Learning by Building Robots:
(Directed Constructionism)

Howie Choset
Robotics Institute
Carnegie Mellon University
Motivation

• Inspire

• Endow rigor

• Design

• Work in teams
Robotics Minor @ CMU

• Required Courses:
  – Introduction to Robotics (16-311)
  – Controls
  – Manipulation / Kinematics

• 2 Electives
  – Mobile Robot Programming
  – Mechatronics
  – Vision
  – Independent Study
Introduction to Robotics (16-311)

- Rube Goldberg Machines
- Braitenberg Vehicles
- Control
- Motion Planning
- Design
  - Mars Rover
  - Urban Search and Rescue
- Forward Kinematics
- Inverse Kinematics
- Non-holonomic constraints

Gateway robotics course for undergraduates
Creative Expression: USAR Lab
### Instructional Model

- Lecture based.
- Flow of knowledge is unidirectional.
- Caters well to curricular structure.
- Works well with large class size.
- Leaves us wondering if students really connect with the knowledge.

### Constructionism

- Students develop knowledge by engaging in the construction of physical artifacts.
- Individual, self-paced.
- When building things, a lot can go wrong - a strength.
- Experience is enhanced when projects are personally meaningful.
- Instructor becomes a mentor.
Directed Constructionism

- Artifact construction component that parallels curriculum.
- Design/construction tasks are open ended, allowing opportunity for creativity.
- Assignments attempt to be personally meaningful in that they validate theory on a real artifact.
Lab format and presentation

- Introduction
- Reading
- Pedagogical focus
- Challenge statement
- Evaluation criteria
- Construction tips
- Things to think about

Lab Culture
- Many late nights in the lab.
- Students teaching other students.
Example Lab Format

go to http://generalrobotics.org
Future Investigations

• Upgrading the NXT Systems
• What new tools and methods are required to allow robot building to enhance other subjects?
  – Math, physics, biological sciences.
• Other levels of education
  – Secondary, elementary
Conclusion

• **Inspire**
  – Robots are cool
  – Not teach CS for CS sake

• **Endow rigor**
  – Learn something real
  – Teach the realities and depth of science

• **Design**
  – Use tinkering to explore
  – Connect high and low level issues

• **Work in teams**