Massachusetts Institute of Technology

16.412J/6.834J Intelligent Embedded Systems
Final Project Description

Objectives

The purpose of the project is to develop a deep understanding of one or two methods for creating cognitive robots and intelligent embedded systems, and to innovate upon these methods, to lend novel insight into their behavior through analysis or to apply the method in an innovative manner.

More specifically, you should demonstrate the ability to:

- Clearly state and motivate an interesting, focused innovation to intelligent embedded systems. An innovation may be an important analytical question, a novel algorithmic extension or an innovative application.
- Extract and evaluate the relevant literature using the web and library resources.
- Provide a simple explanation for the algorithms used in your project, using pedagogical examples to highlight key features of the algorithm.
- If a design project, describe the design of the intelligent embedded systems you are creating and the rationale for the method applied in the context of the project. If this is an analysis project, then described the experimental method that you are pursuing.
- Implement and demonstrate an algorithm or application in support of your project goals.
- Evaluate the approach analytically and/or empirically.

Project Grading

- A – represents mastery: the ability to analyze and extend existing methods in a way that is novel and insightful; the ability to explain and motivate in a manner that is particularly intuitive.
- B – represents solid competence: the ability to articulately motivate, explain, implement and evaluate a focused set (i.e., 1 or 2) of intelligent embedded systems methods.
- C – represents partial competence of the above.

General

- The results of your project are to be captured in a written document of 10 to 20 pages and are to be presented orally, with slides.
- You may, if negotiated with me in advance, feel free to use your work in this class to extend or complement the work you do within some other project or your thesis.
- You are welcome to work together in teams, just like in the real world. This will allow you to tackle more ambitious projects. Projects of two tend to work the best, unless there are clear substantial roles for more.

Separate written reports are to be done, as everyone is to have that experience.

Important Deadlines

The project includes a project proposal a presentation of results and a final report. The deadline for each is given below.

The proposal will be up to 6 pages. The purpose is to for you to jump start the project before it is too late to get anything done, so take it seriously. Use the project proposal time to do your background work. The proposal should include:

- Cover page
- Introduction, including motivation for the project and assessment of its value to the technical community.

- Concise statement of the objectives of the project.
- Statement of previous work and methods that you build upon. This is where you do your background work.
- Outline of the proposed technical approach.
- Planning, including time schedule.

Also feel free to send me a one page email bouncing your ideas off me.

You will give team presentations of your project and what you've learned. These presentations will delivered during the last two weeks of the semester.

Your final project report will be due in class the final day of classes. This deadline MUST be met.

Types of Projects

Projects can be one of the following types:

- **A novel intelligent embedded system application.** You pick one or two methods from the papers you've read, perhaps find implementations available over the web, and you design and implement a novel intelligent embedded system. You demonstrate this capability in simulation, or if REALLY ambitious, on real hardware. You report the innovations of the design and any experimental results and findings.

  For example, you might want to prototype a simulated version of a soccer ball size micro-spacecraft that navigates around space station, looking for leaks in its hull. You might focus on the path planning problem, evaluating one to two path planning algorithms such as PLR, or you might try different approaches to reactive control, such as Jim Firby's Raps system.

  Developing a complete system is typically manageable if a simulator is used, but very risky if you plan on integrating with a real system. You can accomplish more by working in teams, and this is encouraged. Be careful not to be exposed to weak-link problems exacerbated by the part-time, conflicting pull nature of student labor. Alternatively you might propose a novel intelligent embedded system,

but then only implement a selected critical piece to assess feasibility. Caution is advised, along with especially clear and modest goals. On the other hand, if you succeed this can be extremely rewarding. Once you come up with an idea I might be able to suggest places where you can get code or additional papers. Many past projects have been sufficiently interesting, to warrant publication in the Innovative Applications of Artificial Intelligence Conference.

- **An extension.** You pick one of the papers that describes an implemented idea for controlling embedded systems and you reimplement it, reporting on the surprises that emerge as you understand the work at the level required to write code. You then create and report upon a novel extension to that approach, reporting on the improvements you make. Alternatively you propose a novel integration of two methods for creating intelligent embedded systems.

  For example, you could redo any number of the papers presented or to be presented in the course: a model-based diagnosis system, a temporal planning and execution capability, a system for solving POMDPs,…

  Remember that implementation has its risks, because everyone seems to overestimate what they can do by the end of the semester. On the other hand, if sufficient detail is given, and you are a good coder, this can be a lot of fun. So assess your programming skills and select accordingly.
- **A simulated area exam.** This concept is most familiar to Course 6 students. It is somewhat of a safety net project, although valuable when done well. You read a focused set of papers and do a scholarly critique. This should consist of one to two primary papers, or at most three, that describe a couple, closely related methods. Part of the process is to pick a set of dimensions along which to compare and contrast the papers. To demonstrate competence you should clearly explain the approaches and perform a careful analysis, often including an implementation of one or two of the methods and some empirical evaluation.

  To demonstrate mastery you should move beyond comparison, highlighting novel insights about the work and proposing interesting ways in which the research approaches can be synthesized or

extended. You conclude with a description of the degree to which you are moved, and why.

For example, you could contrast a Bayes net approach to state estimation with a model-bsed deductive approach; you could compare two recent papers on planning based on propositional encodings, plan graphs and model-checking; or you could contrast techniques for tracking hidden Markov models both from the POMDP and model-based reasoning literature. When you have a rough idea of what you might want to do, please bounce the idea off me. I can suggest directions to focus, or perhaps recommend particular papers.

Programming

Your project should demonstrate some level of implementation, either by implementing an intelligent embedded systems algorithm or by applying that algorithm to the creation of an interesting embedded system. The language you use is of your choice, and you may choose to use implementations of existing algorithms that authors may make available on the web. Languages typically used include C, C++, Lisp, Scheme, Prolog and Matlab.

Project Do's and Don'ts

It is easy to select an over ambitious project. Based on past experience, the following are some useful lessons:

- Maintain a tight focus. Go deep on one or two methods. Beyond two, it is easy to produce a project that offers no deep insights. Likewise, in an innovative extension or application, focus on one or at most two innovations.
- Carefully evaluate what is practical from an implementation standpoint, based on your programming skills. Most algorithms take two to four times longer to implement than most might expect.
- Create a plan of what features to descope, as a contingency for slips in your schedule. First brainstorm what you would like to accomplish in your project. Then create three plans:
    - An enhanced plan: What you would like to accomplish in your wild dreams.

    - Baseline plan: What you expect to accomplish, given a modicum of conservatism.
    - Minimal plan: A minimum project you need to accomplish to meet your learning goals. Don't be surprised if this is all you accomplish.

- Start early, start early, start early …..