1988) architectures, both of which have been implemented using the materials from the course—and robot designs that are more flexible than those given in (Baum 2000a). As we complete this work, it will be made freely available from `http://agents.cs.columbia.edu/er/`.

The contests in which the students take part are designed to test the full range of what is possible with the Mindstorms, at least using the sensors it is supplied with (albeit with the addition of an extra light sensor for the second contest), and the use of the Not Quite C (NQC) language[9].

The first contest we have been using is intended to be introductory, and therefore relatively straightforward. The main task is to follow a black line on a white background, around a number of curves (both left-hand and right-hand). Sometimes the contest involves climbing and descending a gradient (see Figure 2) in order to make the task harder. The fastest robot to follow the line from one end to another wins, and to add a little complexity there are three further tasks. The first of these additional tasks is that the robot has to detect an obstacle using touch sensors—once it indicates it has detected this (by backing up), the obstacle is removed. The second additional task is to recognise colored areas in the middle of the line and to indicate they have been found by playing a tune. Since these have to be detected using the same sensors as detect the line, and give a reading between that of the black line and the white background, they introduce a trade-off between speed and accuracy of detection—failure to detect the areas results in a time penalty as does detection of areas that aren't there. The final additional task is to detect colored areas at the end of the course (which are of another color completely) and stop when they are reached.

Part of the reason the first contest is simple is that the world the robot has to operate in is static. In contrast the second contest involves a dynamic world—groups have to build robots to play a simple form of one-on-one robotic soccer. The pitch for this game is shown in Figure 3. It is carpeted with a grayscale, to give some directional information that can be read with a light sensor. The ball used in the contest emits infra-red light, and so is relatively easy to detect using a second light sensor. The contest takes the form of a soccer tournament. The project groups are split into two leagues and each league plays a round-robin of 3 minute games (which seem to be long enough to get a result fairly often, but short enough that the entire competition can be fitted into a class) followed by a final between the winners of the round-robins.

The combination of kits, challenges and language seem appropriate for the students who take these courses. It is possible, though, to go beyond what we are doing in terms of sophistication while still using the Mindstorms platform. It is possible to extend the kinds of tasks students can address by purchasing differ-

[9] `http://www.baumfamily.org/nqc`.

ent sensors. For example, LEGO sells a rotation sensor that can be used to supply odometry information, and it is possible to purchase inexpensive infra-red range finders[10] and even a compass[11]. It is also possible to extend the range of possible tasks by using a more sophisticated programming language. BrickOS[12] (Baum 2000b) provides a more complete version of C than NQC, and one can program the Mindstorms in Java (Laverde, Ferrari, & Stuber 2002) and Lisp (Klassner 2002; Klassner & Anderson 2003).

Course evaluation

While we believe that bringing robotics into AI courses is beneficial to students on the courses, we are keen to produce research evidence that this is true. As a result, we have been regularly assessing what the students think of the courses, in particular how they feel the robotics element helps them, by administering surveys to every cohort of students that has taken one of the robotics-enhanced courses. Here we give results from the classes that have been offered at CUNY and Columbia over the last two years. Figure 4 gives the parts of the course that that the students identified as being helpful in *learning* the material, and Figure 5 gives the parts of the course that the students felt were in *demonstrating their knowledge* of the material.

The surveys asked each student to identify those aspects of the course that they personally found useful in both these ways, the figures show the percentage of students who identified each element. Separate results are given for each offering of the course, which has been run 6 times, twice for graduate (G) students and four times for undergraduate (U) students.

These results show fairly consistently that the project was felt to be less helpful in learning than more traditional elements like lectures, lecture notes and homework but is more helpful than additional readings or the textbook. In terms of demonstrating knowledge, the students felt that the project was more helpful than the final and some felt it to be as helpful as the midterm.

The one set of results that is inconsistent with these findings are those for the Spring 2003 graduate course. For that offering the projects were performed using simulated robots (because of a lack of the LEGO kits). It is tempting to take the much lower figures as reflecting the fact that programming simulated robots is less satisfying than programming the real thing, but it might equally well reflect the fact that the simulator[13] presents a more challenging programming environment than the RCX[14].

[10] `http://www.mindsensors.com`

[11] `http://www.wiltronics.com.au`

[12] `http://brickos.sourceforge.net/`

[13] The RoboCup Soccer Simulator, `http://sserver.sourceforge.net/`.

[14] Verbal feedback from the students suggested that those who mastered the simulator were very happy with the project.

Overall we think these results, and the similar results from other offerings of the course, are encouraging enough to warrant our continued use of robotics. Maybe more encouraging than the survey responses, though, are the free-form comments we have received, which include:

- "When working with the robot, I learnt that nothing is perfect in the real world. A lot of times the outcome is very unexpected."
- "It reminded me of why I want to stay away from hardware as much as possible."
- "It helped immensely! It helped me understand some of the concepts covered in the lecture."
- "Project helped [me] to realise how important [it] is to divide complicated tasks into smaller ones in order to solve it."
- "It is nice to put theory to practice."

In addition, in the most recent undergraduate offering of the course, 14 of the 15 respondents used these free-form comments to indicate that the use of robotics had been beneficial, and that the project work had been fun.

For more information on the evaluation aspect of our work, see (Sklar, Parsons, & Stone 2003; Sklar, Eguchi, & Johnson 2002).

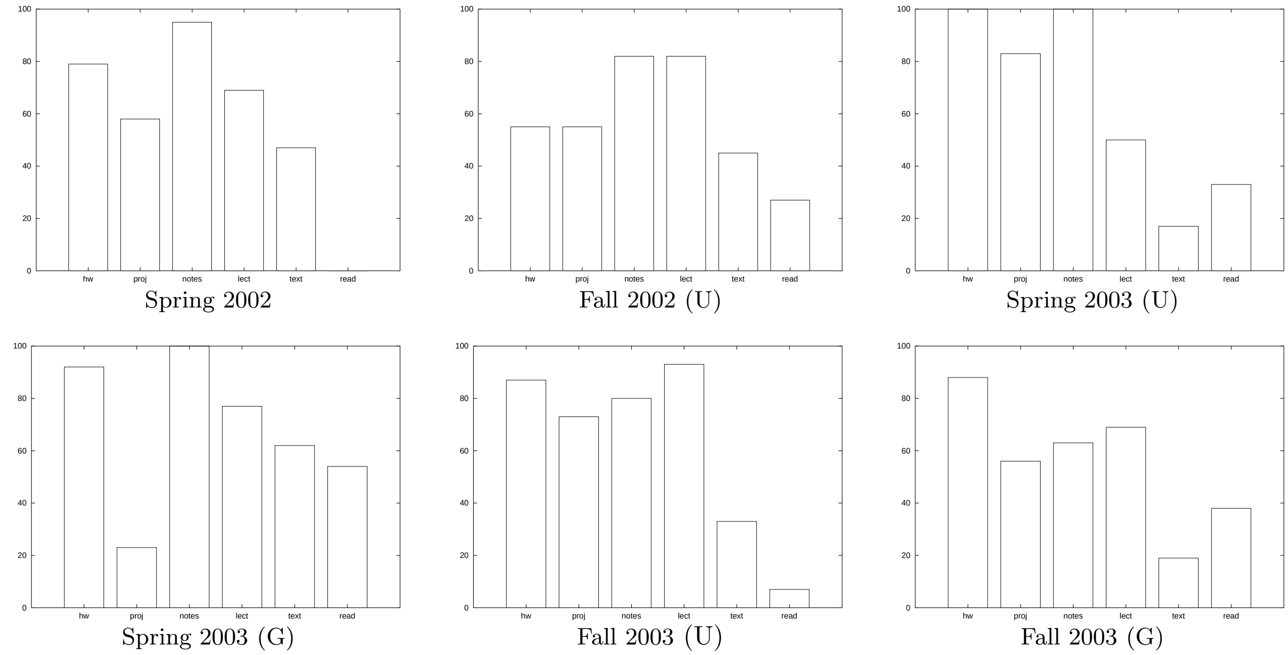
Related work

We are, of course, not the first to use robot kits in an undergraduate classroom as a hands-on learning environment. In 1989, Martin created the MIT Robot Design project course (6.270), following from Flowers' "Introduction to Design" (2.70) course that was offered in the Mechanical Engineering department (Martin 1989). The work on this course culminated in a textbook (Martin 2000). Students learn about the basics of building robots from kits and the course ends with a contest. Yanco (2001) has adopted this course using the Botball[15] game as the tournament at the end of the term. Mataric has developed an award winning course called "Introduction to Robotics" (Mataric 1998) which takes a hands-on approach to the introduction of the basic concepts in the field of robotics. Students use both the Handyboard[16] microcontroller and the LEGO Mindstorms system. The syllabus covers in detail the basic components of robotics from a technical standpoint and the course ends with a contest where robots play a ball game in a hexagonal field. Another introductory course on robotics that uses LEGO Mindstorms is the "Building Intelligent Robots" course, taught by Dean at Brown in 2001 (Dean 2001).

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" (Littman 1999) teaches about a variety of methods for programming, as its title says, under

[15] `http://www.botball.org`

[16] `http://www.handyboard.com`



key:

hw	= homework	proj	= robot project	notes	= lecture notes
lect	= lectures	text	= textbook	read	= additional readings

Figure 4: Elements of the course which students identified as helpful to them in learning the material

uncertainty, including Markov Decision Processes and POMDP's, and a variety of machine learning techniques like reinforcement learning and genetic algorithms. Students in this course used LEGO robots to demonstrate their knowledge of the methodologies studied. The course ended in a project, where some of the students developed their own applications for their robots, from line-following tasks to making breakfast. One of these, described in (Baum 2000b), carried out on-line reinforcement learning to complete a task analogous to pole-balancing—an indication of what is possible at the upper limit of the Mindstorms' capabilities.

Finally, we must mention the work of Klassner (2001) who, unbeknowst to us until recently, has been teaching introductory AI using LEGO Mindstorms (and indeed (Russell & Norvig 2003)) since 2001. Klassner's students make extensive use of the robots, and are supported by tools developed by Klassner and his colleagues (Klassner 2002; Klassner & Anderson 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.

Conclusions

For the past two years, we have been using the LEGO Mindstorms robot platform in teaching artificial intelligence courses. This paper describes some aspects of this work. We briefly described the history of our use of the Mindstorms, and then elaborated on the current offering of the course. To try to quantify the value of using the robot kits we are surveying all the students who take it, and we gave some of the results of those ongoing surveys.

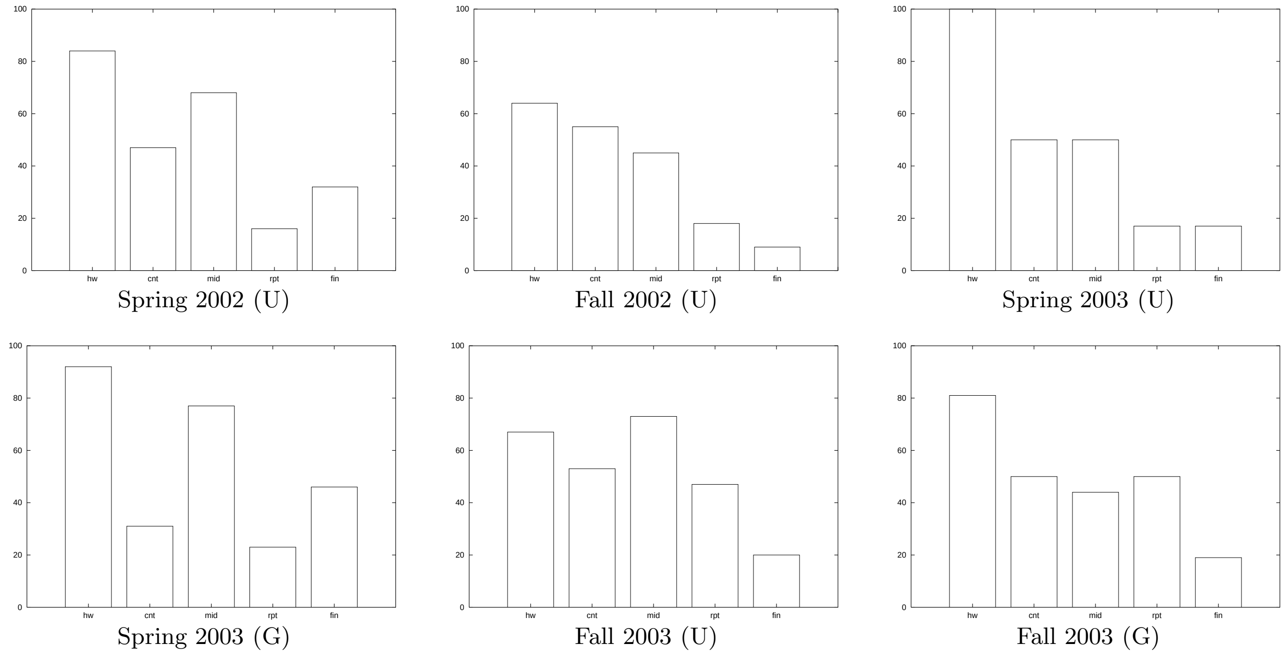
While we don't think that we have fully tapped the resources of the LEGO hardware as yet, we are also exploring the use of more sophisticated platforms. We participate in the RoboCup Legged League using Sony AIBO robots. While these are more complex to program than the LEGO Mindstorms, and are nearly an order of magnitude more expensive, we believe it is still possible to use them (albeit for a small class) provided one gives the students code that provides basic image processing and motion—allowing the students to concentrate on applying their AI knowledge to the writing of control programs. Of course we are in a privileged position, having put together code to do these things, but both our code and that of many other RoboCup participants is freely available for download.

References

Baum, D. 2000a. *Dave Baum's Definitive Guide to LEGO Mindstorms*. APress.

Baum, D., ed. 2000b. *Extreme Mindstorms: An advanced guice to Lego Mindstorms*. APress.

Bratman, M. E.; Israel, D. J.; and Pollack, M. E. 1988.



key:

hw	= homework	cnt	= robot contest	mid	= midterm exam
rpt	= project report	fin	= final exam		

Figure 5: Elements of the course which students identified as helpful to them in demonstrating their knowledge of the material

Plans and resource bounded practical reasoning. *Computational Intelligence* 4:349–355.

Brooks, R. A. 1991. Intelligence without representation. *Artificial Intelligence* 47:139–59.

Dean, T. 2001. CS148: Building Intelligent Robots. `http://www.cs.brown.edu/~tld/courses/cs148/01/syllabus.htm`.

Franklin, S., and Graesser, A. 1997. Is it an agent, or just a program? In J. P. Müller, M. W., and Jennings, N. R., eds., *Intelligent Agents III*. Berlin: Springer Verlag. 21–36.

Klassner, F., and Anderson, S. 2003. LEGO MindStorms: Not just for K-12 anymore. *IEEE Robotics and Automation* (in press).

Klassner, F. 2001. Introduction to Artificial Intelligence. `http://www.csc.vill.edu/~klassner/csc4500/`.

Klassner, F. 2002. A case study of LEGO Mindstorms suitability for artificial intelligence and robotics courses at the college level. In *Proceedings of the 33rd SIGCSE Technical Symposium on Computer Science Education*, 8–12.

Laverde, D.; Ferrari, G.; and Stuber, J., eds. 2002. *Programming Lego Mindstorms with Java*. Syngress.

Littman, M. 1999. CPS196: Programming Under Uncertainty. `http://www.cs.duke.edu/~mlittman/courses/cps196/`.

Martin, F. 1989. 6.270: The MIT LEGO Robot Design Project Competition. `http://fredm.www.media.mit.edu/people/fredm/projects/6270/`.

Martin, F. 2000. *Robotic Explorations: A Hands-On Introduction to Engineering*. Prentice Hall.

Mataric, M. 1998. CS 445 Introduction to Robotics, a LEGO-kit-based hands-on lab course. `http://www-scf.usc.edu/~csci445/`.

Russell, S. J., and Norvig, P. 2003. *Artificial Intelligence: a Modern Approach*. Prentice Hall, 2nd edition.

Sklar, E.; Eguchi, A.; and Johnson, J. 2002. Robocupjunior: Learning with educational robotics. In *Proceedings of the 6th RoboCup Symposium*.

Sklar, E.; Parsons, S.; and Stone, P. 2003. Robocup in higher education: A preliminary report. In *Proceedings of the 7th RoboCup Symposium*.

Yanco, H. 2001. 91.450: Robotics I. `http://www.cs.uml.edu/~holly/91.450/`.