CS148
Building Intelligent Robots

Introductory Missive *Sep 2006*

CS148 is an introduction to fundamental topics in autonomous robot control. This course focuses on the development of "brains" for robots. That is, given a machine with sensing, actuation, and computation, how do we develop programs that allow the machine to function autonomously? We answer this question through a series of class discussions and group projects.

CS148 projects center on a "Roomba Pac-Man" task, where we program a robot vacuum cleaner to play a game of real-world Pac-Man. Various approaches to robot control (spanning reaction to deliberation) are covered using the Player/Stage/Gazebo (PSG) robot framework and Robot Roombas.

CS148 class meetings explore the technical, societal, and philosophical aspects of autonomous robots and human-robot interaction. Discussions amongst the class pose and address questions related to how robots can contribute to society, what technical functionality is needed for such contributions, and how will such technologies affect the human robot dynamic.

Meeting time: T,Th 2:30-3:50 (K Hour)

Prerequisites: CS4 or CS15/16 or CS17/18 or permission from the instructor

The current syllabus is available on the course web page, `www.cs.brown.edu/courses/cs148`. Labs and projects will be posted there as they are assigned. Check the web page and course newsgroup, `brown.cs.cs148`, often. Any important announcements posted on the newsgroup will also be posted on the MOTD page.

Course Staff (`cs148tas@cs.brown.edu`)

Instructor	Prof. Chad Jenkins (`cjenkins`)
Head TA (`cs148headtas@cs.brown.edu`)	Dan Hartmann (`dhartman`)
Teaching Assistants	Mark Moseley (`mmoseley`) Brendan Dickinson (`bcd`) Aggeliki Tsoli (`aggeliki`)

Prerequisites

CS 4, 15, 17 or permission

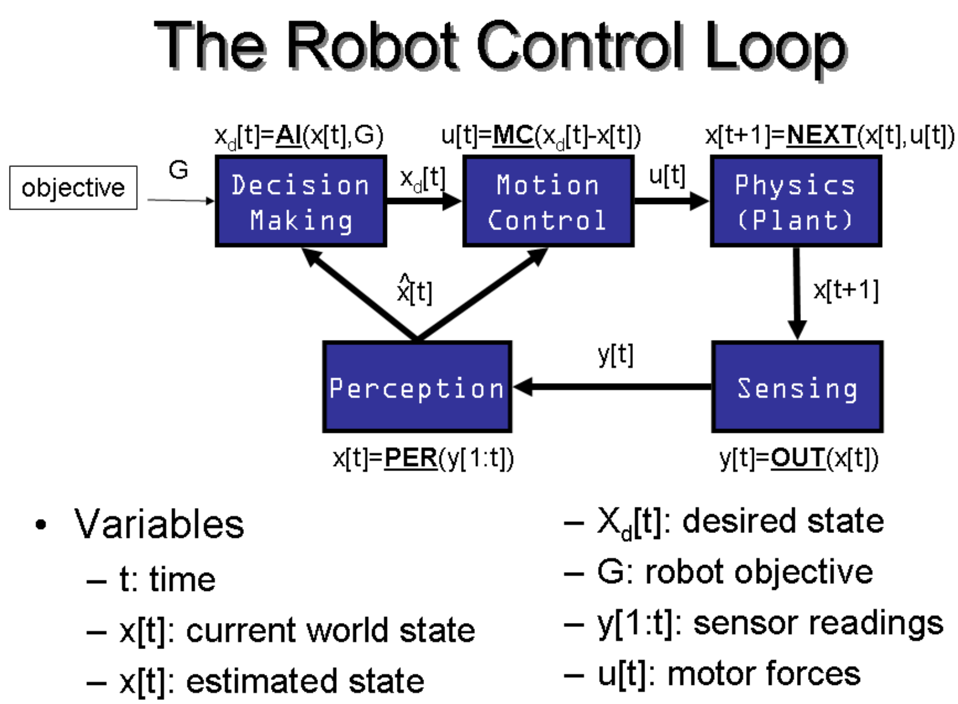
Disclaimer: the real world is unforgiving!

This course involves a significant amount of programming and decision making under uncertainty. Robotics is about understanding and functioning in the real world. However, the real world is highly dynamic, uncontrolled, and nondeterministic. These factors result in uncertainty that is unlike the structure and determinism of traditional computer science. Programming in the face of such uncertainty is a persistent challenge faced at all levels of robotics.

CS 15 or 17 should provide an adequate programming background for the projects in this course. In addition, motivated students who have taken CS 4 may also enroll in CS148. Interested students who have not taken any of these courses at Brown, but have some other strong programming experience should consult with course instructor.

Lectures

There will be no explicit lectures during class meeting periods. Instead, the class as a whole will develop basic robotics concepts through group discussion. Over the course of the semester, we will be assembling the components of the basic robotics control loop (shown below), comprised of perceiving the physical world, making decisions, and acting on those decisions.



Class discussions will be highly interactive, relying on student discussion to build fundamental concepts and societal relevance. Students are expected to prepare for class discussions through review a list of discussion points (distributed days before) and becoming familiar with the Robots section of the AI Topics Library.

Class Format

As stated above, CS148 will have both lecture and lab sessions, 3 assigned projects, and final project. Lab sessions will be devoted to a hands-on introduction to PSG and working with the Roomba platform leading up to the projects. The projects involve various approaches to programming autonomous Roomba Pac-Man. The final comprises developing a competitive Roomba Pac-Man and a written feasibility and technical analysis of a fictional robot. For 200-level credit, the final is an independent robotics project and a fictional robot analysis.

Roomba Pac-Man

The objective of the Roomba Pac-Man task is to vacuum as many "food pellets" as possible in a fixed period of time without contacting a "ghost". The player Roomba will be equipped with a camera, color blobfinder, IR ranging, and bump contact sensors. These sensors will be connected to a laptop running the Player robot interface.

Player/Stage/Gazebo (PSG)

PSG is a robot interface and simulation software suite, commonly used in robotics research. The core of this suite is the Player robot interface. Player is a network server for robot control. Running on your robot, Player provides a clean and simple interface to the robot's sensors and actuators over the IP network. Projects and labs will involve writing client programs that talk to Player for controlling the robot. Due to Player's client/server framework, these client programs can be written in a variety of languages, using a TCP socket for reading data from sensors, writing commands to actuators, and configuring devices on the fly.

Labs

Labs are designed to be completed during class time, though some may require longer. Students will initially work individually on labs and then in groups for working with the Roombas. When completed, labs must be demonstrated to an appropriate TA and checked off. There are no handins for lab.

The three labs are described by the following text and flowchart:

- Player/Stage/Gazebo Introduction: an introduction to the Player client/server robot interface and Gazebo robot simulation system. Teleoperation of a simulated robot and implementation of a collision avoidance robot client.
- PSG Roomba: interfacing PSG with a Roomba vaccum. Joystick control of the Roomba. Accessing the vacuum and sensory interfaces.
- PSG Vision: using the PSG camera and blobfinder devices. Performing color and camera calibration.

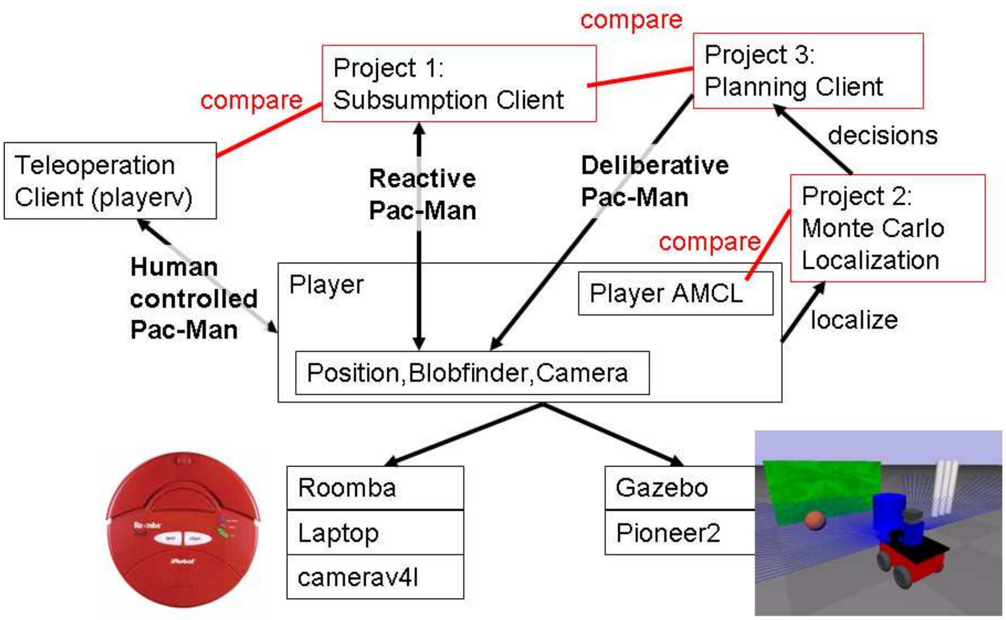
The first assignment (Lab 1) will be on Thursday, September 14th. The full lab and project schedule can be found in the syllabus on the course website.

Projects

Projects are graded assignments that consist of a implementation and a writeup. Projects are to be implemented by student teams with individually composed writeups. Implementations will be demonstrated during scheduled class periods for grading by the course staff. Following these demonstrations, writeups about the project will be due the following Monday. More information about the writeup format is described later in the missive.

The three projects are:

- Subsumption: develop a reactive robot client for playing Roomba Pac-Man
- Monte Carlo Localization (MCL): given a map, determine the location of a moving Roomba over time



- Path Planning: given a map and localization, develop a deliberative robot client for playing Roomba Pac-Man

Consult the course syllabus for more information about the timing and due dates for these projects.

Final Projects

The final project involves implementing a competitive Roomba Pac-Man and a written analysis of a fictional robot. The Roomba Pac-Man competition will be held during CS148 Demoday at the end of the semester. Student teams are expected to field their best Roomba Pac-Man client for this competition, refining and going beyond techniques used in previous projects.

The final paper for the class is a 10-page document analyzing a fictional robot. This paper will discuss the feasibility of the robot (in terms of factors such as its engineering, algorithms, social sophistication, cost) and necessary technologies for realizing this robot. Additionally, this paper must answer the following question: "What is the point of robotics?". Specifically, the answer to this question should include an argument about the need and most pertinent applications for robotics in society.

For 200-level credit, the final involves an independently designed and developed project along with the 10-page paper.

More information about these projects will be distributed as the semester progresses.

Note: It is crucial to attend and participate in class meetings. Many of the concepts that will be useful for the final project and paper will be discussed exclusively in class. You will be held responsible (in terms of critiques towards grading) for any material covered in class.

Demo Day

Final Demo Day will be be December 12th from 12pm-3pm. All students are required to present their projects for the duration of Demo Day.

200-Level Credit

CS148 can be taken for 200-level (graduate) credit. 200-level credit involves an independent final project that must be proposed earlier in the semester. Please consult the course instructor for more information.

Project Deliverables

Project Writeup Format

Each writeup is meant to be a scientific report about your project, the methods underlying your work, and its basis in the science of robotics. As such, the paper should follow the scientific method: observation, hypothesis, experiment, analysis, and conclusion. It should be objective and scientific in tone (avoid informal writing and use the first person sparingly). The paper should have a central thesis and everything in the report should contribute toward the validity or invalidity of the thesis. The paper should include the following sections:

- Abstract: A short 3 sentence description of your work.
- Introduction: The introduction should briefly state the problem and why it is relevant, state the thesis, and give a brief overview of how the paper will validate/invalidate the thesis.
- Approach: The approach describes the technical details of your work. This section includes the underlying design and methodology and relevant details of the technical components. Save comments about future extensions and the quality of the results for the discussion section. Code snippets are acceptable in this section, however, you should not copy your entire program into the report.
- Evaluation: In this section you should present the specific criteria that was used to gauge how the project validates/invalidates your thesis. Presenting the results from multiple runs of your system is encouraged.
- Discussion: This section includes analyses of challenges and problems encountered, the strengths and shortcomings of the project, and potential future extensions. This is also the section

where you should discuss why you made certain decisions regarding the methods and implementation of your project. For final projects, this section will also contain brief comparisons to existing work.

- Conclusion: A brief 1-2 paragraph summary of the central thesis, its validity/invalidity, and what was learned from the project.

A writeup should be crisp and succinct. It should be formatted in two columns, with 11 point font. The body of the writeup should not exceed 2 pages. Detailed or highly technical explanations may be added as appendices after the main body of the writeup.

An example writeup is available from the Head TA.

A final note: illustrations with good captions and labeling are extremely useful. The instructor is a sucker for pretty pictures.

Handins

In order to use the handin script correctly, electronic submissions need to follow a specific format and execute from a Linux machine in the CS department. The submission directory handed in should be a directory with your login as the name. Your login directory should contain your project writeup in PDF format, and another directory labeled "code". The code directory should contain all the source code for the project and a README detailing how to use the code. The project handouts will provide specific instructions as to how to run the handin script. Typically, the handin script is executed with a command in the format:

```
% cs148_handin <project_name>
```

Warning: Writeups submitted in formats other than PDF (e.g., plain text, Word) will not be accepted.

Grading

The grade distribution is as follows:

Labs	10%
Project: Subsumption	10%
Project: Monte Carlo Localization	20%
Project: Path Planning	20%
Final Project or Paper	30%
Class Participation	10%

Grading of all projects will be interactive. Therefore, it is important that you finish your projects on time. Project implementations will be graded during demonstration periods (see the syllabus for dates and times).

Late Policy

Labs should be completed during lab time, but if you are unable to do so, you may finish up during your own time and show it to a TA on hours. Each lab builds on the material from the last, so it is important to finish each lab in time for the next. Therefore, students will not receive credit for labs not completed by the next lab session.

Each student will be given 2 late days for use on projects or writeups over the course of the semester. If a team is a day late for a project demo, that assignment is considered late for every member of the team. Beyond those two late days, projects and writeups will lose credit at a rate of 10% a day. That means a maximum of 90% on an assignment that is one day late, and 80% for two days late, etc.

Any requests for extensions on projects (demos or write-ups) should be made to Chad. Extensions will be given at the instructor's discretion.

Late final projects will not be accepted.

Collaboration Policy

Students will do lab work in pairs and project work in groups. Each group will have access to a Roomba. Students in a group are expected to work as a team to implement projects. Of course, students on a team can and should collaborate fully. Help between teams may occur in a limited fashion. They may discuss the assignment, and even approaches to a particular problem. Students on different teams should not, under any circumstances copy code, or dictate code to each other.

Students will complete their write-ups individually. Data and images from tests and demos may be shared, but each team members's writeup must contain their own thoughts and ideas. Any usage of material, ideas, or concepts other than your own must be explicitly cited in your submission and will not count as central to your submission.

This policy is reiterated in the cs148 collaboration policy contract, which all students must sign and return prior to the first lab assignment.

Resources

Roomba Lab (CIT 472)

The Roomba Lab has a card reader on the door. All students in the class will be given access to the lab. Everybody will have access to the lab so its in everyone's best interest to keep the lab neat and clean. Final Note: The lab should never be left open and unattended; it contains expensive hardware.

Important: If you are leaving the lab, even just for a minute, DO NOT leave the door propped open. It is critical that the robots and machines for the course remain secured. Careless treatment of the course equipment will not be tolerated.

TA Hours

See the website for the individual TA's hours and locations.

Students will have access to the Roomba Lab at all hours. Students may use the nodes in the Roomba Lab.

The schedule is available the course web page. Please let us know if the current times don't work. We don't want to be holding hours when no one can take advantage of them, so feedback (sent to `cs148tas@cs.brown.edu`) would be appreciated.

Required Reading

There are no required textbooks for this course. All required readings will be taken from the AI Topics Library, namely from the Robots section.

Below are additional textbook resources that are available:

Recommended Books

- F. Martin: *Robotic Explorations: A Hands-On Introduction to Engineering*, Prentice-Hall, 2001.
- H. Choset, K. M. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L. E. Kavraki and S. Thrun, "Principles of Robot Motion: Theory, Algorithms, and Implementations" MIT Press, Boston, 2005

Additional Reading

- Craig: *Introduction to Robotics: Mechanics and Control (3rd Edition)*, Addison-Wesley, 1989.
- Bekey: *Autonomous Robots: From Biological Inspiration to Implementation and Control*, The MIT Press, 2005.
- Thrun, Burgard, Fox: *Probabilistic Robotics*, The MIT Press, 2005.
- Matarić: *The Robotics Primer*, pending publication, 2004.

Newsgroup

The newsgroup for this course is `brown.cs.cs148`. This is an appropriate place to post general and technical questions that other students are also likely to have. The course staff will also post announcements and comments to the newsgroup. Before emailing the tas with questions, please check the newsgroup to see that your question hasn't already been answered.