THE GREAT COURSES®

Topic
Science & Mathematics

Subtopic
Engineering

Robotics

Course Guidebook

Professor John Long
Vassar College

PUBLISHED BY:

THE GREAT COURSES
Corporate Headquarters
4840 Westfields Boulevard, Suite 500
Chantilly, Virginia 20151-2299
Phone: 1-800-832-2412
Fax: 703-378-3819
www.thegreatcourses.com

Printed in the United States of America

John Long, Ph.D.
Professor of Biology
and Professor of Cognitive Science
Vassar College

Professor John Long is a Professor of Biology and a Professor of Cognitive Science on the John Guy Vassar Chair of Natural History at Vassar College. He also serves as the Director of Vassar's Interdisciplinary Robotics Research Laboratory, which he helped found in 2003. He has taught 27 different courses in four departments and programs, including Perception and Action, a course in the Cognitive Science Department that features robotics and laboratories in which students study and program mobile robots. Professor Long received his Ph.D. in Zoology from Duke University, where he specialized in biomechanics and received an excellence-in-teaching award.

Professor Long is known internationally for his work in the burgeoning fields of biorobotics and evolutionary robotics. As an associate editor, he has helped launch two scientific journals in robotics: *Frontiers in Robotics and AI* in 2014 and *Soft Robotics* in 2013. He creates self-propelled, autonomous models of animals, both living and extinct, in order to study how the animals work, behave, and evolve. He also designs and builds bioinspired robots in collaboration with computer scientists, electrical engineers, mechanical engineers, physicists, and development-stage robotics companies.

Professor Long's research currently is funded by the National Science Foundation (NSF), and he has received previous research funding awards from the NSF, the Office of Naval Research, the U.S. Small Business Administration, and the Defense Advanced Research Projects Agency. He also serves as an expert reviewer of robotics projects for the European Commission. Many students working with Professor Long present their research at national and international scientific meetings and earn coauthorship on scientific research papers, and most go on to careers in science, technology, engineering, mathematics, or medicine.

Professor Long is the author of *Darwin's Devices: What Evolving Robots Can Teach Us about the History of Life and the Future of Technology*, and along with his students and collaborators, he has published more than 50 papers in scientific and engineering journals. (For details, see his Google Scholar page: http://bit.ly/1fFeMrf.) Professor Long and his robots have been featured in the international press; on radio, television, and podcast news programs; and in science documentaries, including *Through the Wormhole* with Morgan Freeman on the Science Channel and *Evolve* and *Predator X* on HISTORY. He also has been profiled in the journal *Science* for a special issue on robotics. ■

Table of Contents

INTRODUCTION

LECTURE GUIDES

Table of Contents

Table of Contents

Disclaimer

This series of lectures is intended to increase your understanding of the principles of robotics. These lectures include experiments in the field of robotics, performed by an experienced professional. These demonstrations may include dangerous materials and are conducted for informational purposes only, to enhance understanding of the material.

WARNING: THE DEMONSTRATIONS PERFORMED IN THESE LECTURES CAN BE DANGEROUS. ANY ATTEMPT TO PERFORM THESE DEMONSTRATIONS ON YOUR OWN IS UNDERTAKEN AT YOUR OWN RISK.

The Teaching Company expressly DISCLAIMS LIABILITY for any DIRECT, INDIRECT, INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOST PROFITS that result directly or indirectly from the use of these lectures. In states that do not allow some or all of the above limitations of liability, liability shall be limited to the greatest extent allowed by law.

Safety

As you work through activities in robotics, it is essential that you follow sound safety procedures. If you are under 18 years of age, you should only proceed with the supervision of an adult. Some of these activities involve potential hazards that include, but are not limited to, flying parts that could harm your eyes; sharp parts that could abrade or cut your skin; mild electric current that could cause momentary discomfort; spinning motors that could entangle loose hair or clothing; solder, soldering irons, hot glue, and hot glue guns that could burn your skin or ignite a fire; and hand tools that could poke holes, abrade, or cut your skin.

To improve your chances of avoiding injury, always start with a clear work surface. Any electric tools, such as a hot glue gun or soldering iron, should never be left plugged in and unattended. Any electric appliance, such as a hot glue gun or soldering iron, should never be used on or placed next to anything combustible, such as paper, clothing, or solvents.

Also, before you begin to work, tie back loose hair and secure loose clothing. Remove any necklaces or bracelets. Keep a charged fire extinguisher and a first aid kit nearby.

Never work alone. With a partner, you will be assured of help in case of an emergency. Make sure that you have a phone handy and that you know the number for emergency help.

When you are done, make sure to unplug all tools and appliances.

Robotics

Scope:

Like computers and self-propelled vehicles of the 20th century, robots are a technological revolution of the 21st century that impact nearly every aspect of our lives, businesses, and security. Robotics unites many fields of science and engineering, and these synergies help us create a new kind of device that we've been dreaming about for millennia: a driverless, self-controlled, goal-driven machine that moves itself or objects, in order to accomplish everything from simple, repetitive tasks to complex missions.

Robotics is the principled study of these remarkable machines with the goal of being able to design, build, test, and operate them. This course begins by introducing the many different shapes and sizes of robots: giant jet planes, tiny nanobots, unwieldy welding arms, barking toys, and low-slung circular appliances.

The principle that forms the backbone of the course across the many kinds of robots is autonomy. An autonomous robot can act on its own to achieve its goals. Those goals are built into its body and programmed into its brain-like computer.

To understand autonomous robots, we spend the first five lectures analyzing them, experimenting to see what they can do and how they do it. We take robots apart, and put them together, and we survey five essential components: bodies, actuators, sensors, energy supply, and controllers.

We learn how to create and program a simple behaviorally autonomous robot: The key is to create a perception-action loop that links sensors and motors. That linkage is physical, electrical, and information-based, mediated through the robot's computer and the physical interactions with the world in which the robot is moving. We use an open-source computer controller called an Arduino, a simple sensor, and a servomotor to build a swimming robot called a Tadro.

More complex robots also take their inspiration from nature. In Lectures 6 and 7, we explore how humans and other animals offer solutions for basic functions that nearly all robots must accomplish. Navigation is central, and robots that use maps and models of the world can make and enact plans. Successful navigation also requires that robots adjust rapidly to the unexpected, including avoiding objects, following walls, and escaping from tight spots. These are reflexive behaviors inspired by the successful workings of insects.

Roboticists need to know some basics of working with wires and electronics, and some simple techniques are introduced in Lecture 8. The do-it-yourself approach is great for putting the principles of robotics to work to build simple robots.

For robots to accomplish complicated tasks, we need to turn to the formal design processes of engineering (Lecture 9), which involves understanding the task that the robot needs to accomplish. Task analysis is explained in Lecture 10, laying the groundwork for breaking down any task into the steps that are needed to build and program a robot.

In Lectures 11 through 18, we explore the diverse environments where we use robots—from factories, homes, hospitals, and the road to the air, the water, space, and the battlefield.

In the last six lectures, we address the latest research and designs for robots. Extreme robots are the focus of Lecture 19, in which we analyze the biggest, the smallest, and the fastest robots, with an emphasis on robots with legs.

In Lecture 20, we examine swarms of robots and how they coordinate their collective movements. We push the limits of the idea of modularity, the semi-independent functioning of elements within a single robot or across multiple robots.

In Lecture 21, we examine what else life does well and find that eating, developing, self-replicating, and evolving provide further goals for robotics—four additional kinds of autonomy. We discover robots that attempt to harvest their own energy, assemble themselves, build copies of themselves, and redesign themselves in response to feedback from the world about their performance.

One of the most exciting uses of robots is to have robots cooperate with and work alongside humans. This requires attention from humans, and the design of attention-grabbing robots takes us to the new field of social robotics (Lecture 22), in which emotional interactions can make or break a human-robot relationship.

Lecture 23 focuses even more directly on humanoid robots. We explore the value of humanoids in environments that cannot be modified to accommodate the robot. For example, first responders during a disaster confront degraded spaces and equipment typically designed for humans, not robots, creating a situation in which humanoid robots might succeed where other robots would fail.

Robotics is a capstone field with multiple origins, and there are multiple futures to be expected, too, including miniaturization, cloud robotics (in which robots share what they learn through the Internet), Watson-style cognitive robotics, biohybrids, and evermore-powerful modularity. Autonomous robots are amazing machines, and the study of robotics allows us to understand and benefit more fully from how they work—and to design and build our own. ■

The Arrival of Robot Autonomy

Lecture 1

The field of robotics offers an endlessly exciting way of seeing our world. Almost everything around you—any machine, any electronics component, any animal, anything that humans do—has potential implications for robotics, which has implications for you. Robotics is a field where many disciplines come together, so the sources of potential inspiration are endless. In this course, you will discover how autonomous robots work and learn the science and engineering behind how they are designed and built.

Robots in Our World

- Mars has **robots** exploring the planet's surface on our behalf. NASA's Curiosity, also known as the Mars Science Laboratory, is 140 million miles away. With a generator powered by nonexplosive plutonium 238, Curiosity uses its camera, along with lasers and abraders for pulverizing rocks, a grabber for taking samples, and a miniature onboard laboratory for analyzing samples, to explore Mars. Curiosity is an explorer and a scientist, and if any signs of life are found on another planet, that discovery may come from a robot.

- In our oceans, we also have robots at work. These robots search for lost airliners, fix leaking oil pipelines, and discover long-lost shipwrecks. Just like with our explorers on other planets, underwater robots work in dangerous conditions and for periods of time that humans cannot.

- For decades, tens of thousands of robots have been added every year to manufacturing plants. Multi-jointed robotic **manipulators** move heavy car doors into place for assembly. Robotic arms weld parts together, and robotic sprayers paint the body. Increasingly complicated teams of robots even assemble cars.

- Robots are no longer working just in caged-off areas; they now work side by side with humans. We call this field collaborative **robotics**.

- Some robots can drive themselves. Thirty percent of Curiosity's driving on Mars during its first 18 months was self-controlled. Using **stereovision** from two cameras, Curiosity created a **drivability map**, an estimate of the danger of the path ahead. By navigating on its own, Curiosity can cover more terrain than would be possible if it had to rely entirely on remote control by humans.

- On Earth, we have driverless cars getting licensed to carry passengers on ordinary roads. They navigate and drive themselves. The whole car is the robot. There are **sensors** on top, helping the robot navigate, and a computer on board makes decisions about which way to steer, how fast to go, and how wide to go when passing another vehicle.

- Robots, such as Husqvarna's lawn-mowing robot, can automatically cut your lawn, and there are robots available to wash windows, such as the Winbot by Ecovacs. Robots also clean our floors, and millions of homes have iRobot's **Roomba** or other robot vacuums on the job.

Lawn-mowing robots make landscaping simpler and less time consuming than it would be if a human had to do it.

- Robots perform other work in extreme environments, for example, search and rescue, after disaster strikes. Robot snakes can crawl through tight spaces, carrying cameras and sensors to show topside humans what's inside and to search for survivors.

- Robots were our first eyes in the Fukushima nuclear plant after disaster struck. The iRobot PackBot brought in a live video feed and took temperature and radiation measurements, and it can even climb up flights of stairs.

Robotics in Popular Culture

1927	Maria, the first robot to appear in a movie, stars in the silent classic *Metropolis*.
1941	The word *robotics* is first used in the short story "Liar!" by writer Isaac Asimov.
1942	The three laws of robotics are created by Isaac Asimov in his short story "Runaround."
1950	*I, Robot*, a collection of robotics short stories by Isaac Asimov, is published.
1982	KITT, a fictional autonomous talking car, appears in the television series *Knight Rider*.
1982	Fictional "replicants" appear in *Blade Runner*, a science fiction film adapted from Philip K. Dick's *Do Androids Dream of Electric Sheep?* (1968).
1987	*Star Trek: The Next Generation*, a television show, debuts with android lieutenant commander Data.

- Robots are in hospitals, helping medical personal treat patients. Teamwork between surgeons and their robotic assistants, such as da Vinci's surgical robot, allows surgeons to perform minimally invasive surgery.

- Robots are also taking education by storm, thanks to efforts from companies like ArcBotics, **Arduino**, LEGO, Robomatter, and VEX Robotics. Robotics clubs and competitions are everywhere and entice people of all ages, from elementary schools to colleges, and even to million-dollar-challenge events attracting some of the best companies in the world.

The Development of Robots

- After World War II ended in 1945, the technological advances brought by that conflict had everyone looking for peaceful uses for weapons. Science fiction took off as a way to think about a new future—to prototype a future.

- Isaac Asimov led the charge, coining the word *robotics* in 1941 and publishing a short story in 1942 called “Runaround,” in which he introduced his now-famous **three laws of robotics**: Don’t injure humans; obey humans whenever that command doesn’t injure humans; protect yourself, the robot, as long as you obey and don’t injure humans.

- Science fiction helped us dream a world of fantastic robots. Robby the Robot starred in the 1956 film *Forbidden Planet* as a good robot, helping humans. Of course, the real Robby was a metal suit worn by a human; we didn’t have real, capable **humanoid** robots in 1950.

- Flying cars, personal helicopters, and trips to the Moon were all being promised. In 1956, a group of computer scientists met to discuss how to make machines that were humanly intelligent. Nine years later, fueled by the advent of electronic computers, future Nobel Prize winner Herbert Simon optimistically predicted that “machines will be capable, within twenty years, of doing any work a man can do.” The pursuit of this goal created the field that we know as artificial intelligence (AI).

- Half a century later, we know that what Simon said was wrong, at least in terms of a delivery date for the promise. But while personal aviation may have stalled around 1980, at least in terms of the number of private pilots in the United States, robots have been a very different story. We now have humanoid robots, such as Honda’s **ASIMO**.

Honda's humanoid robot ASIMO is able to react to its environment and interact with humans.

- It's only in the 21st century that we've seen case after case of robots working not just in the laboratory but also as reliable commercial products, such as the Roomba. We even have robot aircraft, called **drones**, which are used by the militaries of many countries and, as fast as permits are issued, by everyone from delivery services to Hollywood movie makers.

- But why has this process of creating intelligent machines taken so much longer than the 20 years that Simon predicted? The gestation period for the working technology of robots has been much longer than the memory of anyone living today.

- Mechanical figures we call **automatons** have been around since the Middle Ages in Europe and were known in China and ancient Greece long before that. The first serious design for a humanoid robot comes from Leonardo da Vinci working in the late 1400s and early 1500s.

- Leonardo never had a chance to build his humanoid. But from his notebooks, we can tell that his robot had an exoskeleton of armor. Inside, pulley, gears, and cables were connected to move the hands, wrist, elbow, and shoulder. These **actuators**, which are what create movement, might have been able to connect with a mechanical cart, making it possible to reprogram the figure. This combination is what made Leonardo's ideas an advance over the clockwork automatons that had already been livening up public squares in medieval Europe.

- There are three ingredients that are missing from Leonardo's humanoid that we see in today's autonomous robots: electric sensors, electric motors, and digital electronic computers. These three ingredients combined result in robotic complexity that we don't see in purely mechanical machines.

- The great virtue of a digital electronic computer is the algorithms we can put into the computer itself. Those algorithms, which can be reprogrammed easily in software, take the place of hardware circuits that have to be rebuilt.

- This kind of potential to make machines more intelligent is what Simon and other computer scientists got so excited about—and rightfully so. But what caused Simon's timing to be overly optimistic is that we knew less about how humans and animals work than we thought.

Types of Robots

- The moment we started to try to create machines that could see—a field called computer vision—we realized that identifying an object in a visual field was extremely complicated. The robot named **Shakey**, a mobile robot from the late 1960s, was the first to have and use object identification to move around a room. Shakey had rudimentary vision and the algorithms to turn a pattern of light intensity into information about objects in the world.

- It was in Shakey that everything came together: electric motors, electric sensors, and electronic computers. Shakey was a mobile robot that could sense its world, reason about the state of the world and its place in the world, make plans about how to move in the world, and then enact those plans.

- Shakey had a computer, **sonar range finders**, a video camera, and bump detectors. Shakey was built to navigate an internal space, much like a simplified office building. It was given a map of that space that included objects like walls and blocks. Given the presence of walls and objects, Shakey calculated a path to its destination and moved forward and around objects.

- Shakey was a robot that had intelligence, as seen through its behavior. Its intelligent behavior set the standard for modern mobile robots, and nearly every mobile robot today owes something to Shakey.

Robotics Firsts

~1495	Leonardo da Vinci designs a mechanical knight using pulleys and cable, as well as a mechanical cart capable of serving as a programmable controller.
1897	The remote control is first patented and demonstrated on a model ship by Nikola Tesla.
1956	*Artificial intelligence* is coined and launched as a research area by John McCarthy and others at a conference at Dartmouth College.
1968–1970	Shakey, the first autonomous mobile robot with a digital electronic computer, is built by Stanford Research Institute.
1984	*Vehicles*, a book by Otto Braitenberg, lays the groundwork for the design of the simplest-possible autonomous robots.
1990	The iRobot Corporation is founded by Rodney Brooks, Colin Angle, and Helen Grenier.
2000	ASIMO, a humanoid robot, is introduced by Honda Motor Corporation.
2002	Roomba, an animal-inspired robotic vacuum cleaner, is introduced commercially by iRobot.
2005	Arduino, an inexpensive open-source microcontroller, is created by Interactive Design Institute in Ivrea, Italy.

- In addition to mobile robots, the other great class of robots is stationary, and we call them robotic manipulators. While they don't move around to get their work done, they move an appendage, often an arm, in order to pick up and move an object in the world.

- There is more than one way to put a manipulator on a mobile robot—that is, there is more than one type of possible robot body for almost any given task. In fiction, robots most often have bodies like humans. Humanoids are mobile, and they have manipulators.

- But robots do not have to be built to resemble humans to take advantage of biology. Robots can be inspired by the bodies of other animals. The robot BigDog, built by Boston Dynamics, is a famous nonhumanoid **biomorph**. It's been modified to help combat troops transport huge loads of equipment over rough terrain. It's built like a dog, with four legs that have the basic anatomy of dog legs.

- By contrast, robots with no biological inspiration can be called **mechanoids**, a category that would include wheeled vehicles, such as Shakey and the Mars rovers.

- Other robotic devices are worn by the user and can be strapped to the outside of your body, leading to the expression *wearable robotics*. For example, there's a walk-assist device that can help people with muscle weakness from disease. It was a spin-off from technology developed by Honda to build the humanoid ASIMO. It's a robot in the sense that it has sensors, a computer, and motors. Some people like to call these wearable robots exoskeletons.

The Land Walker robot is an example of what some might call an exoskeleton.

- In still other cases, robotics are being implanted in humans in a more permanent way, such as the DEKA Arm. The term "wearable" robot doesn't work for the DEKA Arm; it's a smart **neuroprosthetic**. It uses electric signals from the human to grasp, lift, and release. What sets apart robotic prostheses is that they have onboard intelligence, coupled with sensors and motors, components that we would find in any robot.

- Some of the greatest advances in robotics during the past few decades have come from learning to simplify robots—to simplify demands on their computers in particular—making them less like a human brain. The Roomba vacuum robot moves much faster than Shakey in part because Roomba does not try to figure out its environment before making a move. Instead, Roomba gets right to work, taking its inspiration from animals, figuring out what to do next while already moving.

- Roomba is an example of **behavior-based robotics**; Shakey is an example of **model-based robotics**. These terms refer to the different strategies that we take in the software of the **controller**. And they are not mutually exclusive; most modern robots use a combination of these two architectures.

- Intelligent robots, such as Roomba and Shakey, operate on their own—it's a closed loop. Shakey has **servomotors** responding to feedback from the sensors. Operation on its own is referred to as behavioral autonomy.

- Compare that autonomy to a robot that is completely remote controlled, with a human sensing the world and then determining the actions of the robot. In this case, most or all of the intelligence of the robot is located in the human, not the robot. A behaviorally autonomous robot, acting on its own, has its own intelligence. It doesn't need a human in the control loop.

Activities

Moving the Robot

Check out this tutorial from ArcBotics, maker of Sparki: http://arcbotics.com/lessons/moving-the-robot/. It's a great introduction not just to Sparki but also to any robot that moves using differential drive. The principles here will work on any robot with two actuators moving in a plane.

Building a Tutebot

Tutebot is a very simple robot, making it a great place to start building your own. *Mobile Robots* by Jones, Flynn, and Seiger has the full instructions. Chapter 2 also includes an introduction to the electronic components and how they work.

Tutebot does not have a programmable digital microcontroller. To "reprogram" it, you have to change the electronics: It is an analog computer. If you build your circuit on a breadboard, then you can rearrange components more easily than if you create a soldered circuit.

You can also "reprogram" this analog computer by altering the settings on the potentiometers using a small screwdriver. The circuit has four potentiometers; they are the little blue boxes with the white screw on top. By changing the resistance of the two potentiometers linked directly to the capacitors in the back, you control how quickly the capacitors release stored charge, which determines how long the motors operate in reverse.

Capacitors are a special type of energy storage device. They are charged quickly when the switch on the bumper is activated and Tutebot begins to back up. If the potentiometers are set to low resistance, the capacitors discharge rapidly when the bumper is released. If we set the potentiometers to a high resistance, then the capacitors drain more slowly and the motors will keep spinning in reverse for a longer period of time. If you set the two potentiometers at different resistances, then Tutebot will back up and turn in an arc. This reverse-and-turn motion is what gives Tutebot its ability to perform wall-following behavior.

Important Terms

actuator: The moving parts of a robot that allow it to act; any part, appendage, or mechanical system that uses motors to move a robot or manipulate the world through movement.

Arduino: An open-source hardware and software company known for microcontrollers.

ASIMO: Type of humanoid robot emphasizing bipedal mobility over uneven terrain; in development since 2000 by Honda.

automaton: A mechanical machine shaped like a human or animal that works automatically without feedback from sensors or direct control by humans.

behavior-based robotics: Design of robots that eliminates or minimizes the use of internal world models and maximizes the use of reflex-like sense-act modules.

biomorph: Any robot modeled after a life-form.

controller: The mechanical, electronic, or computerized part of a robot that converts information provided by sensors into instructions—whether to adjust actuator behavior or overall motion, update maps and other internal models, or provide information for human operators.

drivability map: A planning model that is continually updated and used to plot the immediate course for a robot.

drone: Any unmanned aerial vehicle (UAV), especially one that can fly autonomously (using GPS or other navigational data) and beyond the line of sight needed for radio-controlled (RC) aircraft.

humanoid: A robot designed to look and function like a human. See **Actroid**, **android**, and **cyborg**.

manipulator: A robotic arm that grasps and moves objects; also, any stationary robot that has one or more such arms.

mechanoid: A type of robot built without inspiration from biology.

model-based robotics: Design of robots that maximizes the use of internal world models, ongoing and sophisticated planning algorithms, and complex goals and tasks.

neuroprosthetic: Type of robotic manipulator used as a replacement limb for human amputees and controlled by muscular or neural signals.

range finder: An active sensor that broadcasts ultrasonic or other sound waves and then uses the time it takes for the echo to return to measure the distance to the reflecting surface. See **sonar**.

robot: A type of machine that can be remote controlled, partially autonomous, or fully autonomous as it moves itself or objects in order to carry out tasks. While robots always have controllers and actuators, remote-controlled robots may lack onboard sensors.

robotics: The field of study and inquiry that develops principles and approaches for the design, fabrication, operation, and control of robots.

Roomba: The floor-cleaning robot introduced in 2002 by iRobot; the first fully autonomous home robot to achieve commercial success.

sensor: Any device that detects changes in physical properties or energy patterns in the world or the robot and converts those into electric, chemical, or mechanical signals usable by a controller to adjust actuator behavior and overall motion, update maps and other internal models, or provide information for human operators.

servomotor: An electric DC motor that uses sensory feedback from an internal potentiometer to precisely control position and movement.

Shakey: The first mobile, autonomous robot controlled by a reprogrammable digital electronic computer.

sonar: Stands for sound navigation and ranging; typically an active sensor system that broadcasts sound waves and then uses the pattern of the returning echo to measure the distance and size of objects.

stereovision: Visual sensing that uses two cameras focused on the same object or scene to provide information about depth, range, and three-dimensional shape.

three laws of robotics: Introduced by Isaac Asimov in the short story "Runaround" (1942). First law: A robot may not injure a human being or, through inaction, allow a human being to come to harm. Second law: A robot must obey the orders given it by human beings, except where such orders would conflict with the first law. Third law: A robot must protect its own existence as long as such protection does not conflict with the first or second laws.

Suggested Reading

Bekey, *Autonomous Robots*, chap. 1.

Jones, Flynn, and Seiger, *Mobile Robots*, chap. 2.

Rosheim, *Leonardo's Lost Robots*. chap. 3.

Smithers, "Autonomy in Robots and Other Agents."

Robot Bodies and Trade-Offs

Lecture 2

By taking apart and building, or analyzing and synthesizing, robots, we learn about how robots work. When we do this, we see that we need the five categories of parts: sensors, actuators, controllers, an energy supply, and a body. And there is a sixth category for the support system, which supports, and may change, the way that the robot body interacts physically with the world. Robot bodies teach us a universal lesson about how robots work: There are always trade-offs. The need for movement, and the kinds of movement needed, drive the design of robot bodies.

Robot Parts

- There are two different methods that scientists and engineers employ to figure out how a robot works: analysis and synthesis. Roughly, analysis translates into taking stuff apart. Scientists dissect and destroy in order to build a broader understanding. Synthesis translates into putting stuff together. Engineers design, assemble, build, and construct.

- Even though analysis and synthesis are different, they are entirely complementary ways to figure out how something works. Robotics needs both. Neither the scientific nor engineering approach is better. Engineers need the scientists' understanding of what the bits and pieces are and how they work in order to build something. Scientists need to have instruments and devices designed and built by engineers in order to take new stuff apart, and engineers can tell the scientists when they need to know more about the parts.

- The parts of a robot can be organized into five functional categories that all robots have: sensors, actuators, controllers, an energy supply, and a body.
    - Sensors detect changes in the world. For example, Roomba has a bump sensor, which can detect when the robot touches something.

iRobot's Roomba has sensors, actuators, a controller, a battery, and a chassis that enable it to vacuum independently.

- Actuators create movement. Anything with a motor is an actuator. The actuators for Roomba fall into two categories: actuators that move Roomba and actuators that spin brushes or fans to create movement of the dirt.

- The controller is an electronic computer and the software loaded onto the memory of that computer. The controller links and coordinates the functions of the sensors and the actuators. The controller is located on Roomba's **motherboard**.

- The energy supply is the battery. And the battery gets recharged at the home base, so even though the home base isn't part of the robot, it is part of the energy supply chain for Roomba.

- The dustbin, filter, and top and bottom covers of Roomba are parts of the body. They aren't sensors, actuators, controllers, or the energy supply. The **chassis** is the main frame of the body.

- Sometimes, these anatomical categories are a bit fuzzy. For example, sometimes it's helpful to think of the body as not just anything that isn't a sensor, an actuator, a controller, or the energy supply but as all of the parts put together—the whole robot. And roboticists sometimes switch between these different meanings without telling you, for example, whether "body" is the whole robot or just the chassis and related parts.

- Either way, there's even more than the whole robot to consider. There are some other items that get sold with Roomba, for example, that are not part of Roomba's body in either sense of the word. These are parts of what we call the robotic system, including the home base, which charges Roomba's battery; the virtual wall lighthouse, which communicates with Roomba about where to go and where to avoid; and the wireless command center, which is the remote control. The support system is really a sixth kind of component for an autonomous robot.

- As soon as the human remotely controls the robot, it's no longer autonomous; a human is in the functional loop, making decisions in place of the robot's controller. In fact, a human is almost always in the background, maybe only in the deep background, as another part of a robot's support system. When and how a human enters the control loop is always important. That's why many robots have a built-in ability to do both: be autonomous and be remotely controlled.

Trade-Offs

- As with any animal or robot, evolving or designing it for a particular task involves trade-offs. You can't do everything well. So, while Roomba does its job—cleaning floors really well, even transitioning from bare floors to carpet—it is not built for moving up and over, or down and over, steps or drops. In fact, it has cliff detectors to keep it from tumbling down stairs. The trade-off is that it can clean but not climb.

- You might think that this is not really a problem; it's not a trade-off if Roomba never encounters stairs or other obstacles. But many domiciles have stairs that lead to other places you might want Roomba to clean. The short-term solution is to get a Roomba for every floor of your house, but that's expensive, and it still leaves the stairs uncleaned.

- With stairs and other obstacles in mind, iRobot, the maker of Roomba, has designed a different kind of robot: PackBot 510. It has a very different body when compared to Roomba. PackBot has tracks instead of wheels, and it has a compound track with a joint in the middle that lets it get up stairs. The design of PackBot's body converts stairs into an elevated track. However, PackBot is too large to clean under furniture. So, the trade-off for PackBot is that it can climb but not clean—the opposite of Roomba.

- Unlike wheels, tracks do not get caught in the pits and valleys of uneven terrain. Just like with stairs, PackBot's tracks are great for moving over debris. So, we can think about a trade-off at the level of the actuators. Wheels can be small and take up just a small part of the body, while tracks tend to be larger and take up a large part of the body. So, one trade-off for these actuators is between size and traversability.

- We see trade-offs like this in other robots; trade-offs are built into any design. With drones, or aerial robots, the body's actuation system has propellers, and the body is very lightweight. So, it is a system built for flying.

- A **quadcopter**, or quadricopter, has four propellers. Quadcopter drones are highly maneuverable. They can hover, flip, and twist with ease. That's why a drone can be so good for taking aerial pictures. It may even have a whole cinematography mode built in.

© ivansmuk/iStock Editorial/Thinkstock.

Drones are useful in aerial photography due in part to their maneuverability.

- Compared with fixed-wing aircraft—what we call airplanes—helicopters suffer in terms of energy requirements. A plane generates lift by moving its wings forward, while a helicopter uses propellers. For a robot body designed as a helicopter or quadcopter, the trade-off is maneuverability versus efficiency.

Robotic Manipulators

- We can see differences in robot bodies that relate to the environment. Terrestrial robots have wheels and tracks. Aerial and aquatic robots usually have propellers. Robots that move quickly must be streamlined, whereas slow-moving robots will have a shape determined by other considerations, such as how to move through confined spaces.

- So far, the focus has been on differences in locomotion: tracks, wheels, propellers, and flippers. For mobile robots, which are built to move themselves around, the needs of that mobility drive the design of the body.

- But there are other robots, called static robots, stationary robots, or manipulators, that stay in place and move objects rather than themselves. This is typically a robotic arm, so the movements of the arm dictate the design of the body.

- **Baxter** is a type of robotic manipulator that has two arms and can do a type of task that is very important in many industrial situations: pick up objects and place them somewhere else. This is called **pick and place**. Packing, unpacking, sorting, and supplying are all jobs that involve picking and placing. Picking and placing is what we humans do when we reach for, grasp, and move an object.

- Very fine and accurate picking and placing in three dimensions requires many joints. For Baxter, every reach and every movement of an object involves the controlled motion and coordination of the joints. For roboticists, the human arm has been a source of inspiration. If a joint is a hinge joint, such as an elbow, it can only bend in one plane. In engineering terms, we say that the elbow joint has one degree of kinematic freedom.

- The shoulder joint, however, is a ball joint, and it can elevate and depress, abduct and adduct, and supinate and pronate very complex movements with a single joint. The shoulder joint has three **degrees of freedom**; each type of rotation operates at 90 degrees to the other.

- When we think about the joints in the human body, we almost always talk about rotation. But we can define motion along a straight line as translation. Baxter's manipulator, or grabber, has fingers that move along a linear track in order to close around an object. They have a single degree of freedom, in translation.

- More degrees of freedom mean more maneuverability, more space that can be reached, and more and different kinds of objects to be handled. But there are trade-offs. Each degree of freedom needs its own motor, and more motors mean more power consumed. Also, more degrees of freedom mean that the mathematics of coordinating the motion of the whole arm becomes more complex. So, more degrees of freedom mean that you need a bigger and faster controller.

- No matter whether you are looking at the body of a robotic manipulator or a mobile robot, you always see trade-offs. Taking a robot apart helps us see additional trade-offs. We can see how actuators and sensors work to give certain kinds of behaviors, and we see how, for example, actuators themselves have trade-offs. But another way to think about trade-offs is to recognize them in the robots that you are building yourself. This is the value of synthesis: understanding by building.

Activity

To build a Tadro for yourself (see Lecture 5), you will need the following electronics, all of which you can order from SparkFun Electronics (www.sparkfun.com).

- Arduino Uno microcontroller, dev-11224
- USB cable, rtl-10423
- 9V battery holder, prt-10512
- Servomotor, generic, high-torque, rob-09347
- Mini photocell (photoresistor), sen-0988
- Fixed resistor, 10 kW, com-08374

You also need a circular, plastic food container; a short length of wooden dowel; duct tape; a hot glue gun; hot glue; and a 9V battery.

Important Terms

Baxter: A two-armed robotic manipulator introduced by Rethink Robotics in 2012 for light manufacturing tasks, featuring rapid reprogramming and safe interactions with humans.

chassis: The primary structural support system of a robot's body.

degrees of freedom: The number of independent motions available in a joint, structure, or robot, whether the motion is linear (prismatic, translational), angular (revolute, rotational), or spherical (like a ball joint).

motherboard: The main and largest printed circuit board in a computer.

pick and place: Fundamental task for robotic manipulators, involving grasping, moving, and then releasing an object.

quadcopter (quadricopter): A rotary-winged aircraft with four propellers.

Suggested Reading

Braitenberg, *Vehicles*, chaps. 1–3.

Pfeifer and Bongard, *How the Body Shapes the Way We Think*, chaps. 1–2.

Other Resources

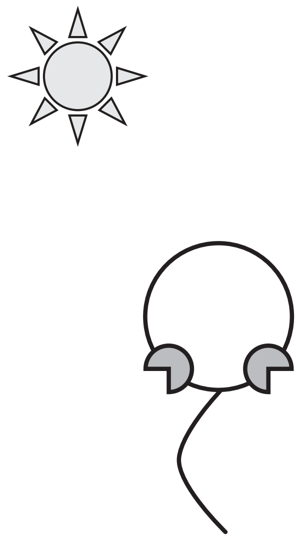
Bugworks (http://www.sussex.ac.uk/Users/christ/bugworks/). This simulator models Braitenberg's vehicles. Wire and rewire from sensors to motors and see how rearranging the body creates different behaviors.

V-REP (http://www.coppeliarobotics.com/). This robot simulator allows you to quickly test different types of bodies and see how they impact behavior and function. Simulate a huge array of robots, including Baxter. Try the free educational version.

Questions to Consider

1. Think about functional trade-offs in the design of the human body: What do we trade off for our upright, bipedal form of locomotion?

2. You can have a robot behave differently just by changing its body and nothing else. If you want a two-eyed Tadro to swim *away* from a stationary overhead light, which of the following bodies would do the trick? Note that Tadros swim continuously and that their eyes (looking like pies with a piece missing) can only pick up light in the direction of the opening.



Decide how the information from the two light sensors is used to calculate where the light source is relative to the robot. How could you convert that sensor information into an instruction for Tadro?

Robot Actuators and Movement

Lecture 3

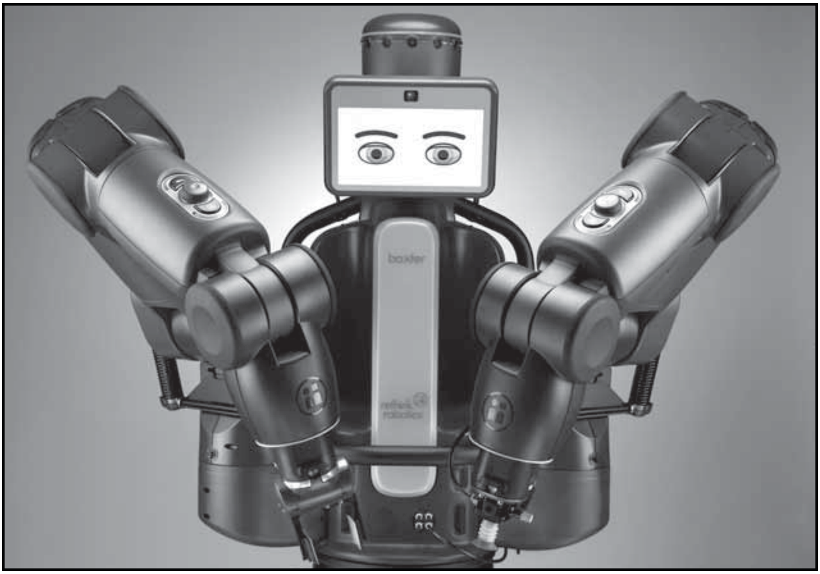
Actuators of all kinds define how a robot moves, how it moves the world, and how it can change the world by its movements. Movement defines what a robot is, but actuators—the motors and transmissions underlying robot movement—can do other things to change the world besides move. Servomotors, with their embedded and tightly linked internal sensors, steer us toward an understanding that actuators need sensors. For an autonomous robot, all of its motors, and all of the actuators that they move, are responding at some level to information provided by sensors. Many machines move, but autonomous robots need both movement and sensors.

Actuators

- Robots are machines that move with purpose, to achieve their goals and to get work done. They move objects. They move themselves. Roomba, the home robot from iRobot, moves itself to clean the floor. Roomba is a mobile robot that moves and maneuvers using two wheels that are independently motorized. By independently controlling the speed of each motor—and, hence, each wheel—Roomba is able to go forward, reverse, and turn in what is called differential steering.

- Baxter, the manufacturing robot from Rethink Robotics, moves objects to do things. Baxter is a robotic manipulator built to work alongside humans in a manufacturing environment. For any task you might design for Baxter, the key is that Baxter moves its body in order to move objects. Baxter's movements are created by motors. Baxter has many different joints, each of which is powered by its own motor. Each motor either is an actuator or is part of an actuator. It's the full actuator that moves the robot.

- In robots, we take the complex motion of a single joint and decompose it into its fundamental elements. For the shoulder, three-dimensional motion takes three separate joints, each restricted to a



© Rethink Robotics, Inc.

Baxter is a robotic manipulator built to work alongside humans in a manufacturing environment.

single degree of freedom—in this case, angular motion in a plane that can be described by a single variable, such as the angle between the skeletal elements.

- Rethink Robotics calls joints with a hinge a bend joint, because it changes the angle of the elements to which the joint is attached. The other kind of joint they call a twist joint, because the immediate elements don't change orientation relative to each other.

- We create three different joints—two twist joints and one bend joint—in the shoulder for two reasons. First, most motors are built to power angular rotation with a single degree of freedom. For example, when we connect a **direct current** (DC) electric motor to an energy supply, we create an electric circuit, and the motor rotates. The motors that power Baxter's joints are a special kind of DC electric motor called a brushless servomotor, which includes a sensor so that the motor's motions can be carefully controlled.

- The second reason to create three separate joints is control. When we decompose complex three-dimensional motions into separate two-dimensional ones, then the motions are easy to control and coordinate when we link up the motions of multiple joints.

- In Baxter's shoulder, each plane of each joint is nearly at 90 degrees with respect to another joint. So, if we want to control the position of the **end effector**—the hand—at the tip of the arm, then we have just three motors to operate.

- Compare that with your shoulder, which has more than 10 different muscle groups, depending on how you count them, and every muscle group has its own innervation, or neural control, that is independent and has to be coordinated with other muscles. Baxter's shoulder is an elegant solution: three degrees of freedom in rotation and full three-dimensional motion.

- Baxter's great range of motion is made possible by the configuration of its joints and the motors that drive them. What makes Baxter safe for humans to work with is that it has **compliant actuators**. The motors transmit their force to the segments of the arms through a transmission system that has springs.

- These springs mean that the joints give way when the arm hits a human or runs into something. Also, we can back-drive the motors, or push the joints backward and get them out of our way. Finally, Baxter moves at speeds that don't create too much kinetic energy. Combine that with Baxter's relatively lightweight arms and padding, and it is a safe companion for side-by-side cooperation.

- The end effector and the motor are connected by the transmission—gears and shafts that not only transfer the motion from the motor to the end effector but also further transform it by changing the motion's speed, leverage, and **torque**. Together, the motor, transmission, and end effector make an actuator.

Automatons

- An automaton is a machine with a hidden mechanism that operates automatically. Centuries ago, even a clock might be called an automaton, but by the 18th and 19th centuries, an automaton came to mean any mechanical figure that simulates the movements of living beings, especially humans.

- But while their movements may be lifelike, these machines aren't autonomous in a robotic sense: They don't use sensors to react to changes in the world, and they don't use controllers to change their behavior in response to sensors. They are automatic in the sense that you turn them on, they move, and they run through their routine.

- The springs in toys, automatons, and mechanical clocks are very useful motors, but they have to be wound up regularly. In terms of physics, when we wind up or load a spring, we are adding potential energy to the machine by using the muscular energy of our human bodies. Energy transfer is all around us, all the time, and the trick, for robots, is to put the energy in a form that creates movement.

- When it comes to creating movement in robots, probably the most important kind of motor is the electric motor. And all motors work on a similar principle: Motors use electricity to spin magnets.

Electric Motors

- Batteries are a common source of energy for actuators in robots, so it's important to understand how batteries work. When we connect the poles of the battery to the wires of the motor, the DC motor and the battery create a physical connection called an electric circuit. The name *circuit* is used because of the idea of tiny electric particles traveling in a circuit from one end of the battery, through the motor, and back to the other end of the battery.

- The particles are charges in two different ways, positive or negative, and opposites attract. You put a bunch of positive charges on one side of the circuit and let them run to a negative attractor on the other side of the circuit. This attraction of opposites is the force that drives the flow of electric particles, a flow that we call the electric current.

- In a DC motor, the electric charges flow directly from one pole of the battery, through the motor, to the other pole of the battery. In contrast, in an alternating current (AC) motor, the charges vibrate—or alternate—back and forth rather than traveling like water in a pipe, which describes a direct current.

- Batteries contain two different chemicals. The battery is designed to keep those chemicals physically apart. You can think of one chemical living at the negative end and the other chemical living at the positive end. One of those chemicals provides the charged particles, and the other chemical collects them. And they exchange those particles through the electric circuit.

- Voltage is a way to measure the strength of the attraction between the different chemicals in the battery. The higher the voltage, the greater the attraction. And the greater the attraction, the faster the flow of the particles through the circuit.

- When a fully charged battery is not part of an active electric circuit, we say that it has potential energy—that is, the separation of the charges gives the battery the potential to make the circuit work. As soon as we turn the circuit on by throwing the switch or connecting the wires, then we turn the potential energy into actual energy as the charged particles get moving.

- In other words, when we start to use the potential energy, it isn't potential anymore; it's the energy in the motion of the charged particles. That electric current, the movement of charge, causes the motor to move.

- Electric motors have two main parts: The part that spins is called the **rotor**, and the part that stays still is called the **stator**. An electric motor works by creating magnetic fields of charged particles that spin the rotor. The stator creates a magnetic field, and the rotor does, too. And you can reverse one of the magnetic fields back and forth so that you can pull and then push the rotor around.

- An electric motor, as part of an actuator, converts the movement of molecules into the movement of a robot.

Servomotors

- For roboticists, one of the great innovations in electric motors has been the servomotor. Unlike the DC motor, which has two wires, the servomotor has three. Two of the wires are for the electric circuit, and in fact, they go to a small DC motor inside. But the third wire is for a control signal.

- The signal, called a **pulse-width-modulated** (PWM) signal, is sent to the servomotor from a computer. The width of the pulse, which is really its duration in time, tells the servomotor where to position its rotor on a circular dial. And that signal is sent every 20 milliseconds—the attention to positional detail that gives servomotors their precision.

- How does the motor know when it has gotten to the right position? The servomotor has to have a sensor. A servomotor needs feedback from the world, and it gets that feedback from a sensor that can tell it where it is, in a rotational sense. In the case of a servomotor, the sensor is a **potentiometer**.

- A potentiometer is a variable resistor. A small version of a potentiometer is built into the servomotor. The potentiometer rotates with the rotor of the motor. The potentiometer then provides feedback to the motor, letting it know where it is rotationally.

- You can think of the feedback as the answer to the question of whether the motor is in the correct rotational position. If it is, then the motor should hold its position as long as the PWM signal tells it to. If it is not, then the motor should try to move into that position. Servomotors are so useful in robotics because of their precise control of motion.

Important Terms

compliant actuators: Motors and linkages made of flexible, soft materials.

direct current (DC): A type of electric current that is unidirectional.

end effector: A tool or other distal element on a robotic manipulator; the part of an actuator that interacts directly with the world.

potentiometer: Type of sensor that converts a mechanical rotation into an electric change in resistance; also known as a variable resistor.

pulse-width modulation (PWM): Method of encoding information in an electric signal by varying the duration of square-wave pulses.

rotor: The part of a DC motor that spins.

stator: The stationary part of an electric motor.

torque: Force applied through a moment arm to rotate or twist.

Suggested Reading

Jones, Flynn, and Seiger, *Mobile Robots*, chaps. 6–7.

Monk, *Hacking Electronics*, chaps. 1–2.

Scherz and Monk, *Practical Electronics for Inventors*, front flyleaves and chaps. 1–2.

Other Resources

Good sources for purchasing actuators and electronics:

Digi-Key: http://www.digikey.com/.

Makershed: http://www.makershed.com/.

SparkFun Electronics: www.sparkfun.com.

Learn about different kinds of motors at Micromo (www.micromo.com/technical-library/technical-library).

Questions to Consider

1. How is a servomotor different from a DC motor?

2. The three most common motors that you'll find in simple robots are DC brushed motors, servomotors, and a special kind of brushless DC motor known as a stepper motor. Which motor is the best choice for each of the following applications?

   a. Power a tank track drive on a mobile navigator.

   b. Precisely move and then hold position of a robotic arm.

   c. Flap the tail of a Tadro.

3. Some roboticists think of an actuator very broadly as any device that changes the world by altering patterns of energy. By that definition, which of the following devices is an actuator?

   a. light-dependent photoresistor.

   b. electrical switch.

   c. light-emitting diode.

Robot Sensors and Simple Communication

Lecture 4

A sensor is any device or mechanism that registers something happening in the world and converts that event into a signal that can be transmitted to other parts of the robot. There are two main types of sensors: passive and active. Passive sensors are built only as receivers, while active sensors are built to send and receive. For robots, the specific sensors deployed—whether single, in arrays, or of different types—determine what they can know about the world. What they do with that information, how they act in response to their sensor readings, marks the origin of intelligent behavior in robots.

Sensors

- Sensors respond to many different kinds of events, everything from changes in light to the presence of dangerous gases. In this modern age of electronic systems, sensors convert their response into a common currency: an electric signal that the robot's computer can read.

- For robots, sensors determine what they know about the world. Because the common currency of sensors is electricity, we look at how sensors change the resistance, current, or voltage in an electric circuit.

- **Photoresistors**, also called light-dependent resistors, are sensitive to light. **Ping sensors**, also known as range sensors, can figure out how far away an object is. Infrared proximity sensors can also detect distance.

- A potentiometer is a variable resistor that can be used to signal rotational position. This is how it is used in a servomotor, as part of the feedback system to allow the motor to achieve precise control of its motion.

- **Accelerometers** convert mechanical vibrations caused by acceleration into an electric signal. Tiny microelectromechanical accelerometers are common in smartphones and robots. They work by converting the deflections of a tiny cantilevered beam with a mass at the end into a change in electric voltage.

- Microphones convert sound pressure waves into electricity by vibrating a membrane that oscillates a small magnet that induces a change in an electromagnetic field.

- Digital compasses are tiny magnetometers that are sensitive to the Earth's magnetic dipole and convert that field into orientation relative to magnetic north.

Examples of Types of Sensors

- We can attach the two wires of a photoresistor to the positive and negative leads of a device called a **multimeter**, which can measure multiple features of an electric circuit or electric parts, such as the voltage of a battery, the amperage of current flow, or the resistance of the circuit.

- If we set the multimeter to measure resistance, then if we cover up the photoresistor, the resistance increases, but if we let light fall on the sensor, the resistance decreases. This sensor gives a continuously variable response to continuously variable changes in the world.

- In contrast, a touch or bump sensor gives a simple yes/no response to touch. We can turn this circuit on by turning on a switch. While the touch sensor is a mechanical switch, with an on or off position, the photoresistor creates a continuously variable response by using the energy of photons to knock electrons onto the band through which electricity is conducted. More light means more electrons. More electrons mean easier flow of electricity and, hence, less resistance.

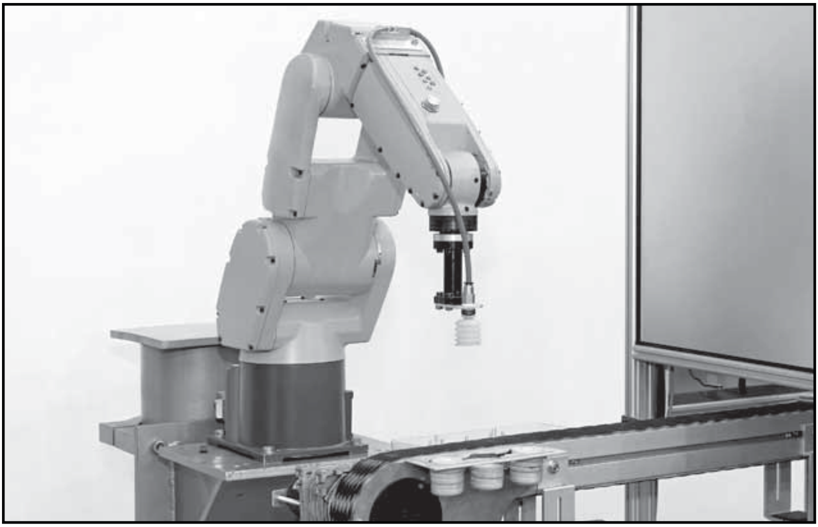
- To take the change in resistance and get it to produce an electric signal for the robot, we have to incorporate the photoresistor into an electric circuit. We can use the photoresistor to build a simple circuit that acts as a light meter. We couple this circuit with a computer, the Arduino microcontroller that is used in many robots.

- Because of the way that the analog electric circuit is put together, the LED (**light-emitting diode**) lights get brighter when more light falls on the photoresistor. In other words, the LED lights must be getting more voltage supplied to them in response to the change in resistance of the photoresistor.

- Another type of sensor makes use of the Hall effect, which is a sideways disruption of the flow of electric current that was first discovered in 1879. That disruption may be caused by a magnet, and it creates a change in voltage that can be detected and measured by a **Hall effect sensor**. By the careful placement of magnets, Hall effect sensors can be used to measure electric current flow, as well as the position and velocity of motors and other moving parts.

- We can understand the importance of sensors if we think about what life is like for a robot without any sensors. The first factory robots had none. They were robotic manipulators, or arms, that would go through the motions with very high accuracy and speed, without having any feedback. Movement without sensing is not intelligent.

- The counterexample is all sensing and no movement. A weather buoy, located in the ocean, is one such device. It carries a variety of sensors. It has an anemometer, up on the tip of a pole, that measures the speed of the wind. That same sensor can measure the direction of the wind using a weather vane. The buoy can also be outfitted with sensors to measure the temperature of the air and the water, barometric pressure, humidity, wave action, currents, water chemistry, environmental factors, and pollution.



© Baloncici/iStock/Thinkstock.

Although robotic arms are precise and fast, they do not receive feedback and therefore cannot be considered intelligent.

- Weather buoys may float freely or may be tethered. Solar panels charge onboard batteries that power the sensors, the data logger, and the communication system. Weather buoys communicate via satellite to computers on land.

- Is a weather buoy intelligent? Movements with purpose, such as seeking light, are the hallmark of an intelligent robot. Robots are built to move themselves or objects. Without moving itself or objects in response to its sensor readings, the weather buoy isn't intelligent. To be fair, it isn't a robot, either.

Bump Sensors

- Tutebot is a robot that has a very simple type of intelligence: following walls. As soon as you turn Tutebot on, the action starts. Tutebot hits a wall. Then, Tutebot backs up and turns. Then, it goes forward. It hits the wall again. Tutebot is working its way along the wall. It is doing a behavior that we see in robots called wall following.

- Tutebot has a bump sensor, which is made up of two spring-loaded push switches. Push either and the same thing happens: The motorized wheels reverse direction momentarily. This is a case where the switch of a bump sensor is switching the current from running in one direction through the motor to running in the opposite direction.

- Rotation of what we call a direct current motor depends in part on the direction of the current flow through its coils. Reverse the direction of the current by switching the positive and negative poles in the connection to the battery, and the motor reverses its rotation.

- You can think of the bump sensor as asking whether Tutebot has run into something. The answer is either yes or no. We can also see simple switches working as bump sensors in a sophisticated robot like Roomba. Roomba's bump sensor is really an array of 11 separate sensors. By having so many, Roomba can know where on its front bumper it has hit an object and then turn in the opposite direction.

- When Roomba is in action, if it bumps on the right side of its body, it backs up and turns to the left. If it bumps on the left side, it backs up and turns to the right. For Roomba, the question this sensor array is asking is more sophisticated: Have I run into something, and if I have, where, in relationship to my body, is it?

Active Sensors

- Sparki, made by ArcBotics, does not have bump sensors. To avoid objects, Sparki has an ultrasonic range finder, sometimes called a ping sensor, and it uses this sensor as a kind of remote object-detection system. This sensor works in the same way that **echolocation** does in dolphins and bats.

- Bats fly at night and hunt moths. They can't see, so in order to chase down a mobile insect and avoid flying into trees and buildings, they basically shout out and listen for the echo of their shout. We don't hear bats do this because the frequency of the sound is much higher than our human ears can hear—ultrasound.

- The ping sensor works the same way, with ultrasound. One side of the sensor is the voice box, a speaker that produces the sound. The other side is the ear, the receiver.

- Using sound for active sensing in engineered systems has been around for 100 years. Sonar, which is short for sound navigation ranging, was originally developed for sensing underwater, but now, it is used in air, too, such as in a pinger for a robot.

- There are both passive and active sonar systems. The passive sonar system just listens. The active one echolocates. The active system, such as the one in Sparki, is very common in robotics. The passive system is basically a microphone. In fact, some robots have microphones, and they will respond to a clap or a loud noise.

- In practice, it's difficult to build a robot that makes very intelligent use of a simple microphone. The problem is that the world is really, really noisy. So, finding a signal that means something to the robot is difficult.

- You can overcome the noise problem by broadcasting your own signal, at a particular wavelength, and listening for its return. Active sensors—send and receive sensors—are very popular in robotics, such as Sparki's ultrasonic range finder.

- Another very common sensor in robotics that works in the same way as an ultrasonic range finder is an infrared (IR) proximity detector. Infrared refers to a place on the electromagnetic spectrum. The electromagnetic spectrum includes what we call visible light—visible to humans, that is. Just outside that range, as measured by the wavelength of the light, is the infrared "light." Night-vision scopes use infrared to "see" in the dark.

- An IR proximity detector has an IR emitter and an IR receiver. Similar to an ultrasonic range finder, an IR proximity detector sends out a signal and then waits for the echo. With light, we call the echo the reflection.

Communication

- One way to think about active sensors working is that they are communicating with the world. They send out a signal and wait for its return. The only difference in practice is that for what we normally call communication, the return signal comes from another person.

- We normally think about communication using radio waves, including ultrahigh-frequency radio waves such as Wi-Fi and Bluetooth. But any part of the electromagnetic spectrum can be used, not just radio waves.

- Microwaves have a smaller, tighter wavelength that is good for sensors to detect motion, such as motion detectors on a house or so-called "radar" guns used by police.

- Infrared is good for even shorter distances—for example, inside a single room—if you have a clear line of sight. For a traditional TV remote to work, you have to aim it right at the TV. That LED light you see is also an infrared LED bulb.

- Visible light has even shorter wavelengths, making it great for communication using tightly focused laser beams but also for wide-beam communication in swarms, too. Visible light can convey information in more complex ways, too. For example, consider the spatial and temporal patterns of light recorded by a camera.

- This quickly gets into a whole field in computer science and artificial intelligence: machine vision. Cameras are a great example of a passive sensor that provides a large amount of continuous information. But the fact that there is a whole field of machine vision should tell you that this gets complicated very quickly.

Important Terms

accelerometer: A sensor that converts changes in velocity into electric signals that can be read by the controller.

echolocation: Active sensing in which an animal or robot broadcasts a sound, senses the returned echo, and uses the difference between the signal and its echo to calculate the distance and composition of the target. See **sonar**.

Hall effect sensor: A sensor that converts changes in an external magnetic field into changes in a voltage or other electric output.

light-emitting diode (LED): Semiconductor-based light source.

multimeter: An instrument used to measure electric voltage, resistance, continuity, and current.

photoresistor: Type of sensor that converts changes in light intensity into changes in electric resistance.

ping sensor: An ultrasonic range finder.

Suggested Reading

Jones, Flynn, and Seiger, *Mobile Robots*, chap. 5.

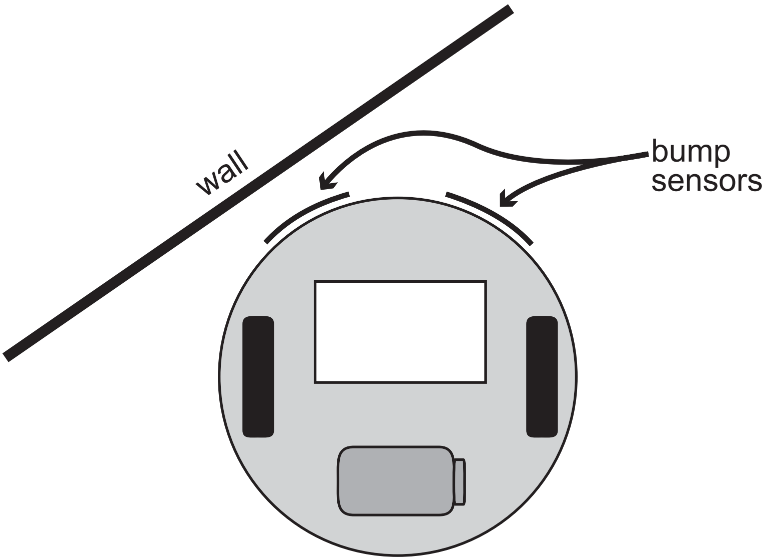
Karvinen and Karvinen, *Make*, chaps. 1–2.

Other Resources

Time to get your own Arduino starter kit: http://store.arduino.cc/category/2.

Questions to Consider

1. The following robot has a bump sensor on its right and left front corners. Only the left sensor is activated when the robot hits the wall.



Which of the following statements best describes how a small Roomba-like robot might react?

IF the left bump sensor is turned on, THEN _____, ELSE _____.

a. then keep moving forward, else reverse direction and turn right.

b. then reverse direction and turn right, else keep moving forward.

c. then reverse direction and turn left, else keep moving forward.

2. Each sensor asks a question. The same sensor can ask different questions, depending on how it is situated on the body, what other sensors it's working with, and what task the robot is trying to accomplish.

This is a Tadro with two light-dependent sensors. We use those two sensors to ask: What is the direction of the light relative to Tadro? This is also called the bearing of the light. How do the two sensors work together to give an answer about the bearing of the light?

Robot Controllers and Programming

Lecture 5

A controller is a special kind of computer whose program modules give autonomy to a robot. The biggest questions in robotics often revolve around what combination of autonomous self-control and remote control by humans is most appropriate for a given situation or robot. Both in the design and operation of a robot, the controller is how any robot's autonomy is created and defined—in the linking of sensors with actuators. The controller is where a robot's autonomy is overruled by humans or extended even further.

Controllers

- Self-control is what turns a remotely controlled machine into an autonomous robot. Where does that self-control come from? Sensors collect information that is coordinated and used to tell the actuators what to do next. These sensor-guided movements depend on a controller, the part of the robot that makes sense of the sensory information and then decides how the robot is going to react. A robot controller or microcontroller gives control, or autonomy, to the robot.

- In the vast majority of modern robots, the controller is a computer that is specially designed to take as input sensor data and communications from a human operator. The controller is specially designed to produce as output signals for actuators and communications for a human operator. Because the controller runs computer code, it is programmable.

- To understand the controller, and how it works with sensors and actuators, we can build a robot using the simplest sensor-actuator circuit. Tadro is a robot with only one sensor and only one actuator. And we can use a class of microcontrollers based on Arduino.

- Let's start with the basic model, the Arduino Uno. You can buy one for about 50 dollars. Arduino uses free software that is available at Arduino.com. Both the Arduino hardware and the software are open-source systems, which means that you can copy, modify, and share your own software and hardware versions for free and without violating any patents or trade secrets.

- A microcontroller like Arduino is more than just a microprocessor. A microcontroller is like an entire computer on a single chip, because it also includes inputs, outputs, a read-only memory, and buffers.

- We can get our sensor and actuator working together by way of the Arduino. The sensor is a light-dependent photoresistor, a very simple sensor that changes its resistance to electric current depending on how much light is falling on it.

- In addition to the input ports, where you connect sensors, Arduino also has output ports. We're interested in the ports through which we can power and control a servomotor.

- Once we have those plugged in, we need the code to run it. We're going to use a program called Knob that can be found on the Arduino website. We connect the photoresistor to the 5V power port and the A0 analog input port. Then, we connect the servomotor to the 5V power port, the ground, and digital port 9. The servomotor's shaft rotates.

- We have compiled and loaded the KNOB program onto the Arduino. The LED lights flash to let us know that we have communication. If the position of the servomotor changes as we shine a light on the photoresistor, then we have a connection from the sensor to the actuator. As the world changes, the sensor sends a signal to the microcontroller that is then converted to a signal for the actuator.

- How can we use this to build an autonomous robot with one sensor and one actuator? The actuator behavior we want is for the Tadro to always be cruising around, flapping its tail at a constant frequency. Where the sensor comes in is that the tail turns, while it keeps flapping, with the direction and amount of turning proportional to the light on the photoresistor.

- How do we get a behavior like flapping? We can go back to the Arduino website. There is a sample program called Sweep, by BARRAGAN studios, that moves the servomotor back and forth. Once we load this program, the servomotor will just swing back and forth without paying attention to the sensor, because it is open loop.

- Next, we need to combine the Sweep program for flapping with the Knob program for turning. The trick is to add the servomotor position from the Knob program—think of this as the turning angle—to the next flapping position that the Sweep program wants to go to.

- To do this, we compile and download the program and observe what happens. The servomotor is flapping. When we shine a light on the photoresistor, the servomotor adds the offset, a turning angle, to its flapping. Now we have a sensor-actuator circuit that will allow Tadro to swim and to react to light.

- For the body, we're going to put the Arduino into a storage bowl and use duct tape to secure the photoresistor on the edge. The servomotor needs something to flap, so we're going to use a pencil as the motor shaft, tape it to the servomotor, and then create a tail with duct tape that will work underwater. We're going to tape the servomotor to the bowl. We've just built a robot from scratch.

- As we move the light, Tadro slowly spirals its way toward the light. We've built a robot with light-seeking behavior. This shows just how simple self-control can be.

- We can use this simple sensor-actuator circuit to point us in the direction of bigger, better, and more complex kinds of self-controlled robotic behavior. When we think about the purposeful movement of the whole Tadro, we can now put the sensory-actuator circuit into its place in the functional loop of an autonomous robot.

- The movement of the Tadro changes the amount of light that falls on the sensor, and the whole functional loop starts all over. Sensor reading changes actuator movement; actuator movement changes sensor reading. Because the conversion processes that take place on board the robot give the robot self-control, it makes sense to call this part of the system the controller.

Shakey

- One of the first robots that had a digital computer for a controller was Shakey, which was a mobile robot built by researchers led by Charles Rosen at the Stanford Research Institute from 1966 to 1972. At the time, digital computers were large and heavy. The final version of Shakey used a Digital Equipment Corporation PDP-10/PDP-15 mainframe.

- The PDP-10 looms large in hacker lore because it helped get the Internet to take baby steps as part of the Internet's precursor, ARPANET. It was very advanced for the time and was used at only a few universities. But the PDP-10 could never fit onto a robot the way an Arduino can now. So, to serve as the controller of Shakey, the PDP-10 had to be in constant radio communication in order to be in Shakey's sensory-actuator loop.

- While Shakey was revolutionary in putting artificial intelligence into a sensor-actuator circuit, the idea of using radio waves to communicate with a mobile vehicle was not. The concept of remote control was patented in 1898 by Nikola Tesla. In his patent, Telsa describes what he called a "tele-automaton," what today we call a radio-controlled or remote-controlled vehicle, and he demonstrated his idea at Madison Square Garden with a four-foot boat that he controlled using a telegraph signal box.

- Tesla did not discover radio waves or invent the devices to produce them. But he did build actuators that could be controlled remotely. Propulsion and steering were his targets, and he demonstrated a remotely controlled boat to the public.

- What we see in many different kinds of robots, by contrast, is neither Shakey nor the Tesla boat, but something we call partial autonomy and partial remote control. The robot has some autonomy, and a human operating remotely shares control with the robot.

- Sparki has a remote control that works a lot like a TV remote controller, sending a few simple signals to its onboard computer. We can signal Sparki to move forward, turn right, or turn left.

- Roomba, too, has a remote control that works in the same way, with an infrared signal from the controller to the same sensor on Roomba that receives the signal from the docking station.

- So, you can also see why old-fashioned remote control is still with our modern robots even though they have the capacity to control themselves. By allowing humans to insert themselves into the functional loop, we can take control, as needed. This involves communication between robot and human.

Behavior-Based Robotics

- We can treat sensor-motor circuits as discrete building blocks, where each block is a behavior module. These modules can function independently. They may function together, their coordination decided upon by the controller.

- This focus on behavior modules driving thanks to sensor-actuator circuits was developed and made famous by Rodney Brooks when he was a professor at MIT. This approach goes by the name of behavior-based robotics.

- The hallmark of behavior-based robotics is that the controller makes decisions about which behavior module gets to operate when. Because it's an all-or-none proposition, we think about this as the behavior modules competing to have the controller decide to let it operate, a winner-takes-all scenario. There is no sharing of the actuators; you either are the behavior that is operating or you are not.

- Rodney Brooks very quickly realized that the great value of behavior-based robotics is that you can use very simple programming, such as what we did with the Arduino, to get robust self-control out of your robot.

- Roomba is one of Brooks's creations. Roomba has several sensors, such as cliff detectors to signal when Roomba is going over the edge of the stairs. Another infrared sensor on top of Roomba is an infrared detector that is paired with the infrared emitter on the top of the recharging station. Together, they give Roomba the ability to find its way back home, to sleep and to recharge.

- What happens when we put them together? How do the modules interact? What decisions is the controller making? We can do pair-wise comparisons to see which behavior is programmed to win in which situation. For example, cliff detection and avoidance overrule docking. We are always comparing our current perception of the state of things with our desired state of things, our goal: Are we there yet? If not, then keep going. If so, then stop.

- This comparison of your desired state to your actual or goal state is at the heart of self-control. The general principle is what's called **negative feedback control**—one of the most important principles that governs the control and regulation of mechanical or organic systems. In fact, the fields of **cybernetics** and engineering control theory are built around issues of feedback control.

Activity

Learn to read the code for the one-eyed Tadro, which you'll find here: http://pages.vassar.edu/darwinsdevices/code/.

The following are a few tips to help you read this code.

// indicates a comment line that won't be read by the computer. You can leave your self-important notes this way.

Setup() is a function that is read once, when the program first starts running. Variables are initialized, and libraries containing important functional routines for hardware are called. The word "void" in front of setup() indicates its data type; void means that this function returns nothing. Contrast that with the variables, such as "int pos" that use integers ("int") for the variable "pos."

Loop() is a function that every Arduino program must have. The instructions inside are read over and over until you turn the Arduino off.

Curly braces { } are always found in facing pairs, and they have different uses. They are used to circumscribe the code used by a function. They indicate the code that belongs to a loop or a conditional statement. It's important to note that functions, loops, and conditionals can be nested and that nesting is organized with the braces.

The #include is an instruction to use a library outside of the program—in this case, the control language for a servomotor.

The semicolon (;) must be included at the end of every line, with the exception of the lines for setup() and loop(). Forgetting a semicolon is an easy way to have your program not compile. Compiling is the process that converts the words and symbols you've typed into the machine code that the computer actually uses for instructions.

A variety of mathematical operators can be used: +, −, * (multiplication), / (division). Comparisons can be made with the following operators: == (equal to), < (less than), > (greater than), <= (less than or equal to), and >= (greater than or equal to).

You can find additional information about the language of Sketch at the Arduino site (http://arduino.cc/en/reference/homePage).

Important Terms

cybernetics: The study of how dynamic systems are regulated, focusing on issues of feedback, control, and communication.

negative feedback control: Regulatory process to maintain stable output of the robot using the difference between the desired output and the actual output.

Suggested Reading

Banzi, *Getting Started with Arduino*, chaps. 1–4.

Monk, *Programming Arduino*, chaps. 1–3.

Other Resources

Arduino Development Environment

With an Arduino starter kit in hand (http://store.arduino.cc/category/2), get software that you'll need to make your Arduino microcontroller work here: http://arduino.cc/en/Main/Software.

RobotC

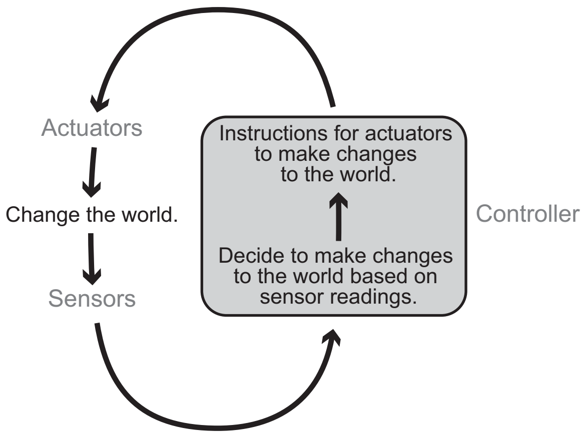
Many programming languages for robots are variations on the language called C. RobotC (free trial) is one, and it can be downloaded at http://www.robotc.net/, which includes many helpful tutorials.

PartSim

You can design and then simulate a robot circuit—before you try to build it—using one of many simulators available on the Internet. PartSim (http://www.partsim.com/) is free and easy to use.

Questions to Consider

1. Here is one way to think about the functional loop of a behaviorally autonomous robot.



If you wanted your robot to work even if the sensors didn't detect any changes to the world, what kind of functionality would you need to add to the program operating the microcontroller?

IF you detect no changes in sensor readings, THEN ____.

a. keep giving the same instructions to the actuators.

b. halt (and/or turn off actuators) and wait for the sensors to detect a change.

c. go into a random search pattern.

2. When we write an algorithm for a computer in English, this is called pseudocode. While pseudocode can't be used directly to program the microcontroller, it is great way to start designing the real code. Using pseudocode, how would you create an algorithm that lets a Roomba with two bump sensors and two infrared (IR) detectors navigate around the house?

Human-Inspired Robot Planning
Lecture 6

Shakey was a mobile robot built by researchers at the Stanford Research Institute from 1966 to 1972. This is the robot that *Life* magazine in 1970 heralded as "the first electronic person." Although Shakey didn't look like a human, it had the ability to reason and plan with what was considered a brain-like machine—a digital electronic computer. In the controlled but real-world conditions of a laboratory, Shakey could successfully navigate, move through the world from one place to another, avoid objects, and, most impressively, adjust to changes in the world that it hadn't known about. Shakey could learn and update its plans.

Navigation

- Key to Shakey's thoughtful, deliberative navigation is a computer-reasoning construct for the robotic controller that we call an internal world model. Model-based architectures are inspired by humans and our ability to visualize, plan, and simulate our actions before we undertake them. Some of the most important actions that we and our robots take are the goal-directed movements that we call navigation. If you use a world model to navigate, we call that model a map.

- In robotics, navigation is the most important problem that you must get your mobile robots to solve. How do you get safely and efficiently from point A to point B? Navigation is a fundamental example of movement with a purpose, and purposeful movement is at the heart of self-control and autonomy in mobile robots.

- There are different solutions to navigating depending on the workplace and task of your robot. Shakey shows us that maps can be useful indoors, and modern robots working indoors often use maps, too—maps of floor plans, for example. Maps are also extremely helpful for sustained and long-distance navigation outside.


© Igor Mojzes/iStock/Thinkstock.

Global positioning systems have made the work of navigation quite easy for humans.

- Over the past 2,000 years, we humans have developed a whole set of formal rules for navigation. Navigation is not a trivial problem, even though we take it for granted, thanks to the computers and **global positioning systems** (GPSs) that do the heavy lifting these days.

Homing

- Navigation begins with needing to know where you are, relative to some reference position or landmark, so that you can set a course to go where you want to go. In robotics, you can have your robot know where it is without using a map, provided that the workplace is structured, predictable, and confined. You can set up a beacon so that the robot can always return home. This process is called **homing**.

- Homing is the ability to go out and get back to where you started. Once you are back to where you started, you know where you are. For robots, homing is important for returning from a mission or recharging batteries.

- For Roomba, homing is called docking, but it's the same action: returning home to where it started. Roomba stops cleaning and starts searching for the infrared signal from the dock. Once it finds it, its behavior changes; it heads straight in, slowly, to the dock.

- Roomba is able to navigate without a map because it can sense the signal from the infrared beacon on the dock. Using a beacon is an example of what we call landmark navigation. A beacon is one type of landmark that actively emits a signal. The classic navigational beacon is a lighthouse.

- For Roomba, operating without an internal world model, all we need its computer code to do is to read the intensity of the beacon's infrared signal with its infrared sensor and make adjustments to minimize the difference between the maximum value measured and the values just measured. This is an example of negative feedback control, where you minimize the error between a desired and actual state in order to regulate your movements.

- We can use negative feedback control to make an algorithm to allow Roomba to navigate to its homing beacon with just a single infrared sensor. We measure the strength of the infrared signal to the homing beacon at a particular moment in time. Then, the robot moves forward and to the right, for example, and we take another measurement. If we take the difference between the maximum value we've measured so far and the current value, the difference is called an error term in control theory.

- But with only a single sensor, we don't yet know what that error tells us. Is the robot going too far to the right or too far to the left? We can see that the robot is actually turning away from the beacon to the right, for example. But the robot doesn't know that; it needs more information.

- One way that robots, including Shakey, get more information is to move and see, or sense, what happens. This is called active perception.

- The robot moves to the right again, and we measure the infrared value again, along with the difference with the maximum value. We can compare the two errors. If the most recent error is greater than the earlier error, the conclusion is that the error is increasing. And because we kept track of how the robot was turning, we know that it was turning to the right and that movement increased the error. So, if the goal of negative feedback is to decrease the error, then the robot needs to turn in the opposite direction.

- The robot turns to the left now as it moves forward. Comparing the most recent error to the previous error, we find that by turning to the left, the robot is decreasing the error. This means that the robot is heading back toward the beacon—it's navigating.

- We've just created an algorithm, or a set of instructions, for navigating toward a beacon using just a single sensor. One of the important tricks of this simple homing algorithm is that we had to know something about how the robot was moving. In this case, we were keeping track of which direction the robot was turning.

Holding a Compass Heading

- The algorithm for homing is very similar to an algorithm for holding a compass heading. Obviously, a compass is used as the navigational sensor. Like homing or docking, holding a compass heading is a very basic behavior in robot navigation.

- A digital compass is a common sensor to put on a robot. The Arduino robot has one built in. It is a Honeywell model HMC 6352, a two-axis magnetoresistive sensor. These sensors change their electric resistance in response to changes in the magnetic field.

- One of the great advantages of a compass over a system like an infrared beacon is that it works at any distance and in any condition—in snow, in rain, or with obscured vision. The one great disadvantage of a compass is that it is sensitive to local magnetic sources, so you need to keep that in mind if you get strange results or have to operate around a lot of machinery.

- In the computer code for Arduino robot, we've programmed the robot to move along a course set by its internal compass. We calculate the error term as the difference between the desired and the actual compass heading. When we turn on Arduino robot, the first thing it does is to turn and then head out straight, along the course dictated by our choice of a compass heading. It makes small turning corrections based on the errors.

- Holding a compass heading is perhaps the most basic behavior of navigation. If a robot can't move steadily on a given compass heading, then it will not be very efficient when it comes to navigation using a map.

- One of the great advantages of using a compass is that you don't need to use a beacon or a landmark. But holding your compass heading only gets you so far. With a compass alone, you don't know where you are, but only in what direction you are headed. Even if you don't know where you are, a compass can be useful to know where you just were, which is how we can build up to homing from holding a compass heading.

Using Multiple Sensors

- For navigation with a compass, in addition to knowing your direction, to know where you are on a map as you travel, you need something to help you figure out the distance you've traveled from a known starting position. If you are on foot, then you can count your steps. If you know the length of each step, then you can multiply the number of steps and the step length to find the total distance you've traveled.

- You can solve this problem on a wheeled robot by counting the number of rotations of the wheels. Electronic sensors can count rotations, such as a sensor called a **shaft encoder**, which is often a magnet or an infrared emitter/detector that counts high-contrast lines on the wheel. With a **stepper motor**, you simply tell the motors how much to rotate.

- If you know the size of your wheel and the rate at which it is rotating, you can build a speedometer. If you have a clock to keep track of how long you've been going in that direction and at that speed, then you can calculate the distance you've traveled, your current position relative to where you started, and how to get back home. The distance traveled is the product of your speed and the time you've been moving at that speed. With the distance and direction you've traveled from your starting point, you know where you are relative to where you were.

- What's important in navigation is that data from multiple sensors—such as compasses, odometers, speedometers, and clocks—are involved. And coordination of multiple sensors sounds like a job for a robotic controller.

- In practice, going straight out and then straight back is rarely very useful. We often want our robots to explore or to carry passengers for hours along a course on the road or the sea that may involve avoiding land, bodies of water, or known obstacles.

- A kind of navigation called **dead reckoning** involves knowing where you are and then heading out and moving along your intended course, making turns after you've traveled a given distance and heading as calculated by your compass, clock, and speedometer.

- One way to know where you are is to have your robot use GPS. One of the exciting breakthroughs in consumer robotics is that robots are now fully capable of navigating for themselves outside with GPS, which is a fleet of navigational beacons, mounted on satellites in orbit, that provide us and our robots with what we call a **navigational fix**.

- A navigational fix refers to fixing or locating your position on a map. To do this with GPS, the robot needs a GPS receiver that reads the signal from up to five GPS satellites. The satellites know exactly where they are, even though their position is changing as they orbit. By knowing where the satellites are and the distance to them, the robot can triangulate its position. This is done with a GPS module, and some trigonometry, which is done by a tiny computer on the GPS module. The GPS module is basically a fancy sensor that gives the robot its position in latitude and longitude.

- With GPS, we can get around the problems of dead reckoning by using waypoints. We can then tell the robot to go to the waypoint. If the robot senses that it's not in the right position, it can make adjustments. Those adjustments are done by negative feedback control.

Shakey: A Pioneer

- Shakey was able to sense, model, plan, and act. In terms of a sensor-actuator system, the information from the sensors was sent to the computer controller. The controller updated the model and then created a new plan for how to best achieve its navigational goals. Then, it sent the plan back to the robot, which moved the actuators accordingly.

- In spite of the fun some modern roboticists poke at Shakey, the robot was ahead of its time. Shakey was the first robot to use a powerful digital computer for its robotic controller. Because of that computing power, Shakey was also the first to use what we now think of as the model-based controller architecture. The use of rules of reasoning and deliberation in the planning routines of the controller link Shakey to the early days of the field John McCarthy called artificial intelligence.

Important Terms

dead reckoning: A method of navigation in which current position is calculated based on speed, heading, and time from last known location.

global positioning system (GPS): Navigational system using satellites that locate a receiver in coordinates of the world geodetic system.

homing: Navigating to a previous location, usually the location where a robot started.

navigational fix: Location found using external reference points.

shaft encoder: A sensor that converts the rotations of a wheel into an electric signal that represents the speed of the wheel's rotation.

stepper motor: A brushless DC motor that can rotate precisely without the need for sensory feedback.

Suggested Reading

Cook, *Mobile Robots*, chap. 4.

Murphy, *Introduction to AI Robotics*, chap. 9.

Other Resources

Shakey is incredibly important in the history of robotics. Learn more about it at the Stanford Research Institute: http://www.ai.sri.com/shakey/.

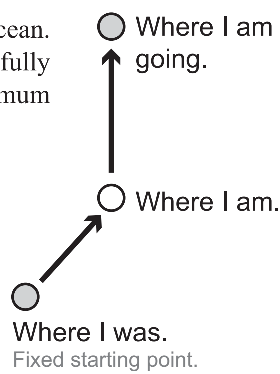
Questions to Consider

1. Imagine that you have a robotic ship on the ocean. If you want your robot to navigate successfully to the next waypoint, what is the minimum information it would need?

a. Map coordinates of the next waypoint.

b. A compass heading to the next waypoint.

c. A compass heading to the waypoint and distance to the waypoint.



2. Imagine that you have a robot that works in an office building, picking up and delivering mail. If you want a robot to be able to update its world model quickly, what's the most efficient method?

   a. Monitor all objects and people at all times.

   b. Monitor a subset of objects and people with which and whom you are likely to interact.

   c. Wait until you run into something that's not on your map and then add it to your map.

3. Model-based controller architectures focus on planning. In contrast, behavior-based architectures focus on reacting. What are the benefits and costs of these different approaches?

Animal-Inspired Robot Behavior
Lecture 7

In principle, building an internal world model is a great idea for a robot, if it has the sensors and the computing power. It senses, plans, and then acts. That's how we thoughtful creatures work, but the trick is that we have to constantly build our model. Humans have to create and update our map while we are operating in the world; we have to rapidly, constantly, and accurately update our internal world model. We have to make and remake our map, continuously, as we learn about the constantly changing world we're in. If we don't, we could be in trouble.

Removing Planning from the Approach

- In the early 1970s, robot Shakey's answer to this problem was twofold. First, Shakey moved short distances, stopped, and then used its sensors to look for changes in the world. Second, if Shakey found objects that weren't on its map, it would update the map. Then, and only then, would Shakey calculate a new navigational plan. This took time, and, looking back, it's easy to criticize Shakey for being slow. But this was 1970, before small, fast personal computers were invented.

- How do we make Shakey faster, smarter, with more intelligent behavior for responding quickly to changes in the world? Most researchers in artificial intelligence (AI) thought they knew the answer: Shakey needed a better brain, a faster computer with more memory. That way, Shakey could update its world model faster, make it more accurate, and map and remap moving objects as it kept track of its own movements. They were right, in part. Faster computers with more memory do help.

- But in the 1980s, this make-the-brain-smarter and use-maps approach had largely failed, and not just for the reason that the computers weren't big enough and fast enough. The fields of AI and AI robotics were stuck in a methodological rut. Mobile map-making robots were still very slow to move and worked only under conditions where the world was structured—that is, a world that is predictably not changing much. Throw most real-world situations at these robots and they simply couldn't move or accomplish their tasks.

- One AI researcher, Rodney Brooks, a professor at the Massachusetts Institute of Technology, began to question the whole sense-plan-act approach of model-based robotics. Instead of thinking about how humans work, Brooks was inspired by the intelligent behavior of nonhuman animals. He watched insects move, fly, navigate, sting, and pollinate.

- He observed that insects behaved intelligently—doing things no robot at the time could do—without much of a brain at all. This got Brooks thinking that part of the problem with robots like Shakey was that humans were trying to build them in their own image, with humanlike intelligence.

- Brooks figured that by focusing on human intelligence—abstract reasoning tasks, such as playing chess, solving math problems, and building internal world models—we had started the field of mobile robotics backward, evolutionarily speaking. Why not work the way that evolution had? Start with the simpler animals and then build up from there.

- Pushing that idea to its limits, Brooks wondered if you could still get intelligent behavior if you removed planning altogether from the sense-plan-act architecture of the robotic controller. Don't create an internal world model but, instead, rely on sensing and acting without planning. And because the planner is the brain, just remove it altogether.

The Biology of Brains and Intelligence

- How do animals that don't have much of a brain, who don't do much planning, manage to live, behave, and be successful? We humans are animals with a brain that seems large in proportion to our body size. A ratio calculated across all animals called the encephalization quotient suggests that our brains are more than four times larger than would be expected from body size alone.

- However, our big-brain pride tends to blind us when we consider the cognitive workings of nonhuman animals and their implications for robots. Most people think that no other animal or robot is as smart, as intelligent, or as adaptable as a human. Depending on the situation, they might be right.

- But intelligence depends on the context, the environment in which your species evolved. Judged by mass, rather than size, small animals such as birds, mice, and even ants have brains that are at least as, or more, impressive than those of humans. Sensors and actuators across the animal kingdom are even more impressive.

- The implication that matters most for building robots is that every species has its own special set of sensors (and actuators) that are adapted for working at a particular job in a particular environment. For example, female mosquitoes are olfactory and thermal

Sensor-guided movement is apparent in every species and is central to intelligent behavior.

predators, using an array of chemical sensors on their antennae to fly up the streams of carbon dioxide that we mammals make as we live and breathe.

- When thinking about insects, Rodney Brooks realized the following important lesson from biology: Sensor-guided movement is central to intelligent behavior. Movement without sensors may be cool, but it doesn't let you navigate. You can't move toward any particular goal. You can't navigate or get prespecified tasks done.

Sensor-Guided Movement

- Whether or not you think a mosquito is intelligent, you have to appreciate their pesky hunting abilities. Mosquitoes can track down mammals with a simple system of sensors connected to a simple nervous system that creates a continuous turning signal for the wings. This is sensor-guided movement.

- Mosquitoes are incredibly successful survivors, and they win the game of life without an internal map. They don't need one. All they need to navigate to the food is sensor-guided movement.

- In the language of artificial intelligence, mosquitoes behave intelligently because they don't waste processing power making maps that they don't need. Instead, simple reflex-like connections between sensors and motors get the job done.

- Instead of sense-plan-act, mosquitoes sense and act—without planning. They behave without a brain. These were exactly the kinds of bioinspired insights that Rodney Brooks was after when he was thinking about the slow progress of the sense-plan-act approach of classical AI robotics.

- If we wanted intelligent machines that could actually do something in the real world, Brooks reasoned that we needed to stop focusing on the computationally intensive problem of building an internal world map. Instead, we needed to think about what happens in the world. Behavior happens. Animals move around and get stuff done.

- So, how could robots behave intelligently without planning and without much of a brain? Once you understand the intelligent behavior of insects, you can use the simple sense-and-act rule to program a robot. And that was Brook's flash of brilliance.

- But a key practical question remained: Will those simple insect-inspired, reflex-like rules really work on a robot? To find out, Brooks and his graduate students at MIT built a six-legged, insect-like robot named Genghis. Like a mosquito, Genghis had onboard sensors to detect mammals. But while mosquitoes sensed carbon dioxide, Brooks focused on another insect sense: heat detection.

- Brooks and his team mounted six pyroelectric heat-detecting sensors on the front of Genghis. Genghis used these sensors and a simple rule to follow warm human mammals around the lab. And it worked beautifully. Finally, in the form of the mobile autonomous Genghis, here was a mobile robot that behaved intelligently. Genghis could move around in the world and do something quite impressive: track and find humans.

- No other robot from the AI community had come close to this kind of ability, this kind of intelligent behavior. And, best of all, Genghis could track mammals in a dynamically changing world, without a map—no planning needed, and no brain required.

- Brooks moved robotics from the classic sense-plan-act paradigm, what we call model-based robotics, to what is now called behavior-based robotics. In 1988, when Genghis was built, behavior-based robotics was a breakthrough in the world of robotics, and Brooks became internationally famous. Brooks and his students went on to use his behavior-based robotics to create the first consumer robot to have widespread success: Roomba.

- Compared to model-based systems, behavior-based systems are simple and robust. They require a lot less computer processing. And simple design plus less computer processing, in turn, means that your batteries last longer. More importantly, Roomba doesn't need a map to work. Like an insect, you put Roomba down, and it gets to work.

- What are the advantages of a behavior-based robotic architecture? Because it doesn't have to plan, a behavior-based robot can react quickly. Because the software that runs the computer on the robot doesn't have to be as complex, it means that the whole robot is less likely to break down. And the modularity of sense-act reflexes means that you can use them like building blocks to easily create complex behaviors.

- But best of all, the sensor-guided movements of behavior-based robotics has gotten engineers to focus on building better sensors. Because behavior-based robots work, they are now a fascinating part of contemporary life, thanks to pioneers like Rodney Brooks.

Activity

Vassar's Robot Ethology Challenge: http://www.mind.ilstu.edu/curriculum/virtual_robotics_lab2/video.php.

By visiting this website, you can select the priorities of different behaviors and see how your programming impacts the overall behavior of a small wheeled robot.

Suggested Reading

Brooks, *Flesh and Machines*.

Dawson, Dupuis, and Wilson, *From Bricks to Brains*, chaps. 6–7.

Walter, "An Imitation of Life."

Other Resources

Fast, Cheap & Out of Control (1997). A documentary by filmmaker Errol Morris about four unrelated individuals in unusual fields having something to do with animals, one of whom happens to be MIT roboticist Rodney Brooks.

Basic Skills for Making Robots

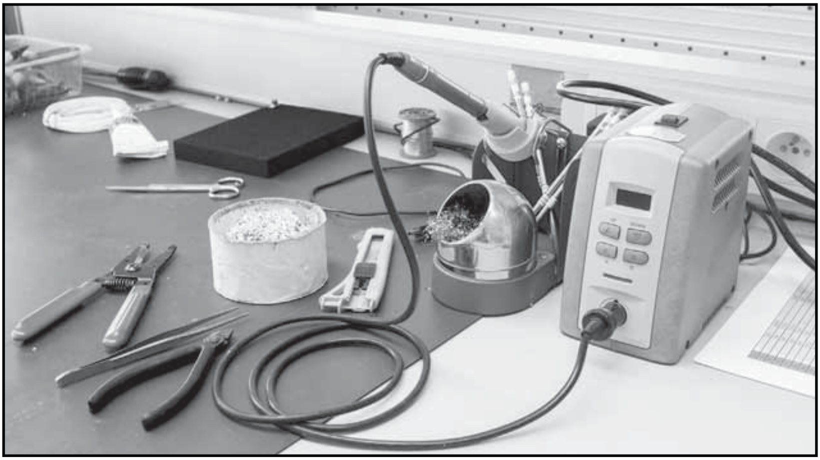
Lecture 8

In order to make robots, there are some basic, preliminary skills that you need to know. The ones you will learn are very basic, but the point is that it takes very little to get started in the world of making robots. Soon enough, you will know what you need to do in order to make a circuit, hack a robot, and even make a toy into a robot. No matter what tools you have in your toolkit, always make sure to start with some kind of safety protection, such as safety glasses.

Supplies of the Maker

The following is a list of tools of the trade that you'll need if you're going to be a maker of robots.

- Wire cutters
- Miniature screwdrivers
- Convertible screwdriver
- Pliers
- Small scissors
- Soldering iron and stand
- Multimeter
- Solderless breadboard
- Hot glue gun
- Solid-core wire: red, black, and yellow
- Stranded wire: red, black, and yellow
- Lead-free **solder** with rosin core
- Black electric tape
- Twist ties
- Permanent marker
- AA and 9V batteries



© Artush/iStock/Thinkstock.

Make sure that you have all of the necessary tools handy before you begin your project.

Skills of the Maker

- Some of the core skills you need to know in order to build robots include stripping a wire, twisting two wires together, and soldering two wires together. When it comes to soldering, safety is important, because you can burn yourself. Some of the safety practices that should be followed include putting on safety glasses and working in a ventilated area.

- When soldering two wires together, first make sure that the soldering iron is fully heated. Then, clean the tip of the hot soldering iron on a damp sponge. The next step is tinning the tip, which involves melting a small amount of metal and putting it on the tip of the soldering iron to prepare it for use.

- The counterintuitive thing about soldering is that you don't actually want to melt the solder directly. Instead, you want to heat up the wires and let the wires melt the solder. To do this, touch the soldering iron to the wire (not the solder). Then, touch the solder to the hot wire near the iron. Let the solder flow into the wire.

- Another skill that is good to have when building robots is how to test for continuity with a multimeter. Multimeters are very inexpensive; you can find them at hardware stores. You need to make sure that your multimeter comes with cables. Specifically, there should be a red and a black test cable, or test lead. Multimeters measure direct current voltage and alternating current voltage. When we build electronics, we're working in the direct current world.

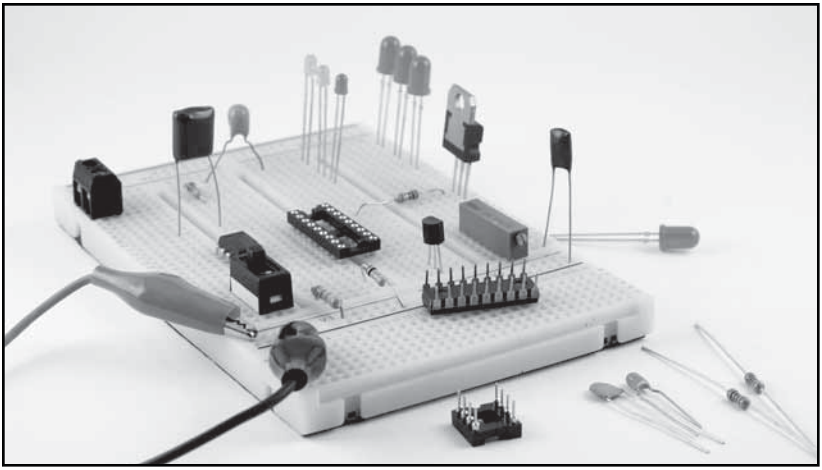
© robeo/iStock/Thinkstock.

Ventilation and eye protection are important precautions to take when using a soldering iron.

- A speaker on the multimeter measures what's called continuity. We want to know if we have a good connection, or continuity, when we wire something together. If you touch the two leads together after they're plugged in to the multimeter, you should hear a beep, which is the signal that you have continuity.

- There is another skill that is useful when working with a circuit: using a breadboard. A breadboard has outer power strips that are connected internally. By putting your circuit on a breadboard, you don't have to do any wire twisting, and it's a way to lay out your circuit and make sure that it works before you do something, such as solder.

- The core of making a robot is connecting a sensor to an actuator. That means building and testing a circuit. **Voltage dividers** show up everywhere in electronic circuits. They are very handy, and they can be used in a circuit to make a light-sensitive robot. A voltage divider involves twisting two resistors together, as well as input wires and output wires.



© swifty99uk/iStock/Thinkstock.

Solderless breadboards are useful for laying out and testing circuits before committing with solder.

Hacking Robots

- Hacking a robot is where you really make it your own. The hacking might start with a toy, or a robot, or any other gadget. You might be changing software code or adding controller intelligence to a previously dumb machine, or you might be changing the body of the robot. Any and all of these count as hacking.

- Hacking is a design process in the sense that you aren't following any set of directions. Hacking requires no specialized training. To hack, you observe, dissect, tinker, commandeer, recombine, and repurpose. The whole idea is to get something that you have to do something that it was never intended to do.

- The problem with hacking is that most of us were told not to break our toys—but forget about that. Because hacking works by taking something else apart, many people get nervous about doing so. But don't be afraid to tinker and make mistakes. Fail your way to success, as counterintuitive as that seems.

- A tank track is a very common thing that is used in robots. There are kits that you can buy that let you build it. It's meant to be the basis of any kind of robot that you might want to build. You can also start with any sort of toy that just goes straight when you turn it on. Tracks are even able to get up and over obstacles, but they're blind. There are actuators and a body, but there are no sensors attached.

- You can use a circuit that you lay out on a breadboard to make a robot. A transistor is what allows you to hack the tank track to make it into a robot. A transistor can be a fancy kind of switch that sends the data to the voltage that is being divided by the voltage divider in the circuit. It's a kind of gate that allows greater amounts of current to pass through it to power the motor. In fact, with some extra steps, you can make a photosensitive robot.

- One of the great things about hacking a robot, or building one with help from a kit, is that anyone can try it. In fact, the robot hackers' secret is that you can hack any robot—and hackers do. With the rise of kits like LEGO Mindstorms, VEX Robotics Design System, and BIOLOID, it has never been easier for beginners to build sophisticated robots.

- There is a kit from VEX Robotics that lets you build Clawbot, which is a great way to get started on building a sophisticated robot—in this case, one that includes a manipulator and a mobile robot base. Best of all, when you start building robots like this and getting them to work, you're getting a shortcut to creating something lifelike and, just as interesting, autonomous robot behavior.

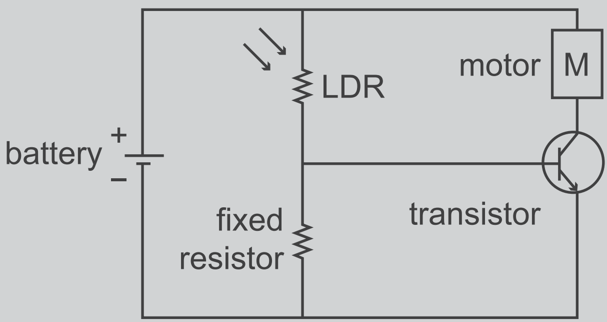
- There's also a Hacking Roomba website to get you started. It is full of resources, including a link to Roomba's company, iRobot. Moreover, the company not only supports this, but in 2007, they also created a Roomba that they want you to hack, called Create, so that you don't have to worry about wrecking your Roomba.

- Instead of a vacuum cleaner, the Create robot has a cargo bay where iRobot has added a communications device that allows you to easily hook in a variety of sensors. iRobot has also made available software, called Open Interface Commands, to make more readily available a number of the behavior modules that Roomba uses. One use of Create has been to mount a camera to perform surveillance.

Activities

Hack a toy!

We've seen a simple hack that involves taking a motorized vehicle and adding a voltage divider with a light-dependent resistor (LDR) as the main sensor. The following circuit diagram shows what you need to make this work: the LDR, a fixed resistor (this will vary depending on your LDR, but start with a value of 10 kW), a small motor (DC brush, 1.5 to 3 V rating), and an NPN switching transistor (type 3904). While two AA batteries (3 V total) were used in the lecture, you may find that you need to double the voltage to 6 V. A 9V battery can also be used with success, but you need to be careful not to run the motor for too long or you may burn it out.



How does this circuit work? When more light falls on the LDR, its resistance decreases. As more current flows through that leg of the circuit, it flows into the base of the transistor, allowing more current to flow through the transistor from the motor to the negative side of the battery. The result: As more light hits the LDR, the motor spins faster.

The hardest part of this hack is finding the right kind of toy. You need a battery-operated car or truck with a motor that powers a single drive axel.

Hack the code!

Peter Staten and John Long combined two programs available from ArcBotics (www.arcbotics.com) for running Sparki. Their hack, which was demonstrated in Lecture 1, combines edge avoidance and remote control to force you to drive Sparki safely. It is called Peter's Safe Driver Program.

If you have a Sparki, you can copy and enter their code from this site: http://pages.vassar.edu/darwinsdevices/code/.

Important Terms

solder: Metal alloys with low melting points used to fuse together separate elements and, in electronics, create secure electric connections.

voltage divider: A simple circuit that uses two resistors, with resistance Z_1 and Z_2, to reduce the input voltage, V_{in}, to a lower output voltage, V_{out}, by the following equation:

$$V_{\text{out}} = V_{\text{in}} \frac{Z_2}{(Z_1 + Z_2)}$$

Suggested Reading

Monk, *Hacking Electronics*, chaps. 2–3.

Scherz and Monk, *Practical Electronics for Inventors*, chap. 7.

Other Resources

The Sparki robot by ArcBotics is a great open-source robot to get you started. It comes ready to work right out of the box: http://arcbotics.com/products/sparki/.

Question to Consider

1. How do you turn a toy into a robot? Many toy vehicles have a body, actuators, and an energy source. What do we need to add to a toy vehicle to turn it into an autonomous robot?

Designing a New Robot

Lecture 9

How do we move from kits and hacks into designing and building complex robots for commercial or research purposes? First, you need people: Complex robots are built by teams, with different members specializing in different aspects of design, construction, and testing. Second, you need money—at least to pay for gear and supplies, and you may even need to pay some or all of the people on your team. Third, you need a clear design goal and a plan to get you there.

Robot Madeleine

- Robot Madeleine is a complex robot that was built by a team at Vassar College in partnership with Nekton Research LLC. This combined team included a biologist, cognitive scientist, conceptual designer, mechanical engineer, electrical engineer, and systems integrator.

- The group had two goals in building Madeleine. First, they had a scientific goal. They needed a robot that they could use to test ideas about how extinct animals with four flippers—such as the plesiosaurs or kronosauruses of the Mesozoic Era—could swim. Of particular interest was why those giant ancient reptiles would swim with all four flippers but the living species of vertebrates that had gone back into the water, such as sea lions and penguins, swam with only two. To answer this question, they needed to be able to control the flippers independently so that they could vary the gait, or pattern, of flipper use.

- Second, studying flippers also contributed to a performance goal. Nekton was a small company, and they wanted to create an amphibious robot, one that could swim and maneuver in the surf zone and then crawl its way up the beach.

- Goals and a plan are crucial, but roboticists also keep something else in mind: Every engineering project to create something new goes through a series of failures before it succeeds. This process of failing your way to success is sometimes called prototyping and involves trial-by-error learning as you go.

- One way to get trial-and-error learning started is to first build the robot in computer simulation. Coppelia Robotics has created software for doing just that. Called V-REP, which stands for virtual robot experimentation platform, it allows you to configure your robot and then program it in the same computer language that you would use in your physical robot. Then, you run your simulated robot in a simulated world.

- A simulation program like V-REP lets you model complex robotics problems, such as path planning for manipulators and vision sensory simulation. As long as you have a good physics engine for your world and your robot, simulations can shorten development time of the physical robot by letting you try out and compare different designs very quickly.

- It took the team about 18 months to build Robot Madeleine, version 1.0. During this period, team members spent about four months just figuring out how to take their overall goal and turn it into concrete specifications for building Madeleine. They wanted Madeleine 1.0 to be self-propelled, fully submersible, and remote controlled (through a cable coming out of the back).

- There are a huge number of decisions that have to be made. Every sensor, motor, part, size, and shape involves making a decision. And it helps to have on your team someone who is the systems integrator to make sure, for example, that the selected motors are compatible with the chosen batteries. You don't want to have batteries that can't supply the motors with enough power.

- It took another 12 months to make Madeleine truly autonomous, a version called 2.0. Part of the issue was that they were applying a behavior-based controller architecture to an underwater robot. The big challenge is control: The robot might detect an object and stop flapping its flippers, but it keeps gliding toward the object. The team needed to figure out how to move Madeleine slowly enough to avoid damaging the robot but fast enough to make progress. The systems integrator was the chief engineer on these issues.

- Scientifically, Madeleine was a smashing success. The team was able to show that four flippers were great for acceleration from a stop and breaking. But once Madeleine got to top speed, that maximum was no different for swimming with four or two flippers. So, given that it takes more energy to swim with four flippers, you might as well swim with two, if you are a cruiser. But if you are a sit-and-wait predator, use all four.

- Nekton's commercial goal was also a success. With special flippers that Nekton patented, Madeleine could swim in the surf and crawl up the beach. Madeleine ushered in a new era of flippered, amphibious vehicles called transphibians. Nekton's transphibian patent was later purchased by iRobot, the maker of Roomba, becoming the start of a marine robotics division.

RayBot

- A new underwater robot, RayBot, is in the process of being built. The RayBot project is a collaboration between the Interdisciplinary Robotics Research Laboratory at Vassar College and FarCo Technologies Inc. The impetus for them to build this fishlike robot came from the U.S. Navy's Office of Naval Research, which in 2008 created and made public a funding solicitation: "Efficient, Highly Maneuverable Artificial Fish for Stealthy Surveillance."

- To come up with a plan, the team needed to get down to specifications. What kind of fish? For a shape-changing, energy-efficient, silent body, the team considered skates and rays. They knew that fish that can glide would be very energy efficient. An electric ray's body is flattened, and it does not flap its body disk. This is different from skates, stingrays, and manta rays. Electric rays use their body disk to glide, in the most energy-efficient and stealthy form of swimming possible.

- Another benefit of the electric ray body plan is that this type of ray can carry a stable payload. Bony fish have to wiggle and bend their bodies in order to swim. The electric ray's body disk doesn't bend up and down for propulsion; instead, the body disk is stable, so its payload, the electric organ, is, too. The electric organs, which can zap and stun prey, are actually modified muscles. But these muscles don't contract; they just store and release electric charge.

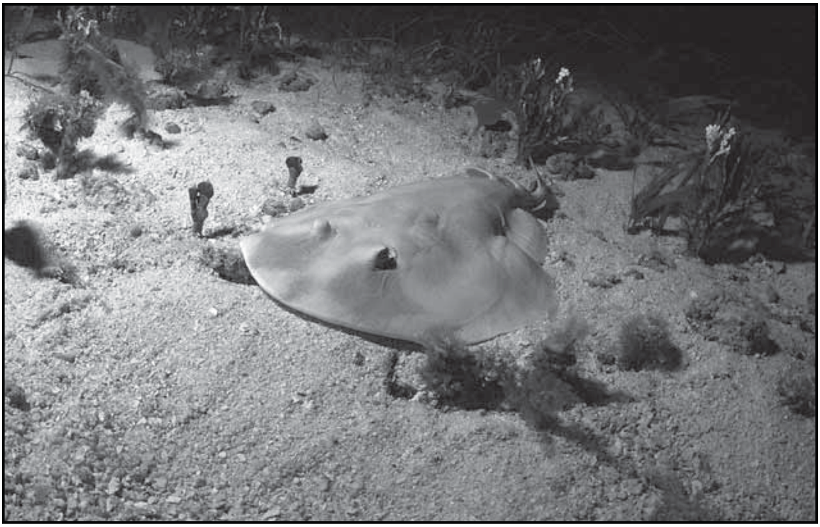
- Electric rays have large elliptical compartments. In a robot built like an electric ray, those elliptical compartments are where those stable payloads can go.

- When the team started building RayBot, they first created an autonomous surface robot modeled on the electric eel. It had a body disk and propulsive tail. The body disk carried a payload that included a controller. The controller ran a fixed-priority set of instructions that operated two behaviors: navigate and avoid objects.

- Navigation is a behavior created by linking the input from the two photoresistors to the steering of the tail. RayBot 2.0 is programmed to swim toward the light. Object avoidance is a behavior created by linking the input from the two infrared proximity detectors, located on the sides, with the rapid turning of the tail.

- The tail itself flaps back and forth, generating forward **thrust**. Depending on whether navigation or avoidance is the selected behavior, the tail can turn gently or dramatically while it flaps, changing the direction of RayBot.

- The team confirmed that the shape-changing tail worked using a servomotor, and then they turned to the problem of creating muscle-like actuators. They did more work to understand how electric rays glide. Their detailed understanding of electric rays made it clear that they needed RayBot to change its posture and the **camber**, or shape, of its body disk.



© Greg Amptman/Hemera/Thinkstock.

Electric rays use their body disk to glide, making them highly efficient and stealthy—an ideal model for an underwater robot.

- To get the surface RayBot working under water, the team enclosed RayBot 2.0 in a larger, flexible, and waterproof body disk. They also added some pectoral fins behind, for steering up or down or to roll—called attitude control.

- Because water and electricity can't interact, the first step in engineering an underwater RayBot 3.1 was to enclose the electronics in a waterproof box, called a hull in nautical engineering terms. While the team needed to keep the electronics dry, at the same time, they needed to put holes in the hull in order to communicate electrically or mechanically with the actuators outside of the hull.

- On the back end, there is a flexible silicone tail functioning as the propeller. The team needed to be able to flap the tail back and forth and control how fast it flaps. Both fins and tail are controlled by servomotors, located inside the hull.

- To transfer the force and power of the motors to the external fins and tail, the team took another chapter out the book of nature and created tendons—polystyrene cables that connected the oscillating servomotors to the flapping fins and tail. To pass these tendons through the hull, they made tight-fitting rubber passages that could be loaded with a viscous lubricant.

- RayBot 3.1 used its flapping tail to swim. Different tail-beat frequencies could be programmed to control speed. When RayBot transitioned its fins from down to up, its attitude changed, and it moved vertically. The fins controlled pitch.

- However, the team ditched the pectoral fins. The way they had them configured didn't work very well for attitude control. So, they put that part of the problem on hold and created RayBot 3.2, which had just a tail. The biomimetic tail acted as both a propeller and a rudder.

- RayBot 3.2 had the ability to detect a light source, navigate toward it, and hold station around it. This was a simple proof-of-concept test, showing that they could build a working autonomous underwater vehicle prototype that was self-propelled and autonomous.

- In parallel with their work on the whole robot, the team also started trying different kinds of muscle-like actuators, and they analyzed whether to make them or buy them. They thought more about the flapping pectoral fins on RayBot 3.1. It turns out that in real fish, the fins are composed of thin rods of skeleton that give the fin support. Research has shown that the fin rods also actively flap the fin and change the camber of the fin.

- The team induced a shear using a **rack and pinion**. But unlike the commercially available rack and pinion attached to a rotary motor, theirs was driven back and forth by a servomotor connected to the pinion. They called this actuator the **fin-shear actuator**.

- This also led them to ideas about how to engineer a better body, one with a body disk that they could now actuate with the fin-shear actuator and control the posture and camber.

- To get the details right, they bought a large electric ray and made a plaster cast of it, which they then could use to make a silicone version. The problem with making a body this way, though, is that it had asymmetries, bumps and bulges that were partly a result of the way they made the cast and partly a result of the imperfections of nature.

- So, the team used the cast of the ray to take a series of very detailed measurements of the body shape. From those measurements, they then created a three-dimensional computer graphic version of the ray in a software program called Maya. Then, they made the computer graphic version of the ray symmetric and found that in order to come up with a mold that their machine could fabricate, they needed to slightly alter some of the lines and curves of the ray's body. The result of this process was RayBot 3.3.

- When they cast the final body of RayBot 3.3, they used a flexible, silicone-based polymer called Dragon Skin. It took nearly 20 pounds of Dragon Skin and a lot of acrobatics to create a mold and then peel RayBot out of it. The mold was in nine pieces so that RayBot could be taken out of it without breaking its body.

- The team also designed into the mold a payload in the belly, on the bottom of the robot body. Into that payload, they placed the fin-shear actuator system with a controller to drive the servomotor to cause the tail to flap side to side. The fin-shear actuator mechanism has sufficient power density to be able to propel RayBot 3.3.

- The challenge for RayBot 4.0 is to add sensors to the body of RayBot and to use the fin-shear actuator system to camber the body and take advantage of the winglike properties of the body for even more energy efficiency.

Activity

Think of the simplest engineering activity you can imagine, and write it down. Then, using that design as a plan, build it. Then, test it to see if it works.

The engineering process is iterative: It is a cycle, a continuous process.

Important Terms

camber: The measure of the front-to-back curvature of a wing that helps create lift.

fin-shear actuator (FSA): A biologically inspired design that converts shear (strain from lateral stress in the layers of the fin) into bending.

rack and pinion: A type of actuator that converts a motor's rotational motion to a translational output.

thrust: A force that propels an animal, vehicle, or robot.

Suggested Reading

Dym, Little, and Orwin, *Engineering Design*, chaps. 1–2.

Hanson, Rus, Canvin, and Schmierer, "Biologically Inspired Robotics Applications."

Other Resources

The IEEE Robotics and Automation Society: http://www.ieee-ras.org/.

Questions to Consider

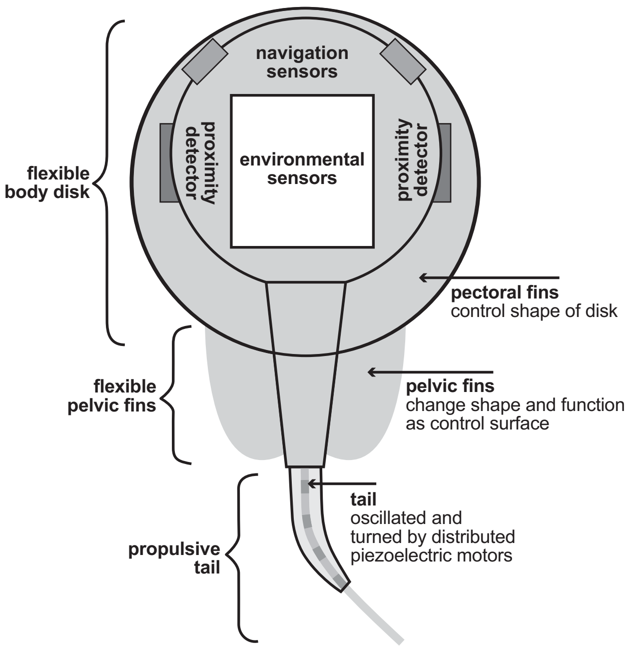
1. In which of the following scenarios might engineering a robot be necessary instead of hacking?

a. You need a robot that carries your webcam around the house when you aren't home.

b. You need a quadcopter robot that can inspect the insides of industrial chimneys.

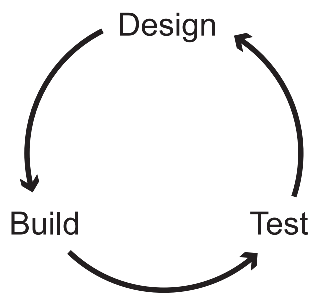
c. You need a robot to explore the liquid water under the icy crust of Europa, one of the moons of Jupiter.

2. Looking at the following conceptual design for RayBot 4.0, what features and trade-offs can you anticipate for operating in the open ocean?



3. Hatching new ideas for robotics companies is an entertaining way to think about designing new kinds of robots. Identify three features of a robotics company that would make it well poised to succeed.

Many people first getting started in design or engineering don't think that the process will be iterative for them. They think that *their* design process can be a linear flow of design, build, and test.



If yours works the first time, then you might be deemed a genius, or you might be accused of having had prior experience with that particular design. Good luck!

A Robot for Every Task?
Lecture 10

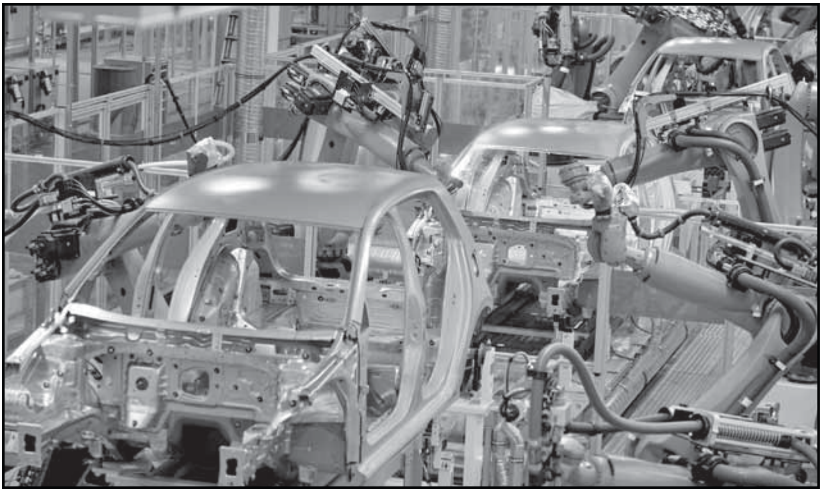
If we want a robot to perform a given task, we have choices about which way to go. We could use a generalist robot, a robot that can do a large range of tasks, including tasks for which it wasn't originally designed or programmed. We could redesign an existing specialist robot, to add to it the new capability we want. We could create new specialist robots, such as robotic trash cans, that focus on one job. The fact that we so often have these choices tells us a lot about how the field of robotics works.

Robots That Do Jobs

- To design a robot to do a job, the key is to first understand the job. Most of the time, a job looks simple, at first, and then the deeper you get into it, the trickier it becomes. Understanding the job involves understanding two parts: the workplace and the task itself.

- Sometimes, the workplace is as specific as a bedroom in a house, or it can be more general, such as any room in any house. The task is done in the workplace you've just specified. You can't have a task without a workplace. In that context, you need to be able to break down a task, analyze it, and determine its subcomponents. This is analogous to what we do when we write an algorithm for the computer in our robot's microcontroller. We need a step-by-step understanding of what the task entails.

- In the case of taking out the trash, start with the workplace. Imagine a laboratory. Think about the layout with regard to the generation of trash. There is a small trash can that is kept in a convenient location and a large trash can into which the contents of the small trash can are dumped. When the large can is full, then it is eventually emptied. In addition, there is some clutter on the floor, including other robots, along with a large lab bench. The floor itself is hard and smooth. There are no stairs or other environmental hazards for a robot.



Because factory floors are stable workspaces, robots do not need to be programmed to sense humans, freeing up energy and computer space.

- When we look at a workplace, a big deal in robotics is to think about how stable it is. A stable workplace is called a structured workplace, and it's one in which the world doesn't change. It maintains its structure: Most objects don't move, and those that do, do so in predictable and regular ways, and people don't come and go willy-nilly. A factory floor with an assembly line is a classic example of a highly structured workplace. You can map out a structured workplace on one day and have that map still be accurate on the next.

- On the other end of the spectrum are unstructured workplaces, which change all the time and change in ways that are often not predictable. Most busy, crowded places outside are unstructured. People, animals, and vehicles come and go. Rain falls. Bicyclists ride by.

- A laboratory falls in between. It is structured in the sense that the walls, table, and configuration don't change. It is unstructured in the sense that people and robots come and go. This kind of intermediate structure is typical of most workplaces, such as houses, schools, and offices.

- Knowing the level of structure allows you to make important decisions about what your robot needs to be able to sense—to keep track of—and what it can ignore. If you can ignore having to identify objects, then you save a lot of time, energy, and computer space. A structured workplace allows you to build into the programming of your robot implicit assumptions about the world.

Taking Out the Trash

- Consider the task of taking out the trash. This will be a series of events. On some regular schedule or when it starts to get full, we want the small trash can to be emptied. It seems simple: To empty the trash, we pick it up, take it over to the large trash can, empty it, and return it. The following is a list of subtasks involved in getting an as-of-yet-unspecified robot to do this job.
    - Turn on when trash needs to be emptied.
    - Navigate to the location of the small trash can.
    - Grasp the can.
    - Lift the can.
    - Move the can to the large trash can.
    - Lift the can higher.
    - Tilt the can over the large can.
    - Empty the contents into the large can.
    - Check to make sure that the small can is empty.
    - Return can to transport position.
    - Return can to its location.

- ○ Home to charging or standby dock.
- ○ Await next set of instructions.

- What's useful about task analysis is that we can immediately start to see what we need the robot to able to do. We can start thinking about what the robot needs in terms of sensors, actuators, and sensory-guided movement.

- In terms of kinds of sensory-guided movement, the trash-emptying robot must be able to navigate around the laboratory, and it must be able to manipulate the trash can. That navigation and manipulation are required is a good sign: They are the two fundamental kinds of behaviors that robots have to execute in nearly every job you might want a robot to do.

- At this point in the design process, it is critical to ask this question: Now that we know what needs to be done and where it will happen, is there an existing robot that can do the job? The practical version of this question is this: Is there an existing robot that can do this job that is available for me to use?

- If the answer to either question is yes, then you want to get the robot and have it do the job for you. You want to stop designing a robot for this task and put your efforts elsewhere, at least for the moment. Don't reinvent the wheel unless you have an idea for a better wheel.

- For the task of emptying the trash in the laboratory, the answer to this question is: We're not sure. So, let's investigate and see what robots out there might be able to do. If we think about navigation in the lab, a cleaning robot like Roomba does a very good job of going out and cleaning.

- However, Roomba, for the most part, is not doing point-to-point navigation. It's not until we ask it to return to its dock that is has a specific place to go. Roomba shows us how we might set up our navigation system in the lab, with infrared beacons like what we have on the docking station. The beacon is like a lighthouse, sending out a steady signal. When Roomba finds the signal, it can move directly toward it.

- To help Roomba find the trash can, we could take one of the virtual wall lighthouses and use it as a beacon to help guide the robot to the trash, which we could set up with the equivalent of a docking signal.

- While Roomba could get going and navigate to a can, fulfilling the first 2 of our 13 subtasks, we see that in terms of subtask 3, grasping the can, we are in trouble. Roomba can't do the job because it doesn't have a manipulator to transport and then empty the can.

- Perhaps instead we can find a robot that is more of a generalist that we might easily program to do the job. When we think of a generalist, we humans tend to think of humans, because of the huge number of different tasks that we can accomplish. So, we think of a humanoid as a machine that could step right in and empty the trash.

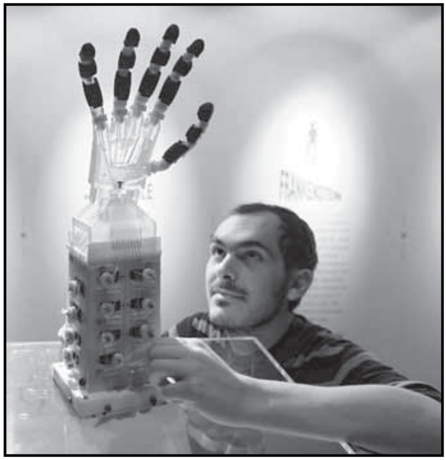
- Baxter is an example of a humanoid that is built to be reprogrammed—and reprogrammed easily, without having to write any computer code. This is an advantage over Roomba. You train Baxter by showing it what to do. Baxter's ability to learn by doing makes it one of the best multipurpose robots that is commercially available.

- Baxter is not a mobile robot like Roomba; it doesn't have legs or wheels to move around the lab. So, neither Baxter, the versatile manipulator, nor Roomba, the able-bodied navigator, can pick up the trash and take it out. We need to combine the two.

- Clawbot has an arm, and it can move around. We can run Clawbot by remote control to test what it can do. When we try to get it to empty the trash, we discover that Clawbot lacks the kinematic degree of freedom that would allow it to twist and get its end effector, its claw, oriented so that it can pick up the can.

- Whenever your robot can't do something, you have two choices: rebuild it or change the world (the workplace). In either case, we are talking about designing a way to allow the robot to be successful. In the case of Clawbot, we'd want to rebuild it by adding at least one and probably two degrees of freedom to the wrist.

- A kinematic degree of freedom in human bodies involves joints. Each simple hinge joint in your finger adds one rotational degree of freedom to your hand. Each degree of freedom in the hand means that you can refine your movements, the things that you can grasp and manipulate.



© Oli Scarff/Getty Images News/Thinkstock.

Each hinge joint gives hands greater freedom and control.

- The claw needs some added dexterity. We need the claw to roll 90 degrees and then pivot so that the fingers are oriented vertically and down. Then, it could pick up the can. We'd need two more joints with motors, and we'd have to reprogram the software to allow this to happen. The additional motors would add weight that the motors support, so the payload maximum in the trash can would have to decrease. Perhaps there's a better way to go than rebuilding this robot.

- For either Clawbot or Baxter, this is when we wish that the arms could telescope if and when we needed them to. This would add a kinematic degree of freedom to the arm. This linear motion of telescoping, such as opening a tripod leg, is called translation, and it's a very different kind of motion than we see in the vertebrate body. The closest we get to telescoping in translation is sticking out our tongue.

- Imagine what it might be like to overcome the shortcoming of each robot by getting them to work together. Clawbot is mobile with very rudimentary grasping capabilities. Baxter has great grasping capabilities but is not mobile. Combining their mobility and manipulation might result in a great power couple.

- First, Clawbot is working by remote control. We'd want to program it to operate autonomously like Roomba—to sit in a corner, charging and waiting until scheduled or called to turn on. Then, it would navigate to the small can, lift it, and move it to the large can. Then, Baxter would take the can and empty it and hand it back to Clawbot, which would then do the return journey.

- Navigation and manipulation are central to taking out the trash. These two abilities are central to a whole host of tasks that we might want robots to do.

Suggested Reading

Angeles and Park, "Performance Evaluation and Design Criteria."

Dym, Little, and Orwin, *Engineering Design*, chaps. 4–6.

Other Resources

Professor Rus's website at MIT: http://groups.csail.mit.edu/drl/wiki/index.php?title=Main_Page.

Questions to Consider

1. From what you know about robots, what tasks are robots particularly suited to do—and to do well?

2. Professor Manuela Veloso is working on a type of common sense for robots: knowing your limits. What might be the best way to program a robot to know its own limits?

a. Anytime the robot fails to complete a step of a task in a certain amount of time, it would then categorize that step (or the task) as beyond its limits.

b. Anytime the robot is unable to carry out steps in the sequence in which they are specified, it would then categorize that entire task as beyond its limits.

c. A robot could run an internal model of itself, a simulation of the task it is trying to complete. The model is updated by the robot's current position and current activity, as well as the state of the local world. With these inputs to the simulation of itself, the robot predicts the probability of success in conducting the next steps or tasks. Low-probability predictions are judged to be beyond its limits.

Robot Arms in the Factory

Lecture 11

More than a million robots are at work in factories all over the planet. Industrial robots help us manufacture cars, electronic and medical devices, medicines and pharmaceuticals, food and beverages, metals, chemicals, plastic, and rubber. Virtually all of these robots are not humanoids; they are specialized robotic manipulators, and their workplace is the highly structured, highly predictable world of the industrial factory. The tasks of robotic manipulators involve repetitive, quick, and highly accurate movements to grasp, move, manipulate, and assemble. With tools in hand, these robots file, paint, assemble, cut, polish, flame, weld, bond, glue, seal, inspect, and sort.

The Anatomy of Autonomous Robots

- Industrial robots all began as variations on a theme: a robotic arm. Robotic arms have a natural affinity with human anatomy. Their grasper can be thought of as the fingers on your hand. The wrist moves the grasper, and there are some elements that look like the forearm and upper arm. The base joint can be thought of as the shoulder, and there also is an elbow.

- In engineering terms, a robotic manipulator is a linkage system, with motion limited to and controlled at the joints. The link is a rigid body that maintains the distance between joints and often contains the motors that power the movement of the neighboring joints.

- The joint of a robotic manipulator is usually one of two types: revolute or prismatic. Revolute joints rotate or revolve, and prismatic joints slide. We humans have mostly revolute joints, and many of those joints revolve around a hinge, so we call them hinge joints. Hinge joints, such as your elbow, rotate through a single kinematic degree of freedom.

Hinge joints rotate through a single kinematic degree of freedom and can be described with a single mathematical variable, which, for the elbow, is approximately 150 degrees.

- **Kinematics** is the study of motion without considering the forces that generate those motions. Include forces in your study of motion and it's called dynamics. Kinematics and dynamics are hugely important in factory robotics because we want a machine whose motions can be precisely controlled.

- When a joint has a single degree of freedom, in kinematic terms, we say that its position can be described with a single mathematical variable, which we typically measure as an angle. For your elbow, its position can be described by the angle between your forearm and your upper arm. In other words, that single degree of freedom can rotate through an angle of about 150 degrees.

- The heart of autonomous robotics in the factory is the design of movements guided by sensors. Careful planning (or preplanning) by humans is key to designing and controlling robots in the factory. Anatomy is at once an inspiration and a challenge for roboticists. Many motors make the predictable, accurate control of motion extremely difficult, even when you have sensors to monitor the position of every joint.

- Designing robotic manipulators always involves trade-offs between range of motion and control of motion. One common engineering trick is to constrain every joint to be a single degree of freedom.

This allows you to simplify the motion and its control. Then, you can add degrees of freedom back in, with more single-degree-of-freedom joints.

- You could also make more complex joints, such as the ball-and-socket joint, but you won't find those on most robotic manipulators. That's because the fundamental function of a robotic manipulator is simple pick and place: pick something up and place it somewhere else. We humans have been doing this for as long as we've been on two feet, picking fruit and wielding tools.

Industrial Robots

- The first record we have of a programmable pick-and-place robot is from 1938. Griffith P. Taylor's robot was programmable, and it picked up a series of bricks and created a circular stack, automatically, in about 50 minutes. In total, Taylor's robot had five degrees of freedom to allow it to pick and place bricks. Some people consider Taylor's pick-and-place robot to be the first manipulator robot.

- As so often happens, it takes years for an idea for a robot to be translated into commercial applications. The first U.S. patent for a commercially successful industrial robot was granted in 1961, to George Devol. The patent involved tracks along which the transfer apparatus moved. Parallel to those tracks ran a conveyor belt. The arm of the transfer apparatus reached across the conveyor belt, grabbed cartons from the pallets, and then transferred those cartons to the conveyor belt.

- A farsighted feature of Devol's patent was a feedback loop between the sensors detecting position and the program controller. Devol's patent envisioned the robot knowing that its arm or gripper had gotten into the right position because a sensor detected that motion. When it works, this is **proprioception**, although reliable sensor-based movement turned out to be an ongoing challenge for robotics.

Industrial Robots

1956	Unimation, the world's first robot company, is founded by George Devol and Joseph Engelberger.
1958	Numerically controlled (NC) machines are first produced by FANUC in Japan.
1961	"Programmed Article Transfer" (U.S. Patent 2,988,237), key intellectual property for Unimation, is issued to George C. Devol, Jr.
1962	Unimate, the first industrial robot from Unimation, goes to work unloading die casts at the General Motors factory in Trenton, NJ.
1973	Famulus, the first industrial robot with six electromechanically controlled axes, is introduced by German robotic company KUKA.
1973–1974	T3, the first commercially available robot arm controlled by a minicomputer, is designed by Richard Hohn and released by Cincinnati Milacron.
1976	Robot arms are used on Viking 1 and 2 space probes.
1981	SCARA industrial robots are introduced in Japan.
1985	FANUC develops robots to assemble other robots.

- By the time that Devol received his patent in 1961, he had already formed a company, in 1956, when he first applied for the patent. The company, formed with serial entrepreneur Joseph Engelberger, was called Unimation, and it became the first company to build robots that went to work for industry. Those robots were called Unimates.

- The first Unimate robot was installed in 1962 in a General Motors factory. It used the Unimate to lift hot pieces of metal from a die-casting machine and stack them. These first industrial robots led to KUKA robots, and many others. These classic industrial robots—all very much like the original Unimates—are still used in thousands of factories worldwide.

Range of Motion versus Control of Motion

- Accurate and repeatable control of robotic manipulators was critical for their development, and two different ways of thinking about movements developed: forward and inverse kinematics. In both approaches, we are interested in the position and orientation of the end effector, the hand.

- In forward kinematics, we can compute the end effector's position by measuring the joint angles and knowing the lengths of the links. Trigonometry will do the trick mathematically, and it allows us to transform what we call the joint space of angles into the Cartesian space of the x, y, z position of the effector. With forward kinematics, it is easy to move the robot and then calculate its position.

- In inverse kinematics, we pose the problem the other way around. This may be much more difficult to do. We first identify the position that we want the end effector to occupy, and perhaps some constraints about the path to take, and then calculate the movements of the arm that will do the job. Inverse kinematics is the formal way to wrap our heads around the problem of the trade-off between range of motion and control of motion.

- Given this trade-off, one of the most important things to consider when building or buying an industrial robot is the minimum number of degrees of freedom that you need. Every time you increase the range of motion by adding a degree of freedom, you exponentially increase the computational problem of controlling that motion. In practice, the range of possible solutions is usually obtained by imposing additional constraints.

- So far, we have mostly been considering what might be a numerically controlled machine tool, or a blind robot in an open control loop. But if we want a robot that knows whether its arm is in the right place, then this is when sensors come into play. Sensors can measure the actual position of each joint and provide feedback to the controller for possible adjustments.

- The sensors that robotic manipulators use to measure the position of their joints and end effectors vary. If the joints are powered by servomotors, the sensory control is actually built into the motor. Inside the servomotor is the electric motor and the equivalent of what we call a potentiometer, a variable resistor that signals angular position by changing voltage in a circuit.

- Many industrial robots use stepper motors, which are more powerful and precise than servomotors. Stepper motors don't have built-in sensors, so you have to use external sensors. Servomotors and stepper motors measure and adjust position continuously.

- From the perspective of range of motion, one of the amazing things about adding a degree of freedom is that by the laws of combinatorics, the number of possible positions, or states, of our system explodes. This increased range of motion, plus control over motion that is more flexible, gives the robot what we commonly call dexterity. The challenge is that as we increase range of motion and dexterity, the complexity of our control system has to increase, too.

- In 1973, the KUKA robot company made headlines when they introduced the first industrial robot with six degrees of freedom. The robot is called Famulus, and it was built for automotive manufacturing, as about a third of the industrial robots are still today. Famulus is an **articulated robot**, which means that it has any number of revolute joints. You will see articulated robots in the food industry.

- A very different manipulator is a **Cartesian robot**, which has three prismatic joints—telescopic joints—that are arranged perpendicular to each other. Because the joints are sliding, they can also be made into very long tracks, into what are called **gantry robots**.

- A spherical or polar robot replaces the revolute elbow joint of an articulated robot with a prismatic joint. The term "sphere" refers to the shape of the working space of the end effector. The Unimate robots were spherical robots with a telescopic joint that allowed them to reach for a sample or weld a particular spot and then retract to get out of the way.

- Because spherical robots have just three degrees of freedom, they, too, are easier to control than articulated robots, which usually have six. But because they lack a large range of motion, we tend not to see many spherical robots in operation these days.

- In between the spherical robot and the Cartesian robot is the cylindrical robot, where the working space is a cylinder. To get this shape, the first joint needs to be prismatic, raising the arm up and down. A revolute joint spins to create a circle, and a second prismatic joint telescopes a revolute wrist joint. Cylindrical robots are used in applications like spot welding.

- A big breakthrough for simplicity came in the late 1970s, when the first SCARA (selective compliance assembly robot arm) robots were developed, supposedly inspired by a Japanese-style folding screen. The trick was in redesigning the work into a series of simpler tasks to suit the simpler robot.

- SCARA robots have three revolute joints, but unlike ordinary articulated robots, all three joints operate in the same plane so that SCARA robots can move the end effector very precisely, more precisely than articulated robots. The end effector is located on a fourth joint, a prismatic joint, that's perpendicular to the plane of the revolute joints.

- Because the three revolute joints aren't restricted in size by needing to move perpendicular to each other, SCARA robots can also have very large motors to power the joints, making them some of the fastest robotic manipulators available. SCARA robots are very good for things like the surprisingly difficult task of placing and tightening screws.

- While the early industrial robots did the same job all day, day after day, a breakthrough for flexibility came with the PUMA (programmable universal machine for assembly) robots. These were programmable robots that were developed at Stanford and supported by General Motors.

- Finally, the world's fastest robotic manipulators are typically parallel robots. The idea is to build multiple arms in parallel, working together, like pairs of arms. The Adept Quattro has four arms in parallel, working for precise, rapid movements.

Collaborative Robots

- Because robotic manipulators work very rapidly, often with a lot of power, part of the design of the robot is to also design the workplace. Safety is a big consideration. In 1979, in a Ford Motor casting plant, Robert Williams became the first human killed by a factory robot. Reports say that the robot had begun moving more slowly, and Williams tried to climb into a storage rack to remove parts by hand when he was hit in the head by the one-ton robot and killed instantly.

- The standard way to make it safe for humans and robots to work together is to keep them from working together, at least physically. The work areas for industrial robots in traditional factories are very clearly defined, and humans are kept out of those areas, which are called safety cages.

- But these restrictions limit the work that robots do. Flexible, easy-to-use robots that can work alongside humans were goals for Rodney Brooks when he created the company that is now called Rethink Robotics, which created the robot Baxter, released commercially in 2012.

- As one of the first collaborative robots, or cobots, Baxter is defining new roles for industrial robots and helping to bridge from industrial robotics to service robotics, where robots are designed, from the start, to interact safely with humans.

- Other robotics companies are also working to make robots more collaborative with humans. KUKA is making slower, lightweight robots with compliant joints for work as **service robots**.

Important Terms

articulated robot: A manipulator or arm that has hinge (revolute) joints.

Cartesian coordinate robot: An industrial robot with three perpendicular, translational degrees of freedom.

gantry robot: Large type of Cartesian coordinate manipulator robot whose degrees of freedom are in translation and at right angles to each other.

kinematics: The study of motion without regard to the forces generating them.

proprioception: Internal sensing of the motion and force of the robot's joints and motors.

service robot: Any robot built to assist humans, excluding robots involved in manufacturing.

Suggested Reading

Anonymous, "An Automatic Block-Setting Crane."

Craig, *Introduction to Robotics*, chap. 1.

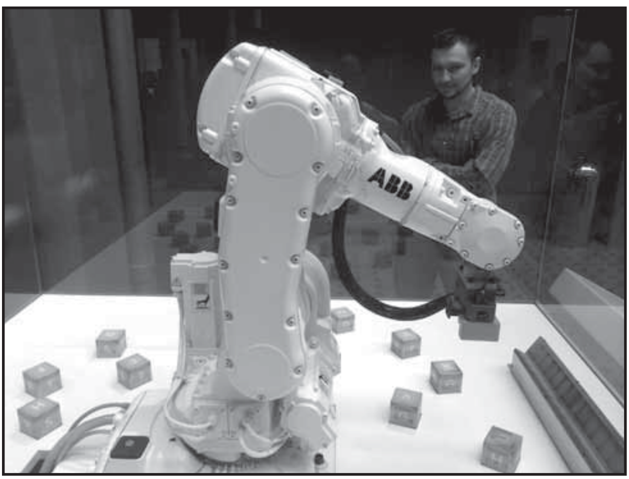
Devol, Programmed article transfer, U.S. Patent 2,988,237.

Other Resources

Rethink Robotics, maker of Baxter: http://www.rethinkrobotics.com/products/baxter/.

Question to Consider

1. The following is a robotic arm produced by the company ABB for the National Museum of Scotland. Visitors type in their name on a keyboard, and then the robot picks up blocks, deposits them on the metal shelf in the lower-right side of the picture, and that spells the person's name.



© John Long.

What capabilities must the robotic arm have in order to do this spelling task?

Mobile Robots at Home

Lecture 12

Service robots can be used for personal or professional use. Most of the time, different jobs are done by different robots that are specialized for one job. For robots that are designed to work at home, three features are proving to be the keys to success of personal service robots: They are specialized for one job, are of small size, and possess autonomous mobility. An important part of being a specialist is that the robot doesn't need to be reprogrammed. Specialists have one job to do, and they do it out of the box.

Roomba

- Roombas weren't the first home robot vacuum, but they were the affordable and reliable machine for personal use that transformed lives. Like any successful commercial product, Roomba has attracted competitors—including Neato, Navibot, Hom-Bot, bObsweep, Deebot, and the traditional vacuum makers, and Roomba itself has undergone a regular series of updates and improvements.

- In 1997, iRobot made a Roomba prototype called Dust Puppy, which was more like a dust mop than a vacuum cleaner. But Dust Puppy showed that you could use very simple behavior-based control algorithms to have a mobile robot navigate around your home and get a job done.

- By 2002, the first model of Roomba was sweeping and vacuuming. One of the important design features is that Roomba goes where you don't see or normally go: under furniture. This explains a decision that they made about the design: It had to be small in height to fit under furniture.

- In 2004, iRobot introduced the second generation of Roomba, originally called Discovery but then renamed the 400 series. Roomba 400s have enhanced autonomous navigation. These Roombas can also be programmed to clean on a schedule, and they have a bigger dustbin than the original models.

- In 2007, iRobot introduced the 500 series. In this third generation, iRobot added automatic docking for autonomous recharging of batteries, as well as the virtual lighthouse, an infrared beacon that can speed Roomba's navigation between rooms or help keep it from going into a room, depending on the setting. These Roombas also move faster and need forward-looking infrared sensors to prevent high-speed collisions. They also avoid getting caught on cords by reversing the rotational direction of the brushes if resistance is encountered.

- The fourth-generation 600-series Roombas were introduced in 2008, and they have a new kind of cleaning head that doesn't get hair entangled. Instead of having bristles, the brushes have treads and are called extractors. These extractors are part of a redesign that includes a higher velocity of airflow sucking up dirt and debris. The 600s also have an improved **acoustic sensor** for detecting dirt, which allows Roomba to spend more time on really soiled parts of the floor.

- Introduced in 2011, the 700 series has better battery life, a HEPA filter on the vacuum, and wireless scheduling. Introduced in 2013, the 800 series has even better battery life and cleaning capacities, including an optical camera to look for fluffy debris.

Scooba and Braava

- In 2005, iRobot introduced its second home robot, Scooba. Its job is to scrub floors, and the model 450, introduced in 2014, uses a three-pass cleaning process that includes vacuuming as the first step. As it vacuums, Scooba lays down a thin film of cleaning fluid to presoak the floor. On the second pass, it scrubs and squeegees, sucking up the dirty solution. The final pass continues to remove the liquid.

- Scooba has the same basic circle-shaped design as Roomba. A circular design is great for a floor-cleaning robot in the home because a circle lacks edges to ding and get caught. The short stature, too, is great for getting under furniture. In other words, the robot body is built to get the robot under and around.

- An exception to the circular body design is found in iRobot's Braava, a robot that does either dry or damp mopping. While behavior-based control architectures are simple and robust, how they clean is inefficient. Some areas get more coverage than others. To make sure that most areas are covered, Roomba and Scooba are programmed to make multiple passes. That takes more time and battery power. Thus, the trade-off for behavior-based cleaning is simplicity and robustness versus efficiency.

- To create an efficient, one-pass cleaning robot, iRobot needed to create a robot that could navigate using a map. GPS isn't a good option inside for two reasons. First, the signals from satellites can be scattered, distorted, and blocked by walls and metal structures. Second, even if you can get a good GPS fix, the accuracy of the traditional consumer fix, even in the best outdoor conditions, has been more than 10 feet, which is not sufficient accuracy to guide indoor robots.

- Evolution Robotics built a navigating robot called Mint. When iRobot acquired Evolution Robotics in 2012, Mint was renamed Braava. Braava works using an infrared navigation system called NorthStar, which uses a beacon that sits on a table and projects infrared dots onto the ceiling. Braava uses three infrared sensors to find its distance from those dots and triangulate its position.

- When iRobot patented their own version of NorthStar's infrared technology in 2010, they called it "celestial navigation." Braava uses infrared navigation common to NorthStar and celestial navigation to create its own map of the room.

- Knowing its position on the map, Braava can plot a course that moves it back and forth across your floor, in beautiful, tight rows. This gives complete coverage in one pass when Braava is dry mopping. In damp mopping mode, Braava still navigates, but it moves with side-to-side and back-and-forth scrubbing motions as it moves along the rows.

- Creating a map and knowing where you are on the map at the same time is hugely important task in autonomous robotic navigation. It is called **simultaneous localization and navigation**. This is a model-based controller architecture that is computationally intensive.

- Now we can understand the square design. If you know where you are and where other things are on your map, you can plot your course to avoid objects. If you know what obstacles lie ahead, then a square design is less likely to get you stuck in a corner. You avoid the corner, and you can also aim deliberately into the corner. Also, Braava can put the cleaning pad out front, which makes it wider than the body. A wider cleaning pad means fewer passes—even better efficiency.

- However, there are a few trade-offs with Braava. First, because it has a wide mop out front, it doesn't head straight into a docking station. Second, in spite of having a map, Braava does sometimes get stuck in small areas where it can't maneuver itself out. Third, Braava has to have a good signal from the NorthStar infrared navigation system in order to clean an entire space. Once the cleaning cycle has started, you also have to take care to not move the NorthStar beacons; the navigation system has to be tuned to make the system work optimally.

- The big trade-off for Braava is between efficiency and robustness. When it works optimally, Braava can navigate and clean very efficiently. But because it is dependent on map-based navigation, it is not as robust as Scooba. Scooba and Braava clean differently and have different trade-offs.

Mirra

- Another popular type of home robot works outside, in your pool. Mirra 530 is an autonomous pool-cleaning robot built by iRobot. Mirra has wheels for locomotion and a rotating scrub brush. It also filters water, which it sucks in through an opening up front, so it removes both small particles and large items, such as leaves. Most impressively, Mirra can move up vertical walls and navigate steps. Its wheels help give it traction, and the suction it creates for filtering water helps it hold its ground.

- Unlike Roomba, Scooba, or Braava, Mirra has a tether. The cable is a low-voltage energy source from a topside transformer that is converting alternating current from the household energy supply to the local direct current that Mirra uses. Low-voltage direct current is very safe to use around water and people.

- Why not just use a battery for Mirra? What's the trade-off when we use a battery? On the positive side, being untethered means that the robot won't get tangled if it goes under and through the legs of a chair, and it won't have stay in close proximity to its external energy supply. On the negative side, batteries have a limited time for which they can supply electric power. To extend the robot's mission time, a bigger battery can be used, but bigger batteries increase the weight that the robot has to lug around.

- The designers at iRobot had a choice: Give Mirra a big enough battery to handle the high power needs of its three motors over enough time to complete three jobs—scrub, locomote, and filter—or power Mirra through a tether with a topside energy supply.

- A big battery, being full of dense metals, would be great for keeping Mirra on the bottom of the pool. The problem would be climbing the walls, when the weight of the battery would tend to peel Mirra off. The way to counteract this would be to create more suction. But more suction makes it more difficult to move. You could increase power to the motors driving the treads, but then that's more power you need. This is a vicious circle.

- By giving Mirra a tether for its energy supply, the designers were able to give Mirra the power to climb walls and even scrub right at the water line, where the robot has to support part of its weight out of the water. More importantly, Mirra's effective time on station was increased, with time limited only by having to stop and clean the filters.

- To keep the tether from becoming tangled, Mirra has a gyroscopic sensor on board. Gyros measure accelerations that come from gravity and turning. This allows Mirra to keep track of how it is oriented and how many times and in what direction it has turned. After a certain number of turns in one direction, Mirra is programmed to turn back. This removes coils in the tether and prevents tangling.

Other Types of Service Robots

- A gutter is a great example of a structured workplace. Gutters vary a little in shape and width, but the overall job is predictable. What is not predictable is what kind of clutter you have to clean out. But if your job is to get stuff out of the gutter, that is a common function across all types of debris.

- A robot that does this job is Looj, the gutter-cleaning robot. It spins an auger and then uses tank track actuators to push that actuator along. An auger is usually a type of end effector for drilling, but here the idea is to take the same rotary action and use it to move material to the sides rather than to the rear.

- A much less structured outdoor workplace is a lawn. There are stable but irregular obstacles, such as plants and sheds, and there is terrain of different types, with rocks, hills, steps, pools, and ponds. However, lawn mowing is a job proven to be well suited to robots. In fact, after robotic vacuum cleaners, robotic mowers are the second largest group of home robots in use.

- Another household job that robots can do is clean the windows. Introduced in 2011, Winbot, created by Ecovacs, has an actuator system that creates suction that keeps it on the glass. It has infrared proximity sensors that allow it to calculate the size of the window, and then, like Roomba, it calculates a plan to clean the whole area.

© Sean Gallup/Getty Images News/Thinkstock.

In the near future, we might see larger, more humanlike robots that perform a wide variety of tasks.

- Effective and successful home robots thus far have been small, mobile systems that specialize. But as larger robots appear that can move safely with us in our homes and other spaces, size alone will open up new kinds of jobs for them. As home robots, and other service robots, gain more skills, and perhaps even a humanoid form, they, like Baxter, will move from narrow specialists toward becoming broader generalists. We'll increasingly be able to train our robots easily and quickly to do new tasks that are unique to our needs and spaces.

Important Terms

acoustic sensor: Also known as a microphone, a sensor that converts sound waves into electric signals that can be read by the controller.

simultaneous localization and mapping (SLAM): Subfield of navigation in which the robot uses sensors and communication systems to know its position as it creates a map of its surroundings.

Suggested Reading

iRobot, "iRobot© Owner's Manuals and Quick-Start Guides."

Xu, Qian, and Wu, *Household Service Robotics*, chap. 3.

Other Resources

Promotional video of the 800 series for Roomba: http://www.irobot.com/us/learn/home/roomba.aspx.

Hacking Roomba: hackingroomba.com/. While this approach is a bit dated, if you have an old Roomba, then get started here.

iRobot's official hackable Roomba, iCreate 2: http://www.irobot.com/About-iRobot/STEM/Create-2.aspx. Designed to interface with controllers like the Arduino Uno, this is iRobot's response to people wanting to hack Roomba.

Question to Consider

1. In the lecture, we came up with this prediction of Roomba's subsumption behavior: If the bump sensor is on, back up and turn, else if the light touch sensor is on, keep moving forward. What is a good way to test this prediction?

Hospital Robots and Neuroprosthetics

Lecture 13

Robots working in the hospital show an incredible diversity of types, from mobile robots carrying materials, to robotic manipulators in the hands of surgeons, to prosthetics using neural signals of the human body. Robots in hospitals work in at least three different workplaces: the busy, social world of bustling humans in the entrance hall, waiting areas, and patient wards and rooms; the carefully controlled world of the surgical theater; and the intimate world of prosthetics and exoskeletons used by an individual.

Courier Robots

- In January of 1991, a HelpMate robotic courier was installed in the Danbury Hospital in Connecticut by Transitions Research Corporation, a company founded by Joseph Engleberger after he sold Unimate to Westinghouse in 1984. HelpMate was the first hospital courier robot, designed to transport items and information.

- With funding from NASA, Transitions Research Corporation wanted to tackle the problem of delivering material in the relatively structured world of a hospital. So, in addition to using a model-based navigation system for hallways and elevators, they needed HelpMate to be able to detect objects and avoid them.

- To navigate, HelpMate used a map-based control system and a variety of sensors to avoid people and other moving objects. HelpMate got around the difficult problem of using a manipulator to open doors by sending a radio signal to each door to open itself. Elevators were also modified to respond to radio control by the robot.

- As a courier, HelpMate could deliver late meals and special dietary foods. It could replenish supplies to nursing stations or return samples collected by the nurses to the laboratory for analysis. Outfitted with a locked cabinet, HelpMate could securely move and deliver medications from the pharmacy, along with patient records stored on its computer. Up to 200 pounds of mail and packages could be delivered during a single trip.

- HelpMate used lights to signal its intent to humans, move through the door, and then signal to turn. In addition, HelpMate pioneered the use of three types of sensors to safely interact with people and things: sonar, video, and bump sensors.

- Sensing with sonar was accomplished with 28 **transducers** on the robot. These transducers send out an ultrasonic signal and measure the time for that sound to return. A short **time of flight** signals that an object is close by.

- Robot video has historically been very difficult to implement, in part because there is so much information to process. The team at Transitions Research Corporation working on HelpMate in the 1980s employed two very creative engineering solutions.

- First, they drastically reduced the amount of information that they were trying to get from the camera. Instead of trying to detect and classify any and all objects, they only sought to detect the ones right in front of the robot. With detection alone as the primary goal, they could secondarily calculate the distance to the object and its size. Bump sensors provided a third way to detect objects: the now-classic way, pioneered in a service setting by HelpMate, of letting contact with an object depress a simple electric switch.

- Why have three different sensor systems delivering information about objects? A single sensor system can give you an ambiguous answer. For example, if you relied on the vision system alone, you would only detect objects directly in front of you and nothing overhanging, such as a cantilevered countertop. In that case, the 28 sonar sensors, all sending out sound signals, would come in handy.

- To improve your ability to get the truth about the world, to disambiguate, you combine information from the redundant sensor systems in a process called **sensor fusion**, which requires that the robot be programmed to go beyond simple reflexive reactions and make a decision about when to trust one signal and when to infer the most likely situation using multiple sources of sensory information.

- HelpMate's solution to the problem of navigational drift—small errors that add up over time—was to get a navigational fix, by putting retroreflective tape on the ceiling at fixed intervals. HelpMate could sense those bands of tape, using a pair of long-range infrared sensors that continuously emitted infrared light upward, and then marked the peak intensities of infrared light reflected off the bands.

- One of the challenges that this system did not initially anticipate was the strange or irregular behavior of people. For example, people behaved differently in different parts and times of the hospital. Context cues were added to HelpMate's control system, providing rules for how to adjust behavior depending on location and time.

- HelpMate ceased production in 2006, but its success as an intelligent, mobile courier has spawned a variety of robots in the halls of hospitals, including cleaning robots, disinfection robots, and robots that can check the status of defibrillators and fire extinguishers.

- One of the trade-offs for courier robots like HelpMate is maneuverability versus carrying capacity. A very maneuverable courier robot is Swisslog's RoboCourier, which can turn in place like a Roomba, with a turning radius of zero. But it has a maximum payload of 66 pounds.

- By contrast, Aethon introduced a hospital delivery robot capable of carrying 500 or even 1,000 pounds of supplies. These are called TUG robots. The slim robot up front pulls along a large storage cabinet behind. And instead of needing tape on the ceilings, or other markers, Aethon chose to create preplanned travel routes for the robots, either using computer-aided design drawings of the hospital uploaded into the robot or by using lasers to manually guide the robot through preplanned travel routes in the facility.

- The TUG robots avoid drift by taking their navigational fix from the charging stations. The company developed software that is said to open any public elevator in the world. But because the TUG is less maneuverable, it may get stuck: In that case, one thing the TUG robot can do is ask nearby humans to take specific steps to help.

- Moreover, thanks to increasing computing power, the company can link the TUG robots with an automated tracking system for materials moving within the hospital. In fact, the TUG robots are also monitored by automatic algorithms from the company's Cloud Command Center, and if a problem is detected, a human is contacted who can connect to the robot remotely and even take remote control if necessary.

Telepresence Robots

- As good communication connections continue to improve, there are increasing opportunities for **telepresence** robots. iRobot, the company that made Roomba, teamed up with Cisco to create Ava, a platform for a mobile service robot capable of wireless telepresence. This makes possible a robot that can map and travel a floor but also engage in desktop-quality teleconferencing.

- Another company, InTouch Health, has used the Ava platform as the foundation for the robot called RP-VITA, which can navigate to a new place on its own, having a complete world model of the hospital and the lower-level reflexive behaviors to avoid hitting people and things that aren't on the maps.

- One of the things that VITA can do is find a patient. Sometimes, especially when a hospital is overrun with victims of a large accident, patients can be temporarily misplaced. To identify a patient, VITA uses a combination of face recognition and radio frequency identification tags. In a fluid situation, you could even have a bunch of VITAs keeping track of which patients were where, what their status was, and what was to happen next and then help schedule and guide the hospital staff.

- As we move from the mobile robots working hallways, wards, and rooms and into the surgical theater, we see telepresence in a completely different kind of robot. For surgeries and patients that qualify for minimally invasive surgery, Intuitive Surgical created the da Vinci Surgical System, a teleoperated robot that is a variation on the classic robot arm, also called a robotic manipulator. The difference is that da Vinci moves robotic arms inside the human body.

- Da Vinci has four robotic arms outside the patient. Each of these arms has extensions with sensors and actuators that go inside the body. Surgical robots like da Vinci keep surgeons in the loop. The surgeon functions as the controller in a remotely controlled robotic system. Thus, the surgeon senses the world inside the patient through three-dimensional cameras and issues orders to the actuators inside the body.

Robotic Strides in Medicine

2000 The da Vinci Surgical System, a robot by Intuitive Surgical allowing surgeons to be remotely present and operate inside patients, is approved by the FDA.

2014 The DEKA prosthetic device, controlled by electromyogram (EMG) signals from muscles, is approved for amputees by the FDA.

Capsule Robots

- Capsule robots carry sensors and actuators into the body, and some are self-propelled. Capsule robots got their start from a technique called capsule endoscopy, in which a capsule can be swallowed, and it takes with it into your body a tiny video camera. A camera working in the dark needs light, so robots bring small LEDs that light up the scene so that pictures can be taken of a person's insides to diagnose problems.

- These capsules are passive and not really robotic in the sense that they don't have actuators that help them move. But in 1994, a robotic endoscope was patented and built to be able to actively wiggle its way through a patient's gut. The front end of the snake carries a camera, other sensors, and a small knifelike device to cut tissue samples. The ability to snake around opens up new areas of exploration.

- The problem with the robotic snake endoscope is that it is big, and one danger is that it can perforate the walls of organs. So, a number of biomedical engineers have been trying to combine the small size of a camera capsule with the self-propulsion of the robotic endoscope.

- A group at the Scuola Superiore Sant'Anna in Italy has come up with self-propelled robotic capsules. These capsules can have four propellers on the back end. They are like miniature submarines, controlled through a wireless communication system.

Assistive Robots

- Medical robots are also working with the human body in a more ongoing way, collaborating with us to overcome medical challenges on an everyday basis. There are robots that help in physical and occupational therapy: so-called therapy robots. Known more generally as assistive robots, these are similar to what the press likes to call an exoskeleton. Assistive robots are strapped onto your body to help supplement or retrain your existing abilities.

- The Walking Assist, created by Honda, is a spin-off technology from their work on the humanoid robot ASIMO. The Walking Assist is a jointed, powered robotic limb, in principle and practice, working with the human to help strengthen joints and power each step. This is meant for people with certain kinds of muscle or joint problems who can still walk but can use the help.


© Matc13/iStock/Thinkstock.

The Walking Assist, an offshoot of Honda's ASIMOV technology, aids people who have difficulties with mobility.

- In addition, Honda has developed an exoskeleton meant for workers. The robotic legs have their own sensors, controller, actuators, and battery. The exoskeleton helps support body weight for humans when they have to stand for long periods or bend, crouch, or kneel repeatedly. As you crouch, all you have to do is balance; the robotic exoskeleton does the work of supporting your weight.

- The trick is to have the Honda leg cooperate with the human. This works because sensors in the joints of the robot detect initiation of movement by the human. The legs then work to make strides even and to make them longer. The patient has the job of cooperating with the robot that they are wearing.

- An even more impressive area for this kind of cooperation is the field of prosthetics. Prosthetic limbs can be very complicated machines that we attach directly to the nerve endings after amputation. The idea behind the field of neuroprosthetics is to tap directly into the human's nervous system for control.

- This is done with what amounts to a special antenna, called an electrode, that is designed to pick up and amplify the electric signals generated by nerves and muscles. The electrode can be on the surface of the skin, under the skin touching muscle, or in the body right next to or even inside of the nervous system.

- Once the electrode is in the right place to detect nerve signals, then you have to train the human how to voluntarily make and control the electric signals that the electrode reads. The human acts as the controller and is communicating directly to the robotic arm via his or her body's electric signals instead of his or her body's muscular forces.

Neuroprosthetics tap directly into the nervous system for control.

Activity

Find and watch the film *Fantastic Voyage*. As outdated as it is, it is a very important cultural reference, and aside from miniaturizing humans, the idea of micro medical robots working inside our bodies is getting closer to reality.

Important Terms

sensor fusion: The process of combining information from multiple sensors to create information that is not available from individual sensors alone.

telepresence: Using sensors and instruments on robots to observe and measure without the need for a human to be physically present.

time of flight: Duration for a signal to travel from its source to a sensor.

transducer: Any device that converts energy in one form to energy in another. For example, a touch sensor transduces the kinetic energy of movement into electric signals; a motor transduces electric signals into kinetic energy.

Suggested Reading

Burdet, Franklin, and Milner, *Human Robotics*, chaps. 1 and 11.

Ciuti, Menciassi, and Dario, "Capsule Endoscopy."

Grundfest, Burdick, and Slatkin, 1994, Robotic endoscopy, U.S. Patent 5,337,723.

Other Resources

Da Vinci Surgery company website: http://www.davincisurgery.com/. The material at this site is constantly changing, and that's part of what is fascinating. Because the site is really set up for patients, you'll have to dig a bit to find out about their latest technology.

Questions to Consider

1. The da Vinci robotic surgery system was created to have a doctor in the loop. What added capabilities would we most need if we wanted to turn da Vinci into an autonomous surgeon?

2. If you wanted to improve the performance of the DEKA neuroprosthetic arm, how might you do so?

Self-Driving Vehicles

Lecture 14

Robots on the road are one of the most transformative emerging technologies on the planet. Many of us already have elements of robotic autonomy built into our cars, and we have for years. Creating robotic cars involves incredible challenges. Chief among them is the trade-off of speed versus safety: To drive safely, you slow down—but if you slow down, you take longer to reach your destination. The goal of the modern automotive industry is to enhance both speed and safety at the same time. Nearly every manufacturer has already embedded robotic elements into your car to assist you.

Cruise Control and Similar Features

- Cruise control is automatic speed control, and it is a robotic system because it involves sensor-guided movements. You let your car take over the gas pedal, allowing it to adjust the throttle to keep your speed steady as you travel up and down hills.

- Cruise control is an example of a system that works by negative feedback control. A sensor—in this case, the speedometer—is feeding information about the real world back into the system. The controller compares the actual speed to the desired speed. The difference between the actual and desired setting is called an error. Any system that uses negative feedback control is self-regulating and embodies the kind of self-control we see in autonomous robots.

- The automobile industry has been adding, one by one, different autonomous systems to vehicles. In response to cruise control's lack of braking, in 1995, Mitsubishi introduced a laser-based adaptive cruise control. The idea behind adaptive cruise control is called object avoidance in robotics: Don't hit the car in front of you.

- Most adaptive cruise control systems use radar as their primary sensor. Radar is an active sensor system; an emitter sends out radio waves at a particular frequency, and then a receiver collects the echo, the return of that signal. Using radar, you can measure the distance from you to the nearest object. You can also tell the relative speed of that object, whether it's heading away from you or toward you.

- With adaptive cruise control, you set a target speed, like old-fashioned cruise control, but you won't hit the vehicle in front of you because of the object-avoidance function. You end up following at a set distance behind the vehicle in front of you. That distance is adjusted based on the speed.

- To avoid collisions, you need to add an actuator to your toolkit. In addition to throttle control, you also need control of the brakes. You'll often find emergency braking sold as a feature on cars, and it is a part of the whole object-avoidance system.

- Even though adaptive cruise control is meant for the high-speed movements of freeway driving, some companies, such as Bosch, have a related system called Stop & Go that is built for slow speeds. In heavy traffic, Stop & Go will bring the car to a complete standstill if needed and then initiate movement when the car in front moves. This makes traffic jams far less frustrating if you are behind the wheel.

- One common cause of accidents is when you change lanes and don't see another car in your blind spot. Bosch offers Side View Assist. Using four ultrasonic sensors, two on each side of the car, the Side View Assist signals the driver when an object is detected in the blind spot.

- In cases where the car is simply drifting out of the lane—such as when the driver is sleepy—some companies offer Lane Departure Warning to signal the driver and Lane Keeping Assist to actually take over the steering. Video is a common sensor for lane keeping, because most paved roads have clear lines that mark the edges of the road.

- Park Assist can sense other cars and be used when you want the car to park itself. The same sensors that are used for Side View Assist can help detect the open spot, and then commands are issued to the steering and gas to maneuver the car into position.

- Because twice the number of accidents happen at night compared to the daytime, one of the most exciting driver assist features is enhanced Night Vision. This system works by combining infrared light and video analysis. Infrared emitters up front send out signals that are read by the infrared video camera. A special screen on the dashboard shows the scene ahead. Using software that recognizes pedestrians, the Night Vision system can also brightly illuminate people so that they can be seen and avoided.

- With the exception of old-fashioned cruise control, all of these behaviors rely on vision—or a sensory capability that functions like vision, such as radar or infrared imagining. Engineer Ernst Dickmanns was among the first to build a robust, working vision system for robotic cars. In 1986, Dickmanns and his team built a fully autonomous robot car that drove in tests on empty streets in Germany. With participation from the automobile industry, by 1995 vehicles equipped with his dynamic vision system were traveling safely for 1,000 miles at speeds up to 100 miles per hour on public highways.

The DARPA Grand Challenge

- Combining navigation with autonomous driving in robotic cars was the primary challenge put forward in 2004 by the Defense Advanced Research Projects Agency (DARPA) of the U.S. Department of Defense. They ran a robot competition called the DARPA Grand Challenge, which was for fully autonomous robotic ground vehicles. To succeed, a robot had to travel 150 miles, off and on the road, between Las Vegas and Los Angeles.

- What made the DARPA Grand Challenge more difficult than the work already done by Dickmanns's team was all of the off-road driving. There were fewer regularities: no signs and no cars to follow. Off-road driving is less structured than on-road driving. Even though 15 robots started, not a single robotic vehicle finished the 150-mile course.

- Some vehicles were able to navigate to the GPS waypoints, but those robots tended to do a bad job of detecting objects along the path. Other vehicles were better at sensing objects, but they weren't good at navigating to the waypoints. It was clear that a short-range system like Dickmanns's dynamic vision needed to be combined with a longer-range navigation system.

- DARPA decided to hold a similar challenge the next year, in 2005. Given all that had been learned in 2004, this turned out to be a brilliant decision. In 2005, 23 robots started the 132-mile off-road

Although the DARPA Grand Challenge in 2004 yielded no winners, it did spur innovation in the field; five vehicles finished the course in the next year's competition.

course in the desert of Nevada, and 5 autonomous robotic vehicles finished the course. Because of this huge and positive turnaround, many consider the DARPA challenge of 2005 to be a watershed moment in robotics.

- Perhaps the most revealing result in 2005 was that the winning robot, **Stanley**, a modified Volkswagen Touareg, hadn't even competed the year before. Stanley was the brainchild of the Stanford Racing Team, which was led by Sebastian Thrun, then the director of the Stanford Artificial Intelligence Laboratory.

- Stanley's team's approach in a nutshell was to treat autonomous navigation as a software problem. The strategy was that all of the failures of 2004 could be solved by building a better control architecture. The design of a three-module controller, combined with estimating uncertainty and **machine learning**, were the keys to the game that allowed Stanley to finish the 132-mile off-road course and win the race.

- Thrun and many members of the Stanford Racing Team took what they had learned with Stanley to Google to build the Google driverless car.

Driverless Cars

- Nearly all automobile manufacturers have worked on autonomous, driverless cars. They have been helped by components manufacturers, such as Bosch, that are making robotic systems that can be deployed on any vehicles. Like Google's driverless system, Bosch's system uses LIDAR, video and radar, to create a dynamic map of the world.

- Part of the technology transferred from Stanley to the Google driverless car was its drivability map. Google's testing engineers would ride in the passenger seat, essentially looking at the Google car's version of Stanley's drivability map.

- One of the benefits of driving on roads, as opposed to off-road driving, is that roads offer more regularities in the world, and more structure, from lines to signs and curbs. Google also has mapped roads, so the robotic car doesn't have to start from scratch the way Stanley had to in the desert.

- What makes the world of roads more unstructured and, therefore, more challenging is that in suburban and urban settings, roads can be packed with irregular traffic—not only cars and trucks of all sizes, stopping and starting at unexpected times and places, but also dogs, pedestrians, skateboarders, and bicyclists. And their positions are constantly changing.

- States in the United States began to offer driverless cars the right to operate in 2012, and testing began in the United Kingdom, Singapore, and other countries. One of the great things about these vehicular robots is that this technology can be applied to trains, trucks, and buses as well. So, the potential is to completely overhaul our transportation networks that carry people and goods.

Benefits of Driverless Cars

- For all vehicles, an immediate benefit that we see, even with driver-assist functions in a semiautonomous vehicle, is safety. While the number of deaths from automobile accidents continues to trend downward as we've added safety features such as seat belts, air bags, antilock brakes, and adaptive cruise control, tens of thousands of people still die every year from vehicle-related accidents, and millions more are injured.

- The promise of driverless cars, trucks, and buses with even more automation is that they would reduce those deaths and injuries much further. But what about cases where some sort of collision remains unavoidable? For those cases, rules about how to have a collision can also be programmed into the world model of the vehicle.

A Brief History of Driverless Cars

1986	A fully autonomous robot car begins test-drives on empty streets in Germany.
1994	VaMP and VITA-2, driverless cars, operate safely in traffic for more than 600 miles in Germany.
1995	Adaptive cruise control for cars, using lasers, is introduced by Mitsubishi.
2005	Stanley, a robotic car built by a team from Stanford and Volkswagen, wins the DARPA Grand Challenge by being one of five vehicles to autonomously navigate a 132-mile off-road course through the desert.
2007	Boss, a modified Chevy Tahoe built by Carnegie Mellon and General Motors, wins the DARPA Urban Challenge, navigating a 60-mile course while obeying all traffic laws of California, including avoiding pedestrians.
2012	Nevada and Florida become the first U.S. states to permit testing of autonomous vehicles on ordinary roads; Michigan and California follow in 2013.
2015	The DARPA Robotics Challenge focuses on robots for disaster response.

- In addition to safety, driverless cars could increase the independence and mobility of people who, for a variety of reasons, are unable to drive a car themselves. For example, as we age, what often makes assisted living imperative is reduced mobility—not being able to get to the grocery store or a doctor's appointment.

- One of the unexpected consequences of having driverless cars is that we will be able to put many more cars on the road. By some estimates, only five percent of a crowded road is occupied by vehicles. Lanes are much wider than the width of the vehicles, and following distances are kept larger than physically required to compensate for the slow reaction times of human drivers.

- By using robotic sensors and communication to pack moving cars together more efficiently, we could eventually double or triple the capacity of our existing roads. As the human population expands, this would save countries billions of dollars on unneeded road expansion and would keep land available for farming, housing, wildlife, and other purposes.

- But with robotic vehicles, the opportunities go far beyond traffic lights. Stop-and-go traffic of all kinds is the worst for mileage, for two reasons: If you are stopped, then you aren't going anywhere, and when you accelerate, you use more gas than you do when you can simply cruise at constant velocity.

- If the transportation network is tracking all vehicles on the roads, and pedestrian traffic, that information could be used to calculate and send electronic signals to each vehicle about optimum speed and position, to minimize the time that the traffic on average is stopped. Congestion levels that currently bring traffic to a stop could be managed more efficiently so that traffic keeps moving.

Important Terms

machine learning: Computer programs written to make adjustments to their code, with or without direct feedback from a human, in order to improve performance of the code itself or the robot that the code controls.

Stanley: A fully autonomous car that won the 2005 DARPA Grand Challenge, created by Stanford Racing Team.

Suggested Reading

Dickmann, Appendrot, and Brenk, "How We Gave Sight to the Mercedes Robotic Car."

Thrun, Burgard, and Fox, *Probabilistic Robotics*, chap. 1.

Thrun, et al, "Stanley."

Other Resources

"Cars That Think." http://spectrum.ieee.org/blog/cars-that-think. This is the best blog on robotic cars, created and curated by *IEEE Spectrum*, an engineering magazine.

The Great Robot Race. This NOVA program, available to watch online (http://www.pbs.org/wgbh/nova/darpa/program.html), gets you behind the scenes of the teams and under the hoods of the robots at the Grand Challenge of 2005.

Questions to Consider

1. If you want to build the best robotic car possible, which kind of architecture would be better for the controller: behavior-based or model-based?

2. In 2014, a robotic car competition sponsored by Hyundai in South Korea found that the cars did quite well on a dry, sunny day. But when the road was wet and the weather was partly cloudy, the robotic cars had far more problems, including several missed turns and other maneuvers that they had handled easily during the nice weather of the previous day. What kind of sensors would you suggest be added or improved to address the problems caused by bad weather conditions?

Flying Robots: From Autopilots to Drones
Lecture 15

Anytime that we build robots to work outside, the challenges mount. But the challenges are redoubled for aerial robots. The physical demands of flying impact every system on the robot. Sensors and actuators have to support full mobility in three dimensions; energy requirements are high, while weight has to be kept as low as possible, and the entire control loop has to function much faster. However, the challenges of aerial mobility are what made aviation an early and consistent adopter of robotic technologies.

Aerial Robots

- Piloted aircraft often include robotic assist technologies that operate the aircraft autonomously when the pilot turns them on. The best example is the autopilot found on commercial aircraft. While we don't normally think of a plane flown by autopilot as an autonomous robot, it is, at least when the pilots aren't at the controls. An autopilot is an embedded robotic system, with autonomous self-control using sensors to guide movements.

- All the self-control technology assisting human pilots is equally available for unmanned aerial vehicles, too. And unlike aircraft that carry human pilots, unmanned aerial robots can be almost any size, from vehicles as big as commercial passenger aircraft to micro-vehicles the size of insects. And they can even execute maneuvers too stressful for a human body, opening up new design possibilities.

- Aerial robots go by a whole host of names: drones, unmanned aircraft, unmanned aerial vehicles, unpiloted aircraft, unpiloted aerial vehicle, unpiloted air system, and remotely piloted aircraft.

- Unmanned drones come in two types: fully autonomous and remotely controlled. Autopilots operate unmanned aircraft, such as many commercially available drones.

- The Walkera TALI H500 is a hexacopter drone. It has six propellers that provide the thrust for lift, to keep it aloft, and the thrust for moving forward and maneuvering. You control the TALI using a radio control console that transmits and receives signals, including video from the drone.

- The drone controls itself to remain stable. To keep the robot still takes all sorts of control. When some external force attempts to tilt the drone, it senses an acceleration and tilt using its onboard three-axis inertial sensor. Once movement is detected, the onboard controller immediately reacts by having the three actuators on the side pushed downward to increase their thrust to counter the perturbation. After this first corrective action, then the TALI H500 has to modulate the thrust on all six thrusters to damp any wobble that might have been set up.

- Drones can use active ping sensors—located on the bottom of the craft and pointed toward the ground—to measure altitude. If the distance that the ping sensor measures is not one meter, then the thrust is adjusted to either raise or lower the craft.

- Maintaining that height is an example of negative feedback control. The actual height is constantly measured and adjusted to reduce the difference between that actual height and the desired height. The feedback is the information that you get from the sensors about the actual height.

- When the TALI H500 is hovering, it is working on autopilot. It is doing the work of coordinating all six of its motors for the person controlling it. When the person steps into the control loop, he or she can control the altitude, heading, and speed.

- One of the most important autonomous features for any drone to have is landing. You want to avoid a crash landing. So, drones like the TALI H500 have built-in landing autopilot functions that help you preserve your investment.

© marekuliasz/iStock Editorial/Thinkstock.

Aerial robots are useful in many situations, enabling us to go where it might not be safe or feasible for humans to go.

- Because the TALI H500 has autopilot functions but also allows a person to control it, it's a remote control drone with some autonomous functions. Most drones are this mix of remote control by a human and autonomous control by the robot. What varies among drones is the degree to which they offer autonomous autopilot functions.

- The needs to not lose the drone and to avoid destroying it are the reasons that these autonomous flight features come as the standard defaults. One of the great safety features built into the TALI H500 is that if you fly it out of communications contact, it will automatically return to its staring position. It uses GPS navigation to accomplish this.

Designing and Building Aerial Robots

- Designing and building an aerial robot all begins with the basics of flight: getting airborne, staying airborne, and landing gently.

- The bodies of robotic aircraft need to be lightweight to minimize the energy needed to keep the drone aloft. Bodies need to be aerodynamically shaped to help the actuators generate lift and reduce drag. If engine power is lost, bodies also need to be able to glide, parachute, or otherwise survive an emergency landing.

- The actuators of drones need to be lightweight. If the actuators are actively generating lift as well as forward propulsion, then they need to have a high power-to-weight ratio.

- One of the ways to get a motor to be small and powerful is to trade off torque for power. A drone's motors are fast with low torque. The motors of the Parrot AR.Drone 2.0 spin extremely fast but have virtually no torque at low speeds. They are built to operate at very high velocities.

- The sensors of drones are of two main types. Proprioceptive sensors monitor the state of the robot, and they must do so with high speeds of reaction and with good accuracy. Navigational sensors must detect current position on a map, compass heading, airspeed, and altitude. You can fly many drones using an onboard GPS navigation system.

- Energy supply needs to be of high density because the cost of flying is so high. The primary cost is to stay aloft, to generate lift. If this can be done for free, as in a balloon or dirigible, the energy cost of flight is for forward propulsion only. But lighter-than-air craft suffer from needing a large size to carry gases. The large size makes them slow.

- Any rotary winged aircraft, such as a helicopter, quadcopter, or hexicopter, is in big trouble. The Parrot AR.Drone has a pretty lightweight battery. But the trade-off is between the weight of the robot and the energy capacity of the battery: It only carries enough energy to power the drone for about 12 minutes of flight time. The heavier Walkera TALI H500 carries a larger battery for 25 minutes of flight time.

- You can also get replacement batteries that double the flight time. But there are always trade-offs. A battery with more capacity usually means a heavier battery. A heavier drone means that the motors have to spin faster in order to keep the quadcopter hovering. Add a big enough battery to spin the motor for a couple of hours and the drone can't even take off.

- Control of aerial robots includes behavior-based systems that offer rapid, reflexive corrections of flight systems using negative feedback control to maintain proper flight attitude or to engage maneuvers. Rapid control is particularly important in inherently unstable aircraft, such as a rotary-wing vehicle or an air vehicle traveling at high speed.

- Also, certain environments produce wind turbulence that is very challenging for microdrones in particular. Most microdrones have operational limits on wind speed. Model-based control algorithms are needed for navigation to compute heading to the next waypoint, speed over ground, and altitude above ground.

- The challenges of controlling a robot in the air are exemplified by a simple behavior: object avoidance. Object avoidance is very easy to program in land-based robots. For example, Roomba uses bumpers with infrared proximity detectors, which are active sensors that emit electromagnetic energy in the infrared range. The sensor also has a detector that is reading the intensity of the reflection of the infrared energy off an object.

- Object avoidance with microdrones like the Parrot AR.Drone 2.0 won't work with infrared sensors. You have to be close to have them work, and then you have to be moving slowly enough to be able to maneuver out of the way. Drones need long-range sensors.

- Large drones can carry radar, but microdrones cannot. Two developments may soon help microdrones: the creation of small and lightweight radar for cars and the development of object-detection algorithms (from robots like the Google driverless car) that use video. Because many microdrones have video, this would be great. The problem is that to detect objects quickly with video, you need a fast computer, and fast computers tend to weigh more than slow ones.

- One of the fundamental challenges of robotics—navigation—is easier outside and in the air than it is underwater. That's because GPS can be used outside, and at heights above the tree line or the highest buildings, few obstacles exist.

- Aerial robots face different challenges depending on their size. Aircraft-sized drones are large enough to carry sophisticated sensors and high-density energy supplies sufficient for long voyages. Some drones can stay aloft for nearly 24 hours. Because large drones operate primarily in unrestricted and high-altitude airspaces, object avoidance involves primarily other aircraft, and radar systems on board can be used in conjunction with land-based air traffic control.

The Future of Aerial Robots

- One of the long-standing challenges in flight systems is carrying sufficient energy to power your flight. Fixed-wing aircraft have an advantage over rotary-winged aircraft in that they generate lift by propelling themselves forward. Once you get airborne, you can take advantage of fixed wings by getting as energy efficient as you can be: gliding.

- Aerial robots can be built as gliders. Gliders move forward by sinking. The challenge is to catch an updraft and gain height. The best gliders in the animal kingdom do just this, using the rising air off of a warming mountain face to give them the height to glide over.

- Birds can also use a technique called **dynamic soaring**, which involves gliding quickly through regions of increasing wind velocity. Dynamic soaring allows birds to travel for hundreds of miles without flapping and at speeds approaching 30 miles per hour.

- Radio-controlled gliders have been able to use this technique to go even faster—much faster. Radio-controlled gliders flying in a loop over mountainous terrain have been steadily breaking one another's records, with a new record of more than 500 miles per hour reached using a Kinetic 130DP glider in 2014.

- With such results in mind, a researcher at Woods Hole Oceanographic Institution, Philip Richardson, has calculated that glider unmanned aerial vehicles may be able to hit top speeds close to ten times the speed of the winds they are riding, at least for small unmanned aerial vehicles in slower winds.

- Hybrids that use both fixed wings and rotary wings are also possible. This approach has been demonstrated in the Makani Power kite, which was the first fully autonomous wind-harvesting robot, demonstrated in 2013 and bought that same year by Google.

- Still, the biggest frontier is the small flying robots. Compared to larger drones, microdrones operate near the ground, in complex urban environments, and even inside buildings. Being so near the ground, a microdrone encounters a host of obstacles, including buildings, wires, and trees.

- Walkera makes a microdrone called the Scout X4 that pushes the limit of real-time navigation and localization. Using your smartphone, you pair wirelessly with the Scout X4, and then you can get it to follow you and film you.

- Microdrones like the Scout X4 represent the new wave of aerial robots. They are small and affordable, and because of their built-in autonomous functions, they are very easy to pilot and navigate. Drones with cameras on board are exciting for a host of potential uses, including search and rescue, hazardous environment exploration, high-resolution weather and climate mapping, and traffic monitoring.

Important Term

dynamic soaring: A type of gliding flight in which the animal or vehicle gains velocity and height by harvesting energy from steep wind gradients located near surfaces.

Suggested Reading

Jarnot, "History."

Sperry, 1931, Wireless-controlled aerial torpedo, U.S. Patent 1,792, 937.

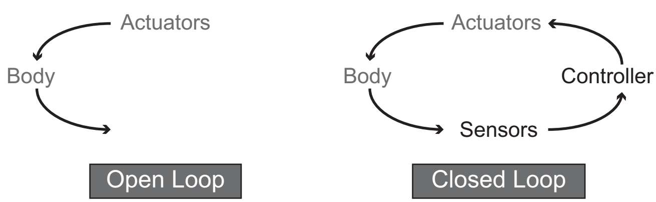
Zaloga, *Unmanned Aerial Vehicles*.

Other Resources

Federal Aviation Administration, Unmanned Aircraft Systems: https://www.faa.gov/uas/. As rules and regulations for unmanned aircraft of all types evolve rapidly, this is the epicenter of change.

Question to Consider

1. When we remotely control a robot, that robot is said to have an open loop; when it is autonomous, it is said to have a closed loop.



Consider the automatic pilot invented by Elmer Sperry. If you are flying a plane and it has an autopilot, is that plane a closed-loop robot?

Underwater Robots That Hover and Glide

Lecture 16

Nowhere are the challenges of working with robots in the wild more difficult than on and in the water. To work in a river, lake, or ocean, every aspect of the robot has to be redesigned. The challenges are especially severe if you aim to have your mobile robot go underwater, where we call it an autonomous underwater vehicle (AUV). The body, energy supply, actuators, sensors, controller, and communications systems of an AUV all operate differently than they do on land or in the air.

Water and Electronics

- Water and electronics do not mix. That's because water conducts current, like a wire, creating pathways between components that were not meant to have direct connections. The result is what we call short-circuiting.

- Always double-check your robot for leaks at the launch site, just prior to putting it in the water. Even when we build robots for swimming on the surface of the water, we need to worry about leaks.

Sensors in Water

- The sensors we use in the air and on land often work poorly or not at all in water. While radio waves work well in the air, they do not work well underwater. The radio frequencies we use in air have wavelengths from about 10 centimeters to 10 meters—a size that is not absorbed by the atmosphere. That's why we use them. But water quickly attenuates those same radio waves underwater.

- Radio waves are part of the electromagnetic spectrum, which also includes microwaves, infrared waves, visible light waves, each of which are shorter than radio waves. Visible light also has problems penetrating water. Oceanographers refer to the region where visible light penetrates sufficient for photosynthesis as the photic zone, with a cutoff point defined as where less than one percent of visible light from the surface can penetrate. The distance where more than 99 percent does not get through occurs on average at distances of only 200 meters (660 feet).

- In order to propagate long distances in the water, electromagnetic energy needs a wavelength of more than 100 kilometers. These long waves oscillate at extremely low frequencies, and they have been used for communications with submarines. But the low frequencies mean that messages have to be short. And they can only be one-way, from station to submarine, because a huge antenna is needed to create the giant waves, and even a submarine is too small to carry one that is large enough.

- A different kind of wave that does travel well underwater is a pressure, or acoustic, wave. Submarines can communicate while underwater using acoustic transmission systems with a speaker and a hydrophone. But while these pinger systems work well over short distances, such as a few kilometers, they suffer from noise created by reflections and attenuation created by scatter off of particles in the water. Also, the amount of information that you can transmit with a pinger is very small, because all that you can vary to represent that information is the pattern of the pulse of sound—the rhythm.

- An underwater acoustic modem system is the most-often-used form of communication in underwater marine robotics. The most sophisticated systems offer short-range communication over the range of a kilometer in shallow-water situations. The rate of data communication is up to 30 kilobits per second, which is slow compared to wired or wireless transmission rates on land.

- Although acoustic modems are the standard way to communicate wirelessly underwater, using a wire, a communications tether, also is still very common. As long as you don't have to have your robot swim very far away, a communications tether can give you great control over your robot. Underwater robots with a tether are much more likely to have remote control and/or an energy supply coming through the tether as well.

- Local sensors, operating over short distances, offer a creative way around the limitations of radio waves, acoustic signals, or tethers. A solution has been prototyped by European roboticists in a project called Collective Cognitive Robotics (CoCoRo). The idea is to use blue light to flash communications underwater. The distance of communication is achieved by a bucket brigade approach: Use a swarm of robots arranged in a linear chain. A signal at one end of the chain propagates along the chain.

- Even though the CoCoRo researchers used blue light for communication, the signal from light degrades very quickly, just like sound waves or radio waves. So, in fact, their real innovation is to show that a swarm of robots arranged in a line can extend communications to new lengths. This swarm-based collective communication system is still in the earliest stages of development.

- A common replacement for vision underwater is sonar. This is another use of sound underwater. With active sonar systems, the sensor sends out a sound pulse or train of pulses and then listens for the return, the echo. Depending on how long it takes the pulse to return and how strong the signal is, the robot can use sonar to detect objects underwater and to map terrain.

- For example, the PAIV robot, jointly developed by Subsea 7 and SeeByte, has a three-dimensional sonar that looks forward and a 2-D profiling sonar that looks down. PAIV has a navigation system that includes a compass and inertial guidance sensors. PAIV also has an acoustic communication system and a camera and lights to gather pictures.

- The key to autonomous movement is to link information from sensors to the control of actuators. Because sensor-guided movement is at the heart of autonomous behavior, the choice of sensors is a very important part of design.

- Sensors underwater face a number of challenges because of the physical properties of water and operating in the wild. Video systems, for example, only work if the water is clear and light is available. Go into murky waters of a delta or into the deep and you are in trouble if you are relying on vision.

- The mission concept for PAIV is to operate as an autonomous inspection system. For example, we humans have sunk all kinds of electric and communications cables in the oceans and lakes. These cables need constant maintenance that can be directly assessed by close-up inspection.

- PAIV locates the cable riser using sonar. Then, it moves along the cable, delivering video images of the cables and floats for a topside human to review. Sonar is clearly the primary sensor system for sensor-guided movements.

- PAIV can navigate the bottom to inspect pipelines. One of the virtues of autonomous robots like PAIV is that because they lack tethers, they can move in and around complex three-dimensional structures without fear of tangling.

Nautical Robot Milestones

1868	The torpedo, the first autonomous underwater vehicle (AUV), is invented by Robert Whitehead.
2009	RU-27, also known as Scarlet Knight, becomes the first AUV to cross the Atlantic Ocean underwater, a journey that took 221 days.
2013	The Wave Glider robot Benjamin from Liquid Robotics sets a world record of 7,939 nautical miles, the longest journey by autonomous surface vessel.

Bodies of Aquatic Robots

- The bodies of aquatic robots are often very distinctive. The first autonomous underwater robot was a torpedo. But while torpedoes are weapons, the physics that makes them so efficient has inspired a whole body type among modern autonomous underwater vehicles (AUVs), many used for other purposes.

- Bluefin Robotics has a torpedo-shaped AUV called the Bluefin 21. The tail doesn't have the protruding control surfaces of a standard Whitehead torpedo: no rudder and no elevator. That sleek design reduces the chances of the tail getting hung up on cables, seaweed, or the like.

- The torpedo shape tells us a lot about the physical challenges of moving in water. A torpedo is an example of a streamlined body. A streamline is a visualization technique to see how smoothly the water moves around a body as the body moves through the water.

The physics that make torpedoes efficient underwater make the body shape ideal for other autonomous underwater vehicles.

Shapes like torpedoes that are tapered in the front and the back make for very smooth traveling through the water. They are said to be streamlined, which means more energy efficient to move.

- In contrast, an unstreamlined shape would have blunt ends and be boxy in shape. This means that you can tell right away when you look at an underwater robot if it is built to cruise or to hover. Cruisers are streamlined, while hoverers are boxlike.

- For its torpedo-shaped robots, Bluefin has designed the tail cone as a "**ducted thruster**" that is articulated. The connection of the tail cone to the hull is like a ball-and-socket joint. A swiveling joint like this gives two degrees of freedom for the adjustments of the thrust. The tail cone can pitch, yaw, or move in some combination of the two. The AUV steers by rotating the tail cone to direct its thrust in different directions.

Energy Supply and Actuators of Aquatic Robots

- The energy supply and actuators of an AUV can be very different as well. Bluefin's torpedo AUV works using batteries, and they have a mission life of about 24 hours. That's great for mapping near the shore, but there are times when you want the AUV to have missions that last longer. For example, extensive monitoring of sea surface temperatures, which is vital for predicting El Niño events, requires time on station that we would measure in weeks or months.

- One way to extend mission time is to reduce the amount of energy that you spend on propulsion. You might even decide to just float along with currents or tides, but then you aren't an AUV because you aren't self-propelled. You'd be a sensor buoy that floated along and collected data.

- A very clever way to be self-propelled but to do so with minimal cost is to glide. There is a whole class of AUVs called gliders, and they look like torpedoes with wings, like Bluefin's Spray Glider. Gliders are so efficient—take such little energy to operate—that they can run on a single mission for more than six months.

- Gliders work by changing their density relative to the water. An object denser than water, what we call negatively buoyant, will sink; an object less dense than water, what we call positively buoyant, will float. A glider at the surface of the water makes itself negatively buoyant so that it sinks and uses that sinking motion to allow it to move forward, thanks to its wings.

- Most of the energy of propulsion in a glider comes from the gravitational effect called buoyancy, so we talk about these AUVs having a buoyancy engine. You can think of a buoyancy engine as sinking a ship by flooding one of its compartments and then floating a ship by bailing out that same compartment.

- Gliders are the long-distance, low-power champions on the opposite end of the AUV spectrum from the hovering AUVs like the PAIV robot. Hovering AUVs are vehicles designed to not go very far and to run for just a few hours.

- Hovering AUVs perform close-up inspection of hulls of ships, submerged piers, or rocky bottoms. To hover and hold station, they maximize something that gliders and torpedoes are very bad at: maneuverability.

- The champion of long-distance and long-duration robots is Wave Glider, which was built by Liquid Robotics and looks like a surfboard with a few antennae. Wave Glider is an example of how to solve two of the big problems that the aquatic environment presents to robotics. First, its wave-based power supply is endless, and because of this, so is its effective time on station conducting its mission. Energy is simply not a problem. Second, because part of Wave Glider is always at the surface, it can always communicate using radio waves and navigate using GPS.

Important Term

ducted thruster: A propeller-based actuator used in torpedo-shaped underwater robots to both propel and steer the vehicle.

Suggested Reading

Bohm and Jensen, *Build Your Own Underwater Robot, and Other Wet Projects*.

Bureau of Ordnance, Department of Navy, *The Whitehead Torpedo*.

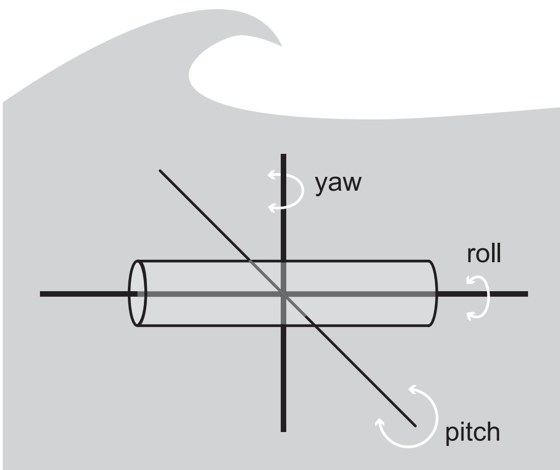
Long, "Biomimetics."

Other Resources

Get started on open-source underwater robots: http://openrov.com/.

Questions to Consider

1. One of the challenges of working with robots underwater is how to control motion in three dimensions. How can we simplify the control of our robot underwater?



a. Use a three-axis accelerometer, compass, and inclinometer to determine and control the rotation of the robot in three rotational dimensions.

b. Use wings and other stabilizers to prevent or reduce rotation in the roll dimension while adding control surfaces to allow for pitch and yaw.

c. Let the robot freely roll, pitch, and yaw as forces dictate.

2. Water is an extremely challenging environment for robots. What makes life difficult for a marine robot and its attending roboticists?

a. For humans working either on the shore or in a surface vessel, any robot below the surface is difficult to keep track of.

b. Any leak in the robot's hull will short-circuit the onboard electronics.

c. The enormous pressure of water at depth can easily crush most robots.

Space Robots in Orbit and on Other Worlds

Lecture 17

Space is the final frontier, and robots are the vanguard of our human explorations of the universe. With space missions, we push the extreme of working remotely with and through our robots. Robots in space include rockets, spacecraft, shuttles, satellites, probes, landers, and rovers. On board shuttles or space stations, we also find manipulator robots and humanoids. Much of the work in space robotics has focused on low-Earth orbit and the surfaces of other planets.

Orbital Robots

- The most exciting project in orbital robotics is the International Space Station, where humanoid robots are being developed to help humans work inside the space station. Robonaut 2 (R2) arrived on the International Space Station in 2011. Like Baxter, R2 is built to interact with humans, safely. It consists of a head and a torso, with two arms that function independently.

- R2 can be controlled by a human operator in the space station wearing a glove and headset or remotely by ground control. R2 can also be programmed to do some tasks autonomously. Many of its low-level operations, such as grasping control, are done autonomously, even if, for example, the object to be grasped is determined by its human operator.

- How can R2 help humans in the challenging work conditions of microgravity? Inside the space station, R2 could set up an area for the manipulations that only humans can do. During an operation, R2 could assist by providing tools or holding something in place. And then it could clean up afterward.

- NASA plans for R2 to work on the outside of the station, too, where conditions are extremely harsh. At the Earth's distance from the Sun, such as the International Space Station, temperatures in sunlight can rise to more than 120 degrees Celsius or 248 degrees Fahrenheit, well past boiling (for water). But on the side of the craft not facing the Sun, heat radiates away so quickly that temperatures on the dark side can sink to −100 degrees Celsius or −148 degrees Fahrenheit, well below freezing.

- Because it takes about 90 minutes for the space station to orbit the Earth, if R2 were working outside for a few hours, it could easily encounter those extreme temperatures. Without thermal control, R2 would likely suffer a number of problems, including camera and battery malfunctions.

- R2 will need to be outfitted with an array of passive thermal control (such as insulation and external coatings) and active thermal control systems (such as electric-powered heaters and coolers).

- Once thermally hardened for work on space walks, R2 will be able to replace humans on these dangerous external tasks. One of the reasons to design R2 as a humanoid is so that it can use the same tools, handholds, hatchways, and equipment as humans.

- Part of R2's great ability to reach and grasp and interact safely comes from the touch sensors that it has on its hands. These sensors provide R2 with feedback so that it can hold onto very fragile objects or apply stronger force when needed.

- R2 is also strong. It can lift a 20-pound weight, and it can do things that humans can't, such as hold that 20-pound weight with arm outstretched for long periods of time.

- R2 has been designed to be compliant. It knows when it has encountered an unexpected object and will shut down immediately. This makes R2 safe to work with. R2 also has a vision system. It can pick up an envelope and examine it. This function involves rudimentary object identification.

- Even though R2 only works inside the space station at the moment, other robots are at work outside. They are all versions of armlike robots that are similar to the pick-and-place robots that work in factories. The first of these outside, outer space robotic arms was the so-called Canadarm. Built for the space shuttles by the Canadian Space Agency, the remote manipulator system was teleoperated by crew members through a computer interface.

- The remote manipulator system was used to unpack the cargo bay of the shuttle or to grab and manipulate objects already in orbit. The joints were key to the functioning of the remote manipulator system. It had a wrist, elbow, and shoulder, each joint having three, one, and two degrees of freedom, respectively.

- The remote manipulator system turned out to be a workhorse for the whole shuttle program. NASA quickly figured out that they could use the remote manipulator system to move astronauts around on space walks. The remote manipulator system proved to be incredibly useful, and it spawned the creation of the Canadarm2 for use on the International Space Station.

- Having built the International Space Station, one of the main jobs of Canadarm2 is to grab, dock, and unload unmanned spacecraft that deliver vital supplies. Canadarm2 has the ability to perform finer-scale manipulations by adding onto its end a two-armed robot called Dextre, which is loaded with many degrees of freedom, allowing it to swivel, pitch, yaw, and grasp.

Extraterrestrial Robots

1970	Lunokhod 1, a Soviet remote-controlled lunar rover, becomes the first unmanned vehicle to explore the surface of an extraterrestrial planetary body.
1997	Sojourner, the semiautonomous rover of NASA's Pathfinder mission, becomes the first robot to land and work on Mars.
2004	Spirit and Opportunity, semiautonomous NASA rovers and geological explorers, land on Mars.
2012	The Curiosity rover, a robotic geological explorer, lands on Mars.

- Canadarm2, which is a robotic manipulator, can use, as its end effector, another robotic manipulator, Dextre. Hanging onto and moving around another larger robot is made possible because of the near weightlessness of being in orbit.

- Dextre is used, along with Canadarm2, to change batteries and replace cameras, jobs formerly done by astronauts in long, tiring, and dangerous space walks. Dextre is built specifically to work on the International Space Station. Dextre has a toolkit from which it can select different end effectors. Different graspers are built for different jobs.

- Robonaut2, Canadarm, Canadarm2, and Dextre are all designed to help humans grasp and manipulate the world. Whether located on a shuttle, crawling around the outside of the space station, or hanging out inside, they all operate as nearby extensions of human arms and hands.

Interplanetary Robots

- In addition to orbital robotics, the other major field of space robotics is that of interplanetary robotics, whose stars are the planetary rovers. Unlike robots in orbit around the Earth, robots on other planets are separated from us by both long distances and long lag times in communications.

- But the robotics challenge is not just about how to communicate with your robots but how to control them over long distances, for purposes of virtual presence and virtual agency. Presence and agency are essential for scientists and explorers and for being able to adjust the original mission as new information reveals faulty assumptions, as the robots have problems, or as increased understanding of the planet provides opportunity for new tasks and goals to emerge.

- Twin robot rovers, Spirit and Opportunity, were built to study the complex geology of Mars. Their primary goal was to understand how the activity of water on Mars influenced the planet's environment over time. Thus, the Mars Exploration Rovers are part of NASA's long-term mission to achieve four scientific goals that extend across multiple missions and into the future:
    - determine if life ever arose on Mars,
    - characterize the climate on Mars,
    - characterize the geology of Mars, and
    - prepare for humans to travel to and explore Mars.

- The incredible success of this mission has set a new standard in space robotics for planetary science and exploration.

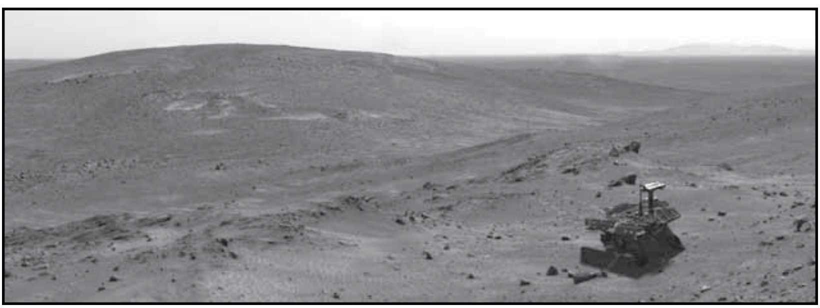
- Because of their scientific tasks, Spirit and Opportunity are often characterized as robot geologists. Four teams on Earth guide the geological work, which falls into four categories: mineralogy and geochemistry, soils and rocks, geology, and atmosphere.

- Early in the mission, these four scientific teams met daily to discuss the latest information from the robots and then created a new plan for the following day. Then, those plans were uploaded, via radio, to each rover. The rovers would then autonomously enact the plans and report back.

- What's neat about this process is that scientists at mission control were reprogramming the autonomous robotic rovers every day. Contrast this with real-time remote control, where, instead of a plan enacted in computer code, scientists have the plan in their minds, and they directly control direction, speed, and any instruments by inputting commands manually.

- Because of the communication delays that result from real-time remote control, NASA decided early on in its planetary exploration programs to avoid any attempts at real-time remote control. For control of these robots so distant in time and space, that leaves autonomy, with the rover operating on its own according to programming on board its computers.

- Autonomy for planetary robotics comes in two types. One type is fixed autonomy, where the programming that controls the interactions of the sensors and the actuators is fixed. The other type NASA calls batch processing, but it is better described as reprogrammable autonomy.



Robot geologists Spirit and Opportunity are not remotely controlled in real time; the robots autonomously enact the plans uploaded by NASA.

- Reprogrammable autonomy is the type employed in Spirit, Opportunity, and Curiosity rovers on Mars. Every day, a new program, a new plan, is uploaded to each rover.

- The trade-off with reprogrammable autonomy is one of adaptability versus time. If you can reprogram your robot every day, then every day it can do something new; it can adapt to the information it discovers and any unforeseen changes in how it is operating. However, it takes time to reprogram, to evaluate what to do next, and to communicate those plans. So, robots with reprogrammable autonomy, compared to those with fixed autonomy, may need more time to accomplish a given task or mission.

- Interplanetary communications, which happen in both directions, to and from Mars, to and from the robotic rovers, are delayed, of short duration, and of small size.

- The robot's energy supply powers the radio transmitter and receiver. All of the sensors and instruments on the rovers also require energy, so their use has to be carefully planned and scheduled. With their cameras, the rovers have created an incredible catalogue of images of Mars. Three scientific sensor systems, all of which are **spectrometer** instruments, look for water, iron, and chemical composition.

- Making it possible for these instruments to do their jobs is the arm, or what NASA calls the instrument deployment device. The rover's arm has five degrees of freedom. The joints of the shoulder, elbow, and wrist are all revolute joints that rotate. The shoulder has two joints oriented perpendicular to each other. The elbow has a single joint. The wrist has two joints. As the wrist rotates, it brings different instruments on the hand to bear on the soil.

- The Mars Exploration Rovers have met and exceeded NASA's expectations. Originally designed for a mission of 90 days, Spirit sent its last communication in 2010, 6 years after it began operations and traveled nearly 5 miles. Opportunity operated for more than 10 years and logged more than 25 miles.

- Scientific achievements of the mission have been many. Spirit took movies of swirling dust devils, found circumstantial evidence of water by detecting sulfate minerals, and found silica deposits that indicate a much wetter past. Opportunity found evidence for water over a wider range of terrain, indicating a past with large bodies of standing water.

Activity

Canadarm2 simulator:

www.asc-csa.gc.ca/eng/multimedia/games/canadarm2/.

Made available by the Canadian Space Agency, this is the official simulator game of the Canadarm2. One of the challenges is to use the three different cameras to create a three-dimensional understanding of the space station in your head. Your mission is twofold: to install a module and then to transport an astronaut. Good luck!

Important Term

spectrometer: Any of a large class of instruments designed to measure how the intensity of a physical property varies across a range of frequencies, energies, or masses.

Suggested Reading

Brooks, *Flesh and Machines*, chap. 3.

Clancey, *Working on Mars*, chaps. 4 and 6.

Other Resources

The official website for the robots Spirit and Opportunity: http://mars.jpl.nasa.gov/mer/home/.

The official website for the robot Curiosity: http://mars.jpl.nasa.gov/msl/.

Question to Consider

1. When we decide that a planetary rover, such as the Soviet Union's Lunokhod, is going to be remotely controlled, we are opting for the same kind of human-in-the-loop architecture that we still see in our partially robotic planes and automobiles. What are the real engineering advantages to these open-loop systems?

a. The controller, the human, is more complex and therefore more capable than any set of algorithms that we can code on a computer.

b. If the vehicle contains sensors, then humans can be remotely present, what some people call "telepresent," in the world in which the robot sits.

c. A human in the control loop allows for careful checks on the status of a valuable robot.

Why Military Robots Are Different

Lecture 18

Military robots lead the pack in terms of innovations in robotics, because they have to be customized to operate in the air, on and under the sea, and even in space. Many military robots have been self-guided weapons; they were designed to destroy and kill. Therefore, there are tensions between semiautonomous military robots and the need for humans to oversee safety and control. There are trade-offs to consider between speed of action and complexity of plan, and the identification of friend or foe becomes far more important in military situations than in any other area of robotics.

Safety and Control

- One way to maintain control over a robot—to make the robot obey you—is to maintain communications with it and give it limited autonomy to act on its own. While partial autonomy keeps control in the hands of the humans, it presents a security problem: Radio communications can be hacked.

- One robotics solution to the changing situations, such as being hacked, is called adjustable autonomy. For example, if a radio link is cut, many drones automatically fly themselves home. They monitor communications and then adjust to the changed situation and become fully autonomous.

- A second concern about maintaining safety and control resides in how decisions are made. In robotics, there are two main kinds of decision-making systems: behavior-based control and model-based control.

- Behavior-based controllers are fast, reflexive systems, such as the escape response on a Roomba vacuum cleaner. A sensor triggers a response: When it detects an object or a change in environmental conditions, that information triggers the programmed software controller to send instructions to the motors of the wheels to initiate a series of movements called an escape. If multiple sensors are triggered, the controller allows one reflex to overrule the others, a decision that it makes based on a set of priorities programmed into its code by humans.

- Model-based systems are relatively slow, deliberative systems that can weigh information from multiple sensors, calculate higher-level inferences, update maps, and make plans for actions in the short and long term. At the heart of this kind of controller is a model of the world that includes the current state of the robot in that world. As both change, the model has to be updated in order to remain accurate and useful. With a model-based system, decisions are made based on the current state of the robot and the desired future states of the robot that are required in order to achieve the explicit goals programmed into its code by humans.

- The tension between these two systems can be characterized as a trade-off between speed of action and complexity of plan. In a military situation, we often need both. Many robots combine fast and slow autonomous control architectures.

- Both types of control architectures rely on the quality and accuracy of the information provided by sensors and by the models of the world. But nothing is ever certain, and the field of **probabilistic robotics** recognizes that explicitly.

- Probabilistic robotics grapples with uncertainty in mathematical terms, building and updating models of the world and the robot that create robust behavioral choices in the face of that uncertainty. The autonomous robot makes decisions based on what it knows and what it doesn't know, and its degree of uncertainty about both.

- Probabilistic robotics has helped build the very successful and safe autonomous robots known as self-driving, or driverless, cars. In this area of robotics, in addition to making autonomous robots operate safely in dynamic real-world situations, we can, with sufficient investment, make them operate much more safely than humans working under similar conditions.

- In warfare, the equivalent of the rules of the road are international treaties, such as the Geneva Conventions. Protocol I is an amendment from 1977 outlawing, among other things, attacks on places of worship. This rule could be written into the programming of an unmanned aerial vehicle: Any place of worship is indicated on the robot's model of the world, its map, as an off-limits area.

- Even with this programming and a probabilistic controller that accounts for uncertainty, we could never expect a robot to operate perfectly. The robot could accidentally bomb a place of worship if the information in its world model were inaccurate. We would also consider it an accident if a missile fired by the robot intended for another target malfunctioned, for reasons that had nothing to do with the programming of the robot, and hit the place of worship. Also, there's nothing to keep an unauthorized human from reprogramming the unmanned aerial vehicle.

- One thing that we wouldn't have to worry about anytime soon is a robot with emergent self-consciousness, free will, and a sudden desire to kill humans. No one working in robotics, artificial intelligence, psychology, or neuroscience has ever created a computer program or machine that is conscious and self-aware in the way that most humans think of those experiences. While we can build robots with behavioral autonomy, their autonomy is not of the same type that we humans possess.

- And that difference in how robots and humans work is a great thing, practically speaking. That's why robots are better drivers than humans. They don't get tired, or distracted, or emotional. And that's why—if programmed correctly—robots can be better soldiers. But we have a long way to go.

Automation and Navigation

- One area in which robots are not yet capable of being better soldiers than humans is in the identification of friend or foe in nontraditional battle situations. This is a huge problem for humans, too; an occupying army faces this problem continuously.

- The general problem is called combat identification, and the military has been working on a technological solution since World War II transponders with coded signals on vehicles or individuals can be used to positively identify a friend, as can careful tracking from known starting positions. But confusion exists if a transponder is broken or tracking is interrupted. In addition, the activation of a transponder can also alert the foe to the presence of your own soldiers.

- The difficulty of automating combat identification and general concerns about maintaining safety and control are two of the reasons that the most widely deployed military robot—the Predator unmanned aerial vehicle and its descendants—is operated by a human pilot and given only partial autonomy.

- In robotics, navigation is one of the most fundamental problems that any mobile robot has to solve. You have to know where you are to get where you want to go. GPS works as a system of navigational

The Predator drone and its descendants are given only partial autonomy to minimize errors in combat identification and maintain safety and control.

beacons that can be used to calculate your position. The current GPS system is a network of 29 Earth-orbiting satellites. The robot needs the ranges and locations of four satellites to calculate its position using a process called **trilateration**.

- What makes accurate navigation particularly important to unmanned military robots like the Predator is that they can remain on remote station for more than 24 hours. If the unmanned aerial vehicle were trying to navigate by dead reckoning, errors in estimating position would increase with time and distance.

- Navigation does not present a problem for other types of military robots as long as they operate in sight, performing jobs in close proximity to the soldier in charge of them. PackBot, created and produced by iRobot, is a class of small, mobile remote-controlled robots with partial autonomy that have been customized for a variety of different tasks in close proximity to the operator.

- They are built to dispose of bombs, inspect for improvised explosives, provide situational awareness to first responders, aid search and rescue, collect video and air samples in hazard responses, and detect and localize snipers.

- In each of these tasks, PackBot increases the standoff distance between a human soldier and a potential threat. As a stand-in for the soldier, PackBot allows operators to have virtual presence and virtual agency. Virtual presence lets the operator perceive the world from the robot's point of view. Virtual agency lets the operator act in the world as the robot.

Militaries use robots to dispose of bombs so that humans aren't put in danger.

Bioinspired Robots

- The legged squad support system (LS3) is basically a robotic packhorse built in conjunction with DARPA by Boston Dynamics. This four-legged robot, which is the size of a horse, can carry 400 pounds and, most importantly, can do so over rough terrain. Because the design of LS3 is inspired by animals, it has a biomorphic body plan. LS3 is partially autonomous.

- LS3 is the descendent of BigDog, which was a big breakthrough in legged robots. Until BigDog, most legged robots were either upright humanoids with two legs or sprawling six-legged insectoids. With four legs, BigDog is more stable than a humanoid and able to carry heavier loads. With an upright mammalian posture, BigDog has better clearance than an insect body form. The problem with using a dog, or any mammal, for a robot design is that they are unstable.

- To solve this problem, the engineers, led by Martin Buehler, had to create a robot with dynamic stability. The trick is that BigDog is almost always moving its legs, even if it is standing still. If you are always moving, then you are actively keeping your center of mass balanced between your legs. But dynamic stability is costly in two ways: It takes energy to always be moving your legs, and it also takes a lot of sensing to do the balancing act.

- In many ways, BigDog and LS3 epitomize bioinspired design in robotics. Working from the anatomy and physiology of trotting mammals, engineers worked backward from nature, identified the key functional principles, and then built machines that worked in the same way.

Missiles and Close-In Weapons

- The first long-range ballistic missile was the V-2 rocket, which was created by the Germans in World War II and was used against the allies. The V-2 is a robot in the sense that it operates on the principle of sensor-guided movements.

Robots in the Military

1944	V-1 bombers, unmanned aircraft (UA), are deployed by the German air force against targets in England.
1944	V-2 rockets, the first long-range ballistic missiles, are deployed by the German military against urban targets in England.
1995	The RQ-1 Predator, an unmanned aerial vehicle (UAV) built by General Atomics Aeronautical Systems, is deployed by a U.S. Navy/Army team in the former Yugoslavia for reconnaissance.
2007	Predator UAVs fire missiles for the first time in U.S. campaigns, in Iraq and Afghanistan.

- The German engineers built an automated controller for guidance based on a **gyroscope** and a pendulum, the Pendulous Integrating Gyroscope Accelerometer (PIGA). The PIGA senses both the magnitude of the acceleration and speed of the V-2 rocket in that direction. If you know your acceleration, speed, how long you are moving at that speed, and your angle of takeoff, then you can calculate where you are going to land using the simple physics of thrown balls and other objects, called ballistics. Hence, a ballistic missile throws itself to hit a target.

- From a range of 100 miles or more, the V-2 could hit within 600 meters of its target. That's a measure of its functional accuracy. After WWII, the United States and the Soviet Union captured as much of this V-2 technology as they could. The V-2 jump-started their own ballistic missile programs and their space programs. Because of their accuracy, PIGA sensors allowed the V-2 rocket to be independent of human control once the rocket was launched.

- Fully autonomous robotic systems have also been developed for defensive actions. The Phalanx, a close-in weapons system, is a very important unmanned system that has been used by the U.S. Navy since 1980.

- The task of Phalanx is more complex than that of a V-2. It's not just to launch and land an explosive device. Phalanx has to detect, evaluate, track, and engage targets that are rapidly moving and attempting to destroy the Phalanx's ship. The task of the Phalanx is to be the last line of defense against anti-ship missiles and high-speed aircraft. This is a do-or-die situation in which both speed and accuracy are critical.

- The environment of Phalanx is likewise more complex than that of a V-2. It's not just the air, but objects in the air and objects on the water. And there are other ships in your own fleet around, too, so the environment is mixed, with friends and foe sometimes intermingled.

- Because of the need for speed, the Phalanx needs a high degree of autonomy from humans. Under most battle conditions, the job of the human operator is to hit a "hold fire" button if so ordered by the fire-control command. Even though Phalanx automatically detects, tracks, and engages targets, it does not decide who is friend and who is foe.

- All detected targets approaching the ship are treated as foe. This is one way around the difficult problem of combat identification, trying to categorize your targets. Admittedly, this is a very dangerous way to program a robot.

Important Term

gyroscope: Mechanical or electronic rotation sensor that is used for maintaining or measuring orientation; often linked with an accelerometer for more precise location in three-dimensional space.

probabilistic robotics: A framework for the design of robots that explicitly models the uncertainty inherent in the signals provided by sensors, the models created by controllers, and the movement instantiated by actuators.

trilateration: Method for determining location using the mathematics of circles, spheres, and triangles.

Suggested Reading

Huang, Messina, and Albus, *Autonomy Levels for Unmanned Systems (ALFUS) Framework*.

Singer, *Wired for War*, chaps. 1–5.

U.S. Department of Defense, *Unmanned Systems Integrated Roadmap FY2013–2038*.

Question to Consider

1. The Aegis system is arguably one of the most complex multi-agent, human-robot systems. If you were trying to design the next generation of naval systems, what might you learn from the Aegis about its vulnerabilities?

   a. Detection is poor of small, slow weapons or robots of any type—air, surface, or underwater.

   b. Use of highly modularized subsystems, such as the close-in targeting system, doesn't allow full integration of modular systems across the whole platform.

   c. Humans in the loop slow down the response time.

Extreme Robots

Lecture 19

Robots with legs offer some of the most extreme physical challenges in all of robotics. The physics of dynamic walking, whether with two or four legs, is a balancing act similar to an inverted pendulum—and is just as unstable. Even more extreme, the physics of running is comparable to a pogo stick, where balancing is interspersed with short periods of flight. Military packhorse robots already confront the challenges of dynamic stability. The challenges become even greater in some of the largest, fastest, and smallest of all legged or limbed robots, making them some of the most extreme robots in existence.

Size versus Speed

- The laws of physics create a trade-off between size and speed. The study of how size matters is called scaling. The most fundamental scaling law is purely geometric. It's the surface-area-to-volume ratio, also known as the **square-cube law**, which impacts form and function in biology and engineering.

- Isometric scaling refers to scaling something in equal measure; proportional relationships are preserved as size changes. The warning that the square-cube law broadcasts is that isometric enlargement grows volume faster than surface area: The increase in surface area is not enough to "keep up" with the increase in volume. An important sequel of the square-cube law is that the mass and weight of an animal or robot usually increases in direct proportion to volume.

- Isometric scaling of the area-to-volume ratio impacts the movement of animals or robots at extremes of size. At very small sizes and speeds, movement through air, and especially water, is dominated by the effects of viscosity, a measure of the stickiness of a fluid. At the other end of the scale, for the big and the fast, fluid forces are dominated by the effects of inertia, the tendency for objects to keep moving once in motion.

- For terrestrial animals and robots, those that don't move about in fluid, the first challenge is to hold up their weight, at least when standing still. Weight is the force generated by gravitational acceleration acting on their body's mass.

- Mass matters for movement. Newton's second law tells us that if you keep the force constant, then doubling the mass halves the acceleration. In the case of muscles, the force that they can generate is proportional to the cross-sectional area of those muscles. As animals get bigger, the square-cube law tells us that the animal needs to get bigger muscles or they will be able to accelerate only slowly.

- In robotics, one way to postpone the trade-off between size and speed is to build walkers with very light skeletons. A totally different way to solve this problem is to build more forceful motors. The challenge is to increase force and power without increasing weight, friction, or resistance.

- A team led by Dan Granett, a former engineer with NASA, is working to create a new kind of power system for large robots called hydraulic accumulators. Powered by a diesel engine, the hydraulic system is pressurized, and then that potential energy can be quickly released to rapidly move joints.

- The dinosaur-sized Tradinno robot, the world's largest four-legged walking robot, also uses a hydraulic system. It has a turbo diesel engine that pressurizes 20 gallons of hydraulic fluid fed through a series of tubes and into the leg actuators. That hydraulic fluid moves the pistons that, in turn, create the torque that powers the joints of the legs.

- Making Tradinno walk was a serious challenge. The legs have to support the weight of the robot and move in just the right way: a pattern of footfalls that we call a gait when we are thinking about animals. This involves careful coordination. The engineers had to carefully examine animal gaits and movement patterns in order to solve this problem in Tradinno.

- Hydraulic systems are useful because they allow the motor to be centralized. The alternative, which we tend to see in smaller robots, such as Hexy from Arcbotics, is to directly drive each joint with its own motor. A great thing about individual motors is that you can easily program different behaviors by controlling each motor.

- With **hydraulics**, you keep the motor or motors inside the body and use fluid to transmit the force out to the joints. For very large robots, a hydraulic system is a clever way around a physical limit.

- Another kind of extreme is large exoskeleton robots, which are powered exoskeletons that increase the capacity of human beings, including their strength, speed, and ability to use and wield weapons. The largest of these exoskeletons are not mobile because it would take additional power to move legs around and to control posture and balance.

- At the other end of the scale, at the really small end, there are challenges of a different kind. There are mini robots based on the bodies of flies and bees. To build a micromachine that flies and can be powered and controlled, with the lateral maneuvers, has been a big challenge. For small robots, a major problem is having enough fuel, or energy storage, on board.

- Another problem related to power is that electromagnetic motors, like the ones used to power the wheels or flap the tails in robots, don't scale well to very small sizes. The magnetic and electric fields at very small sizes just don't produce much force.

- When you try to make microbots, the trade-off is that the longer that you want your microbot to operate on its own, without a tether to an off-board energy supply, the bigger you have to make the battery. Bigger batteries mean bigger robots.

Nanobots versus Large Robots

- One intriguing way around the problem of putting energy and control on board a tiny robot is to build a world outside the robot that contains both. If you can't modify the robot, modify the environment. This approach is allowing nanobots to be created. With nanobots, the idea is to make tiny robots that operate on the level of molecules.

- The scale for nanotechnology is typically from 100 nanometers down to one nanometer, which is one billionth of a meter. At this scale is where single molecules, such as DNA, live. When we talk about building nanomachines, we are talking about being able to manipulate single molecules.

- Amazingly, a team at Tufts University has built an electric motor from a single molecule. While these nanoscale motors aren't yet ready to start powering nanobots, there is work elsewhere on nanoscale walkers. Kyle Lund and his colleagues at Arizona State University built what they called molecular robots that can move autonomously and in a predictable way. They are just four nanometers long. Lund and his team built an environment out of what is called DNA origami that contained information for the nanobots to start, follow, turn, and stop.

- One trade-off that nanobots encounter is that they move slowly. Lund's molecular robots, when they are really moving quickly, can move about 90 nanometers in 30 minutes, or 3 nanometers (about the length of its body) per minute. As a comparison, in terms of absolute speed, Tradinno moves much faster, but its relative speed is almost two body lengths per minute. By this measure, the largest and smallest robotic walkers move about the same relative speed, about one to two body lengths per minute.

- OutRunner, built by Robotics Unlimited, is almost 20 times faster than Tradinno. OutRunner moves about 900 body lengths per minute, a speed in terms of body length that is hundreds of times faster than Tradinno or Lund's molecular robots.

- Robots at this intermediate size can be built to be the fastest-legged creatures. OutRunner is biologically inspired. The legs are key, acting like individual pogo sticks in much the same way that your legs work when you are running. As the leg hits the ground, it compresses, storing energy in internal springs. That spring energy can be reused to help propel OutRunner forward, pushing it forward as the legs rotate backward.

- OutRunner is a very efficient runner; per step, or per body length, very little energy is used. The battery on board will last for two full hours—outstanding mission time for something moving so quickly. Another aspect of OutRunner's design that makes it efficient is its springlike legs, which store and return energy for propulsion. Also, those legs are lightweight and don't have to be accelerated back and forth, like human legs do.

Robot Legs

- Speed is the product of stride length and stride frequency. What limits your speed as you walk is, in part, that your stride length is limited by the length of your legs. When you run, you actually launch yourself into the air. You are leaping from one foot to the other, and that aerial suspension increases your stride length.

- Based on stride length alone, we would expect the largest animals to be the fastest. A horse is a great example of a big and very fast animal that takes advantage of its long legs.

- If we were to redesign OutRunner to make it faster, we couldn't make the legs longer or have the legs spin faster, because that would break the legs. They would start snapping off. Could we make the legs stronger by adding material, making the legs thicker? This adds mass, and additional mass also acts to increase the energy of impact of the foot with the ground, so the legs would still break off. What about a different kind of material, something like carbon fiber that is stronger without adding more mass? There's no infinitely strong material, so legs will break if they are long enough or spin fast enough.

© regoluis/iStock/Thinkstock.

Horses are fast due to a combination of size and stride length.

- Scaling works for animals and robots: You need to scale your skeleton, your support structure, allometrically—which is a variation from isometric scaling—instead of linearly to compensate for volume and mass increasing faster than the bone's cross-sectional area. This is one of the major trade-offs in legged robots: Longer legs need to be stronger.

- Speed and size are linked to each other by the laws of physics. The ability to generate more force per unit mass is key for the largest-limbed robots. To reduce mass as we push the upper limits of size, we can use strong materials and lightweight structural arrangements to overcome problems that come along with simple isometric scaling. The ability to have energy and motors on board are challenges for the smallest robots.

- One way around this whole problem with legs is to turn them into wheels. Wheels can spin quickly, carry heavy loads, and are less likely to break. But legs can be picked and placed, lifted up and over obstacles or steps, and that is very difficult to do with wheels.

- Robots of extreme size or speed pose extreme challenges for robotics, and using legs increases those challenges even more. But for moving through extreme environments, robots with legs are extremely useful.

Activity

You can build your own extreme robot, using biomechanical simulation tools, which are perfect for creating crazy, new, and extreme robot designs.

AnimatLab is a free simulation system (http://animatlab.com/) that allows you to build whole agents (animals or robots), limbs with actuators, and sensors with sensory fields. Neural control systems are an explicit part of each system, connecting and coordinating sensors and actuators.

Try the hexapod walking robot tutorial:

http://animatlab.com/Help/Tutorials/Examples/Hexapod-Robot.

Important Terms

hydraulics: Using pressurized liquid to create mechanical motion.

square-cube law: Mathematical principle that describes the ratio of the surface area to the volume of a class of similarly shaped objects that differ in size alone.

Suggested Reading

Pinheiro, Han, Shih, and Yan, "Challenges and Opportunities for Structural DNA Nanotechnology."

Vogel, *Comparative Biomechanics*, appendix 3.

Wood, Nagpal, and Wei, "The Robobee Project Is Building Flying Robots the Size of Insects."

Questions to Consider

1. One of the problems associated with becoming larger is that it is easier to fall down and hurt oneself. Small animals, like mice, can fall long distances and suffer little damage; large animals, like horses, break bones when they fall. Part of the reason for this difference boils down to differences in kinetic energy, the energy of motion. Kinetic energy is half the product of the mass of the animal or the robot and the square of the velocity of its motion. Given that formula, what is the difference in the kinetic energy of a 0.01-kilogram microbot flyer crash landing at 1 meter per second and a 100-kilogram robotic cheetah running into something at 10 meters per second?

2. Given what you've just learned about kinetic energy, what would you say is the functional trade-off for pushing the extremes of speed?

Swarm Robots

Lecture 20

One of the grand challenges in robotics is to get robots to work together as a group to do things that they couldn't do alone. The functional benefit of working in a group is the premise of swarm robotics, which deals primarily with groups of simple, similar mobile robots. Swarm robotics proves that sets of behaviors and tasks, such as coordinating groups and working together to build structures, can be accomplished by robots. Swarm robotics is a field that is barely in its infancy, but it has vast potential.

Swarm Robotics

- **Swarm robotics** is related to the field of computational modeling known as agent-based modeling. Physicists are also interested in swarms and have created self-propelled particle theory, where a particle is an autonomous agent, to formalize the behavior of the agents and the swarm. The agent is any autonomous actor, such as a physical robot or an animal.

- The core principle of agent-based models and self-propelled particle theory is that simple rules enacted by each agent independently generate complex behavior of the group as a whole. When we apply this principle in swarm robotics, we frame it as a matter of creation and control: How do we create and then control a swarm?

- In terms of controlling the behavior of the robot, its controller, a computer loaded with code, does only part of the job. When we change the configuration of the body, the behavior of the robot changes; the program on the computer controller is unchanged. The robot interacts with the world through its body, and as the body or the world change, so does the specific pattern of the interaction. That interaction of the robot with the world is called behavior.

- Ironically, and importantly, behavior is only partially controlled by what we call the controller. Control is distributed throughout the robot-world system. Each part has a role to play. This principle can be referred to as distributed control of behavior. This principle can be applied to a single agent interacting in the world, but we can also see it in action in swarms.

Ant Robots and Construction Robots

- Algorithms of swarm navigation have been implemented in robotic ants. Simon Garnier and his colleagues in the SwarmLab at the New Jersey Institute of Technology have built small behavior-based robots that follow simple rules. Each robot has infrared sensors oriented lateral to avoid hitting objects and then two light sensors pointed upward.

- In a maze, a robot heads out from the starting point and blunders its way around, eventually finding a path. It leaves not a chemical trail but a trail of light on the ground, thanks to a special set of overhead spotlights. The next robot that comes along uses its light sensors to detect the path laid down by the first robot. It also leaves a trail of light that makes for any even brighter path for the next robot, and that makes for faster navigation.

- Garnier's ant robots are individual agents that use simple rules to enhance the behavior—in this case, the navigation—of the group as a whole.

- A diametrically opposed approach to the control of swarms is to have the agents be centrally controlled by a single system. Annjoe Wong-Foy at Stanford Research Institute takes this approach with tiny construction robots. These microbots, which can vary in size from one to 10 millimeters in length, are magnetic and are propelled by changes in the electromagnetic field on the printed circuit board over which they hover.

- Because they use magnetic forces and can manipulate objects, these microbots have been called diamagnetic micromanipulators. Each microbot can carry a different set of arms so that specialized worker castes, like ants, can populate the swarm. One caste can carry fibers and another superglue. Programmed to work together, they can construct truss-like systems.

- These microbots are very interesting not only because they can be programmed to build things at a tiny scale, but also because they do so by violating the core principle of agent-based models and swarm robotics: Simple rules enacted by each agent independently generate complex behavior of the group as a whole. Instead, the rules, while they may be simple, are enacted by the central control program that moves the magnets around on the circuit board. Each microbot is not an autonomous agent; they don't act on their own.

- The core principle was discarded with microbots for the simple, practical reason that they are too small to carry an energy supply, a computer controller, or sensors. All of those parts and processes are off-loaded to a smart and centrally controlled environment. Once you do this, the limit to the size of the microbots is only limited by your ability to control the magnetic field that drives them. This is a terrific innovation in swarm robotics that has the potential to change micromanufacturing.

Control in an Unstructured World

- We can't always create the highly structured world required for preprogrammed and centralized control of a group of agents. Almost anything we might want a swarm of robots to do outside will be working in an unstructured world that is rapidly changing.

- Leandro Marcolino at the University of Southern California has been conducting experiments on small mobile robots using a group of six identical e-puck robots attracted to a goal—namely, to move from one side of an arena to the other. There are two groups moving in the opposite direction, so they have traffic to navigate.

- In addition to being programmed with the goal to reach the other side, each robot also has an avoidance reflex built in. When a robot hits or approaches another robot closely, it will stop, back up, and then set off in a slightly different direction toward its goal.

- Marcolino calls this first process uncoordinated in the sense that each agent is acting independently. And both groups are able to make it past the other. However, the time to navigate past the other group is reduced by 50 percent with coordination among the individuals.

- The coordination algorithm is simple: The first robot to detect congestion up ahead signals a warning to the rest of its team. The whole group then switches to the state of deviating from the congestion, at a preset angle, as indicated by LEDs illuminating. The result is a beautiful display of well-behaved robotic traffic. Marcolino's traffic experiments point out the usefulness of communication among agents in a swarm.

- Using a similar process of nonverbal communication, fish swimming in a school may avoid obstacles or threatening circumstances. The schooling behavior of rays or the flocking of birds is not preplanned, and there is not some kind of group leader. That is, even though one individual may be in the front of the formation, different individuals take up that position all the time. The fact that schools of fish and flocks of birds can coordinate movements with nonverbal signals suggests the possibility of simple algorithms.

- In 1987, a computer scientist named Craig Reynolds was looking for a way to realistically simulate how individuals move in a group. Agent-based modeling has its origins in computer-based artificial intelligence simulations, and Reynolds's work was a typical example.

- Reynolds proposed three very simple rules for swarming: what can be called attraction, avoidance, and alignment. These guide the movement of each individual in the group. Attraction is the force that makes you want to move toward other individuals. Avoidance is the force that repels you from others, keeps you from running into them. Alignment is the force that turns you to move in the same direction as others.

- If all three rules are working at the same time, this swarm algorithm creates a chain reaction, coordinating the motions of the whole group. Reynolds's three simple rules allow anyone to make realistic computer simulations.

Neighbor Awareness

- Swarming requires that you be aware of what your neighbors are doing. We humans might naturally think of neighbor awareness as something visual, but it doesn't have to be. When fish swim without any available light, they can still school, using a sensory system called the lateral line, an earlike series of external hairs that detect changes in how the water is flowing around the body. The closer you swim to another fish, the more your flow will be disturbed by their wake.

- But vision-based swarming works, too, as seen in nature and in robot swarms. A Lily robot, created by CoCoRo, moves underwater and uses the blue lights on its body to signal other Lily robots. Each Lily also has photodetectors to pick up those signals.

- Lily robots are detecting what their nearest neighbors are doing. If a neighbor's blue lights are on and blinking, then the Lily that detected that signal starts blinking its blue light. Then, the next neighbor detects the blinking lights and starts blinking, and so on.

- In the case of this group of Lily robots, the chain reaction of blinking allows information to be sent along the physical chain of robots. In this situation, the message is that the robots have detected a robot with blinking blue lights.

Fish swimming without light are still able to school using a sensory system called the lateral line; the closer one fish swims to another, the more the flow will be disturbed by its wake.

- To keep the Lily robots organized, the CoCoRo project used a swarm algorithm based on the behavior of honeybees clustering: They called it BEECLUST. Robots start by moving randomly. If a robot detects an obstacle ahead, it stops and listens for a signal that indicates that the object is another robot. If another robot is detected, the focal robot measures the intensity of light. The higher the light intensity, the longer it waits before returning to random movement.

- BEECLUST is a very simple algorithm for swarming, and it works in the water, with Lily robots, and on land with Jasmine robots. The CoCoRo project envisions using swarms of underwater robots to search more efficiently for things like sunken treasure, downed airliners, or leaks in underwater oil pipes. These search-and-detect tasks are exactly the right kind for swarms of robots. You simply can't search far and wide very quickly with a single robot.

- At least in some cases, you don't need to sense other agents in order to swim in a coordinated manner; sharing a goal and colliding with each other are all you need to create teamwork. Sharing a goal is a key principle in swarm robotics. But goals can be shared in a swarm without all of the robots having to act in the same way at the same time. This allows more complicated tasks to be accomplished, such as mapping a novel environment and then navigating through it.

Activity

Try the Boids simulator:

www.runthemodel.com/models/204/.

This simulator runs on JavaScript, so you can operate it from your web browser and think about the rules of engagement for robotic agents.

Important Term

swarm robotics: The coordination and control of multiple robots of similar type, operating at the same time and place.

Suggested Reading

Floreano and Mattiussi, *Bio-inspired Artificial Intelligence*, chap. 7.

Kernbach, "Introduction to Collective Robotics: Reliability, Flexibility, and Scalability."

Reynolds, "Flocks, Herds and Schools."

Werfel, Peterson, and Radhika, "Designing Collective Behavior in a Termite-Inspired Robot Construction Team."

Other Resources

Video of the original Boids simulation from 1986: https://www.youtube.com/watch?v=86iQiV3-3IA.

Question to Consider

1. Do any of the following tasks strike you as a situation in which robots might benefit from designing stigmergy into the system?

a. The Mars explorer robot Curiosity analyzes the chemical composition of a rock outcropping.

b. A fleet of self-driving cars and trucks transports people and their belongings to a new urban center across the country.

c. A group of search-and-rescue robots looks for a group of hikers that might have been engulfed in an avalanche.

Living Robots?

Lecture 21

Throughout this course, we've been paying close attention to the bodies and behaviors of biologically inspired robots. But beyond body shapes, movement patterns, or sensory processing, even core features of life are beginning to become possible for robots. First, robots can eat, obtaining and harvesting energy on their own. This is metabolism. Second, robots can build themselves. This is growth. Third, robots can evolve. And, finally, robots can replicate themselves and build other robots.

Harvesting Energy

- How close to living can we make a robot? Being alive means that you can harvest energy. And when you can't, you cease to function—you die. Robots need to harvest and use energy, too. The energy supply for robots is often a battery. It's not fair to say that the robot harvests energy if we humans put new batteries in.

- But Roomba goes back to its charger. Other robots can do even more: The PR2 test bed robot from Willow Garage can open doors for itself, move on if a door is locked, find a standard wall outlet, and plug itself into any outlet it finds. Other robots cut the cord: Wave Glider uses wave power to swim and solar power to charge batteries that power its sensors and communications systems.

- But none of these methods of energy harvesting is similar to what animals do. We animals eat food. Our stomach, intestines, and digestive organs work to chemically breakdown the organic matter that we ingest and then extract—at a molecular level—chemical potential energy for later use by our cells.

- EATR (Energetically Autonomous Tactical Robot) is a robot capable of ingesting organic matter for food, or fuel. It's a wheeled ground robot under development by Robotic Technology Incorporated, in partnership with DARPA. The basic idea of EATR is simple in concept: Search the environment for organic material, collect it, and then convert it into a usable energy source.

- The guts of the operation is an external combustion engine. Water in one chamber is heated by the combustion of organic matter, biomass, in an adjacent but separate chamber. Once ingested, the biomass, or food, is burned to provide heat. The heat from the food boils the water. That heat is used in a heat exchanger to generate electric power. But EATR isn't digesting its food chemically.

- However, even real chemical digestion and energy creation are possible. At the University of the West of England, Chris Melhuish and his colleagues have been working on a biologically based fuel cell that uses bacteria—microbes—to convert things like rotting fruit into electric energy. Fuel cells convert chemical energy into electric energy, and they are popular as energy alternatives to internal combustion engines in vehicles.

- Melhuish has put the microbial fuel cells into a robot called EcoBot II. As the fuel cells work, the electric current they produce charges capacitors, which are electronic devices built to store up, and then quickly release, electric charge. The stored charge in EcoBot II's capacitors is used to power either its wheels or communication system. EcoBot II has a controller that works with photodetectors to head the robot toward a light source, powered by rotten apples.

- EcoBot II is a type of gastrobot, a robot that gets all of its energy from the digestion of food. EcoBot II is on the road to true energy autonomy, and that is the goal of Melhuish's research project.

- If we could combine EcoBot's chemical digestion talent with EATR's ability to search for and find appropriate food in the environment, then we would be moving robots toward total energy autonomy and operating like life-forms.

Development

- Another thing that life-forms do, that robots don't, is build themselves. When engineering robots or other devices, we normally assemble the whole thing. One exception is the International Space Station, which has had to function while we've been building it.

- How would you build a robot that builds itself? Sam Felton begins with a sheet of composite material that includes paper, copper, polyimide, and prestretched polystyrene. When you heat this composite sheet of material, the polystyrene shrinks. If you arrange the paper and polystyrene in the right way, the heating will cause the layer to fold. Add heat, the sheet folds, and the sheet builds itself. This is like a simple kind of origami, but instead of a human doing the folding, the paper is folding itself.

- Felton and his team in Robert Wood's lab at Harvard have "programmed" this sheet to fold up in a particular pattern: to make legs, engage a motor, and walk away. This machine builds itself. The "instructions" for building itself were not written in computer code or DNA. Instead, the instructions were built into the laminate sheet. Felton and his team call this programming matter.

- A different approach to life cycles in robots has been taken by Josh Bongard at the University of Vermont. He and his team have focused on metamorphosis, the transition from one life cycle stage to another. Working in digital simulation and with embodied robots, they discovered that individuals programmed to change their bodies over time found new, better behaviors more rapidly than robots that held their bodies constant.

- The robot starts by laying flat on the ground, with a sprawled posture, legs out. It has a controller program whose job is to figure out how to move those legs in order to move. Using an evolutionary program, many different and randomly created controller programs can be tried out. The best one is the one that moves the robot the fastest across the ground.

- Once a good controller is evolved that works well for the sprawled posture, it can be used as the starting point for when the robot brings its legs underneath its body. If the body changes in one fell swoop, then it takes longer to evolve a controller that works well compared to allowing the change in the body's posture to be part of the robot's development. During development, the legs slowly change their posture. That helps bridge the original controller to the final controller. The result is that development speeds up evolution.

- This metamorphosis shows the benefit of having a body that can change and the benefit of having a process that can search for new controllers that can the make best use of the ever-changing body. That search process within the life of an individual can be thought of as the compound processes of learning and development. This takes place by trial and error.

Evolution

- A search process that is related to development is evolution. One of the differences between development and evolution is that while development describes the changes taking place to an individual, evolution describes the changes that are taking place from one generation to the next, from one group of individuals to another group of related individuals—from parents to offspring.

- Like development and learning, evolution works by trial and error, but on a different scale. Evolution is characteristic of life, and using the process of evolution, we can evolve robots.

- For roboticists, evolution is one way to design robots. Standard engineering and hacking both involve humans making decisions about the design of the robots, but this is not so with evolution. We take the humans out of the design loop. This is like taking humans out of the control loop, allowing robots to perform tasks autonomously. With evolution, we are allowing robots to design themselves autonomously.

- We can boil life down to autonomy. All life-forms eat, behave, develop, and evolve on their own. As we work from nature to create robots that have more properties of life, we are extending the kinds of autonomy that they have. Eating becomes energy harvesting. Behaving becomes performing tasks. Development becomes self-assembly. And evolution becomes automatic redesigning.

- PreyRo, a kind of Tadro robot, is a model of the 400-million-year-old extinct fossil fish called Drepanaspis. A team at Vassar College is trying to understand what kinds of selection pressures may have caused an ancient fish like Drepanaspis to evolve the bones in their back that we humans have. Those bones are called vertebrae. The answer lets us understand how vertebrates got their start.

- The team's guess was that predators might be selecting for faster fish over slower ones. Based on biomechanical tests with fish and swimming robots, they correctly predicted that faster fish would be the ones with more vertebrae in their back. To test the evolutionary idea about the importance of vertebrae, they put different individuals of PreyRo, each with a backbone with different numbers of bones, into a tank with a robot predator called Tadiator.

- In the evolutionary trials, the team started with two behaviorally autonomous robots, each with different goals. PreyRo was one of six individuals in its population, and it was competing against the other PreyRos to see who is selected—(indirectly) by the predator—to be parents for the next generation. Each PreyRo was cloned so that there were three identical triplets to test against the predator. With six different genotypes, that made for 18 PreyRo tails that were tested every generation.

- All that any individual PreyRo wants to do is "eat" light, so it heads toward the light. The light is a metaphor for food. One PreyRo was tested at a time in the tank, and while it was trying to eat, Tadiator is on the loose. All that Tadiator wants to do is "eat" other robots, so it heads toward PreyRo, which is food for Tadiator.

- The team ran trials on each individual in each generation multiple times. They scored how well the individuals did, and the top three PreyRos in each generation got to pass on their genes to the next generation to make offspring.

- When we evolve robots, we don't know how events will turn out. We have a guess, such as more vertebrae will make for better feeding and fleeing, but we aren't sure. We have to let evolution run its course. What determines the outcome are random factors, such as genetic mutations that make sure that individuals are different, and selection—in this case, the predator.

- Evolution is an automatic design process; we humans don't predetermine the outcome, but we do play a role. For the population of PreyRos, they depend on us humans to conduct a mating algorithm on the computer. We also use a genetic algorithm to add in small amounts of mutation to the reproductive cycle. When we produce the genomes for the next generation, those are actually plans for us to construct, by hand, the offspring of the parents.

Reproduction

- The idea of self-replicating machines has been a serious scholarly topic at least since mathematician John Von Neumann introduced the thought experiment of a self-reproducing automaton in 1948. Much more recently, NASA has funded research proposals into how robots might replicate themselves, because this could be a very efficient way to begin mining, or other large-scale operations, on other planets.

- A partial proof of concept came from Hod Lipson at Cornell University. His team created a modular robot, built of cubes, that could replicate itself if given other cubes. Each module, called a molecube, is its own autonomous robot, and they link with other cubes via reversible magnetic interfaces. Each molecube has the plans for replication.

- Most importantly, each cube has an actuator that allows it to rotate its faces about a single revolute axis. This allows the cube to grab another cube and then rotate itself to lift that new cube to a position above it; in essence, it stacks itself. With a column of multiple molecubes, the modular robot can bend over to construct itself.

- A completely different approach has been taken by Mark Kim, who runs the modular robotics lab at the University of Pennsylvania. He and his colleagues have built FoamBot, which is able to build other robots of various designs. FoamBot sprays foam onto actuators. The foam hardens, and then the new FoamBot can walk away. The key to this idea is that there is one robot whose job is to reproduce, or make, other robots.

- Both molecubes and FoamBot show us that the idea of reproducing robots is feasible. For robot reproduction to work, what we still need is the robot, or the group of specialized robots, to be able to harvest energy and raw materials, use what they harvest to create finished materials and components, and finally assemble themselves.

Activity

Ludobot, Education in Evolutionary Robotics:

http://www.uvm.edu/~ludobots/index.php/Play/Play.

This is a great place to see robots evolving. Depending on your interest level, you can follow the tutorials and learn how to create digital simulations of evolving robots.

Suggested Reading

Long, *Darwin's Devices*.

Stamp, "A Brief History of Robot Birds."

Wood, *Edison's Eve*, introduction and chap. 1.

Other Resources

Living Machines is an international conference for biomimetic technology and biohybrid robotic systems that combine machines with living tissue: http://csnetwork.eu/livingmachines/conf2015/about.

Questions to Consider

1. What features and/or behaviors would a robot need to have in order for you to consider it to be alive?

2. If we are going to evolve robots to let them autonomously redesign themselves, then it is critical to understand how evolution works. Which one of the following best describes evolution as a process involving robots?

a. An individual robot, through development and/or learning, adjusts its body and/or controller to become better at whatever behavior is being optimized.

b. A group of related but nonidentical robots are judged individually on how well they perform a given task. The designs that perform the best are combined and altered to create a new group of nonidentical robots to be tested.

c. A group of robots works together to solve a task, and by virtue of working together, they are able to do a better job than any of them could do alone. They are able to encode how they accomplished this so that the instructions can be passed down to other robots.

Social Robots

Lecture 22

The complex behavior that we call social and emotional communication is a field ripe for robotics. Spoken words, expressions, tone of voice, and body language can all be used to create the user interface between a human and a robot. Social signals, simple but powerful signals, can be communicated very quickly. And thanks to their sensors and controllers, robots can produce and recognize these standard social and emotional signals to interact more effectively with humans.

Social Robotics

- A social robot is an autonomous robot with a very specific function: to interact and communicate with humans and other autonomous physical agents. But findings from social robotics can diffuse much more widely into other kinds of robots as well.

- Social robotics is a field that combines findings from engineering, biology, and human psychology. Of course, psychology continues to be a dauntingly complex field. But we know enough about how humans approach their world and communicate that we can state the following principles to help us design social robots.
  - Humans tend to anthropomorphize. Most owners of a robot vacuum cleaner give it a name. The classic example is when we ascribe our own thoughts and emotions to our pets. We do the same thing to each other, and we call that empathy.

  - Humans expect a social agent to be interactive. We expect a puppy to react to us with eye contact, sounds, or a change in posture. With humans or social robots, if we ask a question, we expect a response.

- Humans expect a social agent to show initiative. We expect a puppy to have his or her own drives and desires, such as wanting to play or chew on our shoes. We expect the same for a social robot.

- These three principles get us started thinking about what robots need to be socially competent. Social robots must be able to perceive and express emotions; understand and express spoken language; understand and express the "body language" of gaze, expression, gesture, posture, and movement; establish and maintain a social relationship; and have a personality.

The design of a social robot's body must invite human interaction.

- While all these skills feed into social communication, their diversity make it clear that the challenges for social robotics is about more than just having a spoken or written conversation. That we can do with chatterbots (or chatbots), computer programs like Joseph Weizenbaum's ELIZA of 1966, whose text-based interactions launched the field of artificial conversational entities.

- Social robotics focuses on communication that is embodied—communication between two physical agents. Therefore, the design of the body of a social robot is extremely important. In fact, if you take a behavior-based approach, the first thing that a robot's body has to do is to invite the human to interact.

PARO

- Cuteness is an invitation to touch, and Takanori Shibata realized that when he sought to design a therapeutic robot in the 1990s. He chose the baby harp seal as a model, with its white fur and big black eyes. In 2001, he revealed PARO as a therapeutic robot meant to offer the benefits of animal therapy in environments like hospitals and extended-care facilities where having animals is problematic.

- When you touch PARO, it responds, thanks to tactile sensors, by moving its head and body and making sounds. If you talk to PARO, it has directional microphones that allow it to turn toward you, and its controller has algorithms that allow it to learn words, such as the name you give it, and recognize praise. PARO interacts with language, even without producing language.

- PARO also learns what behaviors you prefer in order to initiate interaction. Using a stroke as a sign of positive reinforcement, PARO's controller looks for patterns in its behavior that came before the stroking. PARO will then try to initiate the stroking by exhibiting those pre-stroking behaviors.

The therapeutic robot PARO offers the benefits of animal therapy in places where having animals is not possible.

- In addition to tactile and audio sensors, PARO has light sensors so that it can tell the time of day and alter its behavior accordingly. It also has posture and temperature sensors to give information about how it's being held and the nature of its environment.

- PARO is a great example of social communication through body language and nonverbal auditory signals. Research has found that the interactions of PARO robots with humans can reduce patient stress, improve patient motivation, and improve communication between patient and caregiver.

- Humans also use nonverbal body language in the form of facial expressions and vocal intonation. These dynamic features of our social communication are part and parcel to emotional communication.

Kismet and Jibo

- The most important early robot for emotional communication was **Kismet**, a robot developed in the late 1990s, about the same time as PARO. Kismet, designed by Cynthia Breazeal at MIT, made use of the natural and nonverbal emotional communication that we have with infants. The approach was revolutionary, and it set the stage for humanoid social robots.

- Breazeal built and programmed Kismet to react to human emotions in a way that signaled, to the human, Kismet's own apparent emotional state. Breazeal took advantage of our human tendency to anthropomorphize.

- Humans interacting with Kismet use words and intonation. They treat Kismet like a baby, scolding or praising. Intonation, stress, and rhythm of the speech are all part of what linguists call **prosody**, and Breazeal programmed Kismet to respond to five types of prosody: approval, prohibitions, attention, comfort, and neutral.

- Depending on the type of prosody, Kismet responds by changing its posture, hanging or lifting its whole head, and by changing the position of its ears and the shape of its mouth and eyes.

- Kismet is communicating—through its postural changes—that it has understood the emotional content of the human. Kismet signals surprise by raising its eyebrows and ears and opening its eyes and mouth. Kismet signals sadness by tilting its head down, lowering its ears, and drooping its eyelids.

- Humans interacting with Kismet reported that they enjoyed the experience and felt some kind of emotional connection. Breazeal quickly realized that Kismet was allowing her to study humans and what triggers their emotions and empathy.

- Even though Kismet looks like "just" a head, Kismet is actually an autonomous robot. It has sensors in the form of cameras and a microphone. In addition to visual processing, Kismet has an optional auditory system. Kismet also has the ability to vocalize—not words, but coos that sound like those of a young child. The voice synthesizer has parameters that can be altered to add emotional qualities and adjust the perceived personality of the voice.

- The sensory input, motor output, and heavy processing demands are handled by 14 different computers. Each motor must be controlled, and each sensory input must be processed, but in addition, high-level perceptions, motivations, and behaviors have to be modeled and coordinated.

- The motivation system contains Kismet's six basic emotions derived from human psychology: anger, disgust, fear, joy, sorrow, and surprise. Kismet communicates its emotions through its facial expressions and nonverbal vocalizations, and these different emotional states can be varied continuously in order to create smooth transitions and intermediate neutral states.

- Kismet presents these emotional states in response to the person it is interacting with. Kismet can also initiate a particular string of interactions because it possesses what Breazeal calls drives that signal Kismet's own agenda. These longer-term drives, along with the immediate emotions, create the motivation system, all of which is located in the computers networked to the Kismet head.

- In the short term, emotional interaction begins with Kismet assessing whether a stimulus is positive or negative. One example is personal distance. If you move in too close, Kismet backs away. If you move too far away, Kismet makes cooing-type noises to draw you in. All of this interaction—this social communication based on emotional signaling—only works because both you and Kismet can exercise attention. Breazeal built into Kismet an attention system directly linked to the control of the eye motors.

- One of Breazeal's later robotic creations seems to take lessons from Kismet. Breazeal founded a company aimed at offering the first inexpensive family robot, Jibo, which has a body that is small and simple. Its head is a round screen that can project an eye and swivel around.

- The ideas embodied in Jibo's design don't rely on facial expression to engage humans. What has been carried forward from Kismet is voice recognition, speech production, and attention seeking and giving in the form of the movements that Jibo makes.

Pepper

- Launched in 2014, Pepper is built on the mobile humanoid design but with a wheeled base instead of legs. Pepper is built to learn, understand, and adjust its behavior to human emotions. Pepper was the joint project of the French robotics company Aldebaran and the Japanese company SoftBank Telecom.

- Built with the purpose of being a social companion, Pepper can sense and respond to some emotions in humans. Pepper has what programmers are calling an "emotional engine," and it can learn about emotions through its interactions with you and from other

Peppers interacting with other people. Its aim is to interact with you in a way that is engaging and positive for you, whether you're at home or in a commercial setting.

- Pepper has four microphones, a pair of color video cameras in the eyes and mouth, three-dimensional depth sensor behind the eyes, touch sensors in the head and hands, bump sensors, lasers, sonar, and a gyroscope in the torso. Pepper can sense the position of its joints using Hall effect sensors, which use magnets to measure motion. Pepper's base has two sonars, six lasers, three bump sensors, and another gyroscope.

- Pepper has three modules. Starting at the bottom, Pepper has a transportation module. It can roll around at speeds of up to three miles per hour and uses bump sensors, sonar, laser, and a gyroscope to navigate and avoid objects. In the middle is the manipulation module, which has robotic arms that touch, reach, and gesture. On the top is the interaction module, where vision, hearing, sounds, and signals are created to facilitate communication.

- For a very sophisticated robot like Pepper, all three of these sets of functions—interaction, manipulation, and transportation—are likely to coexist and cooperate on the robot. We can think about each of these modules as its own autonomous robot. Each module is independent from the other in terms of how it creates functional loops of sensors and actuators to do its basic set of jobs. However, each module is dependent on the others to get the overall job of the robot done.

- Softbank expects Pepper robots to learn from their interactions and enhance their learning by having them exchange information with one another about what they learn. This may work especially well for Pepper robots stationed in stores, where the same kinds of interactions may occur many times. Pepper can also record and summarize how people respond, making it a likely tool for market research.

- The most obvious change in context for a social robot is the person with whom they are interacting. What a social robot learns about a given person becomes their model of that person. But it might not apply to other people.

Cobots

- Another change in context for social robots is when they ask for help—and what they do if they don't get it. Dr. Manuela Veloso and her colleagues have invented robots that aren't afraid to ask for help, called cobots, which is short for collaborating robots. Cobots have a specific set of tasks that they do: They can navigate to specific locations in a building, deliver messages, greet and escort visitors, transport objects, provide telepresence for conferences, and offer companionship.

- The only problem is that cobots don't have arms, so they can't pick up objects and, therefore, have to rely on humans to place objects in their basket. They have to ask humans to help. When a cobot gets stuck, it waits for a human to come by. But cobots do get impatient. When a cobot simply can't find a human to help, it emails Dr. Veloso. Then, she or one of her students has to go help out.

- Cobots are not only robots that can navigate around a building using low-level behavior-based systems and high-level world-model systems, but they are also robots that understand what they can and can't do. This is not very far from self-awareness, at least in a very functional sense, and it shows the importance of understanding context.

Important Terms

Kismet: Autonomous robotic head capable of social and emotional interactions with humans that was built by Cynthia Breazeal in the late 1990s.

prosody: Properties of speech other than the words, often conveying emotional content.

Suggested Reading

Breazeal, “Cognitive Modeling for Biomimetic Robots.”

Vernon, *Artificial Cognitive Systems*, chap. 9.

Other Resources

The International Conference on Social Robotics: http://icsoro.org/. Their tagline is “Breathing life into machines.”

Questions to Consider

1. What is a key feature in a robot’s communication with a human? That is, where should a robot’s attention be directed in order to know whether a human is attending to it?

   a. Direction of the human’s gaze.

   b. Response time following a verbal question.

   c. Body language of the human.

2. The use of robots as caregivers opens new possibilities for improved medical therapy. For example, what features might you want to build into a robot whose job it was to make sure that a patient, at home, kept to his or her schedule of medications?

Humanoid Robots: Just like Us?

Lecture 23

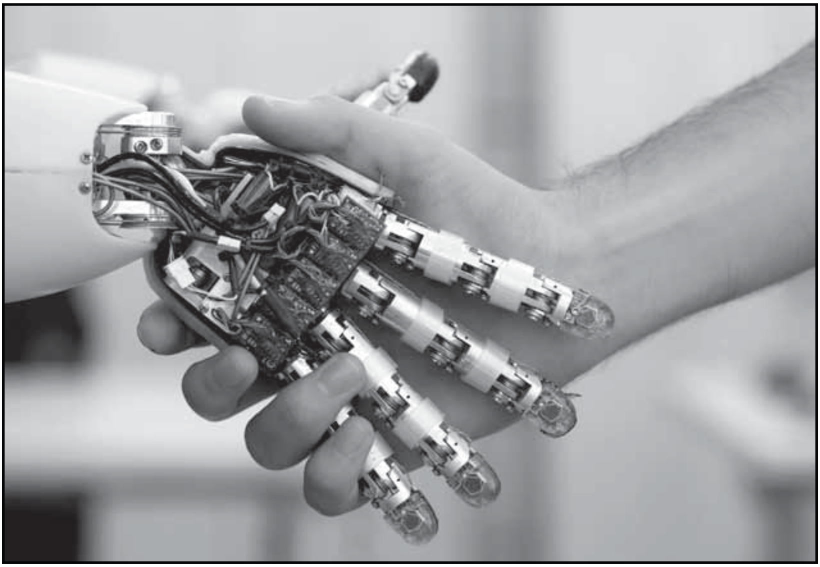
Traditionally, there has been a huge lag between artificial intelligence and humanoid robotics. But we have made considerable progress—if not in closing the gap, at least in advancing what robots can do in the physical world. Most of the advances in robotics have come from building a body for a specific set of tasks in a given workplace. Often, the workplace is modified to help the robot. The science fiction approach to humanoids presumes that we are trying to build robots that are just like us, without thinking of specific tasks or workplaces. By contrast, some projects consider explicitly defined tasks in a human environment that has not been modified to help the robot.

Humanoid Hands and Arms

- Each body part of a humanoid provides its own engineering challenge. Hands turn out to be very complex to engineer, in part because each one has at least 15 degrees of freedom, three per finger. Each degree of freedom needs to be under independent control, yet it's difficult to fit all the electric motors, or pneumatic muscles, into the confined space of a human-sized hand.

- The RAPHaEL (Robotic Air Powered Hand with Elastic Ligaments) robotic hand is an air-powered robot hand, introduced in 2009 at Virginia Tech. The use of compressed gas is what provides the energy and pressure to move the fingers, allowing the hand to hold onto light bulbs or heavier objects with carefully controlled levels of force.

- Dexterity can also be achieved with miniature direct current motors embedded in the hand itself. In 2012, the German company SCHUNK released a commercial five-fingered hand. Each finger has three degrees of freedom. The motions are driven by small, geared direct current motors embedded in the palm of the hand and


© muratsenel/iStock/Thinkstock.

Hands are very complex to engineer, in part because each hand has at least 15 degrees of freedom.

in the finger itself. Careful control of these motors, and feedback from pressure sensors on the surface of the skin, allows the SCHUNK hand to hold a key or a pin. It can also shake hands safely with a human.

- Shadow Robot Company sells a Dexterous Hand with 129 sensors. This is a hand that has 20 actuated degrees of freedom, plus four additional joints that are indirectly actuated. The thumb has five extra degrees of freedom, and the pinky has four. In combination, the hand has greater dexterity. The Dexterous Hand can operate as an autonomous robot and can be attached to other robots.

- Just as the hand can become a module to an arm, fingertips can become modules of the hand. Modular fingertips have been designed by SynTouch, a company started at the University of Southern California, to have the sensitivity of a human finger.

- But there are always trade-offs. For many tasks, the only important aspect of the human hand is the opposable grip of the thumb and other fingers for grasping. Baxter's end effector only has one degree of freedom, but it might be all you need for picking up different types of objects.

- Moreover, a great thing about robots is that we don't need to limit their workings to whatever humans happen to have. A great example of breaking out of the human box is the vacuum cup gripper, which uses a vacuum pump to generate a negative pressure relative to ambient. Factory managers have long known that a vacuum cup gripper can grasp things that are sometimes really difficult for fingers to manage: flat surfaces or smooth shapes. But we haven't been putting grippers like that on humanoid robots until much more recently.

Humanoid Legs

- Robotics legs that mimic the gait of a human walk with incredible biorealistic functioning were introduced in 2012 by researchers at the University of Arizona. Humans have a neural network in the lumbar region of the spinal cord that controls and produces rhythmic neural signals to the muscles responsible for routine walking. The researchers created a simple robotic version of this system, where feedback from load sensors on the legs provided information to a central pattern generator, which varied the frequency of its rhythm in response.

- The earliest humanoid robots relied on "static walking," which was more of a shuffle than a walk. A good example of static walking is Robosapien, one of the most famous humanoid robot toys. Robosapien, made by the WowWee company, is controlled remotely and is famous for its autonomous mode, where it can move around the house on its own, avoiding objects. The Robosapien class of robots was invented by Mark Tilden, known for developing robots with solid-state electronics that do not need an embedded computer.

- The problem with static walking is that it involves a lot of rocking from side to side. In larger humanoids, this movement sideways takes up energy better spent moving forward. Rocking is energetically inefficient. Also, rocking creates instability that is liable to make your robot fall over.

- The key to stable walking is to keep your body's center of mass in between the support provided by your feet. To mimic human feet, we can put pressure sensors on the feet of our humanoids to let them know when they're tipping over.

- When humans are actually walking, we use our legs as inverted pendulums. From the time our heel strikes to the time our toe flips off the ground, we're using our leg to pivot our mass up and over our foot. The key to the dynamics of this motion is to coordinate the transfer of momentum from one leg to the other, always keeping

The Development of Humanoid Robots

1973	WABOT-1, the first full-scale humanoid robot, with systems for vision, conversation, and walking, is developed at Waseda University in Japan.
1991	The BEAM (biology, electronics, aesthetics, mechanics) approach to robotics is started by Mark Tilden.
1997	RoboCup, a soccer match for autonomous robots, has its first tournament.
2000	ASIMO, a humanoid robot, is introduced by Honda Motor Corporation.
2005	Ballbot, a mobile robot that balances on a sphere, is created by Ralph Hollis.

the center of pressure inside the supports of the feet. Even though we think about walking as something we do with our legs, walking involves the whole body.

- The robot often considered to be the world's first self-propelled bipedal humanoid is WABOT-1, who stepped into robotic stardom in 1973. Researchers led by Ichiro Kato at Waseda University in Tokyo built a series of WABOTs over the years, and by 1984, WABOT model WL-10RD became the first robot to achieve dynamic walking.

- Dynamic walking is a balancing act. The robot Atlas, introduced by Boston Dynamics in 2013, is a great balancer. Atlas is able to maintain its pose as a stable inverted pendulum by virtue of making rapid adjustments to its center of mass and, in turn, that center of pressure on the foot.

- For humans to balance standing up on one leg, or even two legs, is an active process. We're constantly using our muscles, sensing the position of our joints and the tension in our muscles. If we were to stop doing all of that work, we would just crumple to the ground. Biped robots moving like a human will fall down unless they actively balance.

- One solution for the problem of a statically unstable structure is to put it in motion. Passive dynamic walkers adhere to this principle. Using only gravity, a jointed linkage system will walk on its own if it is arranged just right on a tilted surface. Passive dynamic walking is dynamically stable because one leg catches the falling system. The system pivots over that leg, and then the other leg catches the system.

- Atlas, the balance master, also has impressive stability when walking. It can sense changes in its center of pressure on its feet and make rapid adjustments to its center of mass. Atlas can keep its balance while walking with obstacles put on the path that it has to step over. Even more impressive is its ability to keep its balance while walking across an uneven surface of good-sized rocks.

- Atlas walks in a manner called "Groucho walking" in human biomechanics. With a Groucho walk, named after the famous comedian, you keep your center of mass level, and you never pivot up and over your planted foot. Your knee never straightens. The Groucho walk is very stable: The robot is squatting all the time, so it has a lower center of mass. Also, a Groucho walk allows you to move with precision that you don't normally have with an inverted pendulum walk.

Humanoid Appearance

- Another frontier in humanoid robotics focuses on how the body looks. Looks turn out to be very important for how we react emotionally to robots. While the promise of humanoid robotics is that robots can fit immediately into our workplaces and homes, we won't want them around unless we feel comfortable with them.

- If your affinity for a robot is low, indifferent, or negative, that eventually means no attention. And no attention means no communication. Professor Hiroshi Ishiguro of Osaka University in Japan addresses this problem of affinity. He is working with a company called Kokoro to build highly realistic humanoids that they call **Actroids**.

- The Actroid model Geminoid HI-2 is built to gaze, blink, turn, gesture, and interact verbally with people. Geminoid HI-2 is remotely controlled by a human. When it speaks, it is the voice of the human in the control loop. Actroids have come in many different models, and more recent ones, such as Actroid-SIT, have conversational autonomy.

- In 1970, robotics professor Dr. Masahiro Mori, working at the Tokyo Institute of Technology, created a hypothesis that captured what turns out to be a complex relationship between the level of our affinity for a robot and how humanlike it is.

- Mori proposed that as a robot becomes more like a human in appearance, our affinity for it grows—at first. However, as soon as the robot gets very close to human likeness, such as what you might see in a scary mask, then the appeal of the robot quickly plummets to creepy. Mori called this small region of disgust and creepiness the **uncanny valley**.

- Mori knew that affinity would matter if we wanted humans to accept robots and be willing to work together with them or to purchase them. However, some researchers now think that the location of the uncanny valley might change over time or depend on culture.

© Koichi Kamoshida/Getty Images News/Thinkstock.

As robots look increasingly similar to humans, our reactions become increasingly positive—until we reach the point of revulsion, known as the uncanny valley.

The Future of Humanoid Robotics

- What's next for humanoid robotics? Smooth transitions from one kind of movement to another (for example, from walking to climbing stairs, or changing pace, or stopping on a dime) will be important. Another will be allowing the arms to carry objects or perform other tasks while the legs continue moving the entire body in various ways.

- A third area of work will be thinking fast and moving fast. Those are two related challenges we see being addressed at the annual RoboCup, the robotic soccer contests that are held around the world. NAO humanoid robots, made by Aldebaran, the maker of the social robot Pepper, are a favorite among robotic soccer teams.

- NAO comes with two cameras, an **inertial measurement unit**, touch sensors, four directional microphones, and two sonar rangefinders. When NAO teams play soccer, they are moving autonomously. They use cameras to detect the ball and to recognize their teammates versus the opposition. Their inertial measurement unit measures acceleration so that they can keep track of their balance and position.

© Jens Schlueter/Getty Images News/Thinkstock.

NAO robots play soccer autonomously.

- Teams of humanoids are another focus. Working with humanoids that look a lot like Baxter, General Motors is working on coordinating the reaching and grabbing—the robotic pick and place—of multiple humanoids. The challenge is that the robots have to be able to know where they are and where they need to be, as well as where others are and are likely to be. Better sensors, better actuators, and better controllers will all be needed and will continue to be the foci of research and innovation.

Important Terms

Actroid: Humanoid robot, manufactured by Japanese robotics company Kokoro, that is patterned after specific humans and built with high detail for facial features and gestures.

inertial measurement unit (IMU): An integrated sensor that measures orientation, velocity, acceleration, and, sometimes, compass heading.

uncanny valley: Hypothesized by Masahiro Mori, a sudden loss of affinity that humans can feel as masks or robots become almost, but not quite, human.

Suggested Reading

Hornyak, *Loving the Machine*, chap. 9.

Mori, "The Uncanny Valley."

Rosheim, *Robot Evolution*, chap. 6.

Question to Consider

1. When you think about the tools that Leonardo da Vinci had for building a humanoid robot, what do you think his greatest challenges would have been?

The Futures of Robotics

Lecture 24

Beyond the many interesting robots already in existence, what makes robotics such an exciting endeavor is that there are so many different possible futures. This variety of possibilities stems partly from the fact that robotics is a multidisciplinary collaboration of mechanical and electrical engineering, computer science and neuroscience, and cognitive science and biology. The moment we encounter new problems and challenges will be the moment that we see the necessity to create new kinds of robots to help us respond. Using the principles of robotics, we learn to understand and design the robots of our many futures.

Specialists versus Generalists

- Robotics is such a diverse field that distinct robot origins and robot futures can be found in virtually every domain of modern life and every field of science and technology. What unites robotics is a fundamental trade-off that comes up in the design of nearly any tool or machine: specialist versus generalist.

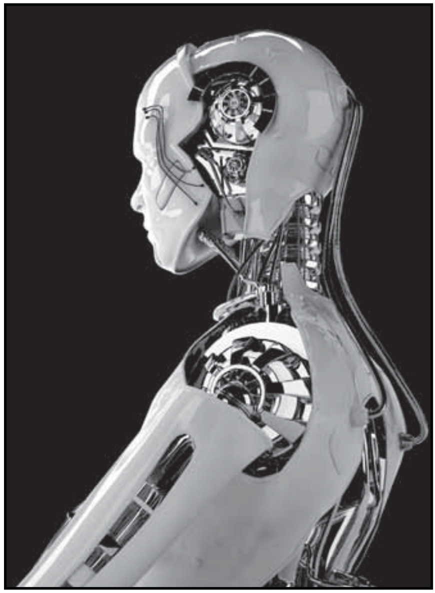
- A specialist does one or a few jobs. Usually, the specialist does the tasks very well, very efficiently, or very cheaply. Roomba is an example of a specialist built to vacuum floors.

- In contrast, a generalist does many things but usually does no single task as well as a specialist would. But the value of a generalist is that a single robot can do many things. Baxter, built by Rethink Robotics, is a great example of a generalist. Baxter is able to do any number of different jobs that involve picking and placing.

- A generalist from science fiction would be a walking, talking humanoid that could work in any of the physical spaces that a human can and do all the physical tasks that a human could.

- In the short term, it's probably more likely that we'll be in a world where the specialist robots dominate. We'll have robotic vacuum cleaners, window cleaners, lawn mowers, dog walkers, and cars. In this future of specialized robots, we won't even see most of these as robots; they'll just be our appliances, vehicles, and workplace machines.

- As with any tool or technology, we have legitimate concerns about how we choose to use robots. We can focus our concerns on something that we know how to create and control in robots: autonomy. For every kind of robot, for every task, in every workplace, we can ask this general question: What type and level of autonomy will allow the robot to accomplish its task without harming humans?



© abidal/iStock/Thinkstock.

A humanoid robot that does a variety of tasks could be considered a generalist.

- Is energy autonomy important because the mission is long term and refueling is impractical? Think about Wave Gliders swimming across the Pacific Ocean and harvesting energy from waves to do so. Is behavioral autonomy important because the mission is remote and humans can't rapidly communicate to the robot about how to perform a task? Think about the Curiosity robot on Mars.

In the future of specialized robots, we will likely see them not as robots but as appliances, machines, and vehicles.

Cloud Robotics

- One way to envision how the various dimensions of robot autonomy will be managed is to focus on what is called **cloud robotics**. Just like with cloud computing, where you share applications or storage capacity over the Internet, the idea behind cloud robotics is to link up individual robots with shared programs and storage capacity.

- RoboEarth is a cloud robotics project started by the European Commission in 2009. The idea is to create a collective database that robots access directly. If a robot encounters a new situation, it queries RoboEarth to see if another robot has already solved the same problem. If a robot learns something new on its own, then it uploads code for its newly created solution.

- In a cloud robotics future, intelligence is linked to your ability to communicate with other robots and learn from their experience without having to learn the task directly yourself.

- To make cloud robotics a reality, the RoboEarth project created Rapyuta, the working cloud platform. In addition to sharing solutions with each other, one idea with Rapyuta is that you can do your computationally expensive processing in the cloud. In this model, you are splitting up the work of the controller into two parts: Part of your controller is operating in the robot, and part of your controller is operating on a clone of your system in the cloud.

- Google was awarded a U.S. patent for cloud-based robotics in 2014. The name of the patent is "Shared robot knowledge base for use with cloud computing system," with Ryan Hickman as the lead inventor. The idea is similar to that of RoboEarth: to connect robots to and through the cloud.

- The fact that Google has been able to patent what may be seen in the courts as a huge part—or all—of cloud robotics gives them an advantage over any other company that wants to use the web to coordinate their robots, have their robots learn from the experience of others, or share controller space.

Watson

- Cloud robotics is not the only future for increased autonomy in robotics. Imagine professional and personal settings where you cannot, or do not want to, expand the abilities of your robot by using cloud computing or cloud robotics. What might be an innovative and alternate way forward?

- IBM's Watson system, an example of what is called **cognitive computing**, is one of the most intelligent computer systems on the market. Watson made headlines in 2011 when it defeated in the game show *Jeopardy!* the two best humans to have ever played that game.

- *Jeopardy!* Watson was followed by a variety of more practical applications, including Dr. Watson diagnosing patients, Chef Watson coming up with new recipes, Finance Watson, and Customer Service Watson helping callers and offering technical support advice.

- All these versions of Watson were not a robot but, rather, a computer program. If Watson had a body, what could we do with Robot Watson? For example, imagine an emergency situation where you need to be able to think on your feet. What if communication with remotely located humans had degraded because you, the robot, were underground, or electric noise was swamping your transmissions. What do you do?

- First responders, acting on their own, are trained to assess the situation at hand: What is happening, why is it happening, and what, if anything, can be done and in what order should I proceed?

- First Responder Robot Watson walks in and compares the readings on its cameras, chemical sensors, and hazard detection systems to data patterns in its database. What is happening? Watson makes an educated guess: Natural gas is being vented into the room. Why is it happening? A pipe has broken. What can be done? The cut-off valve can be closed.

- Because Watson the robot has the ability to judge the confidence of its answers, and make a decision based on the particular situation at hand, it recognizes that if in fact the leak is natural gas, then the situation is extremely volatile—any spark will cause a devastating explosion. Thus, Watson accepts this natural-gas-leak answer with a relatively low confidence threshold. It is better to assume the leak is natural gas and be wrong than to waste time seeking higher confidence. Watson walks over to the shut-off valve and turns off the leaking gas.

- Another great benefits of Watson-style cognitive computing is that a robot can estimate when it does not know enough about the situation to make a decision. Sometimes it's better not to guess. For humans, knowing when not to guess is often the difference between being a novice and being an expert. A robot Watson would be good at knowing when to call in other robots.

Modules

- Because the various approaches to robotics all share the same goal—autonomous behavior—each approach, each discipline, has something to offer other approaches within robotics. The result is that we have solutions, principles, and concepts that serve as modules we can mix and match.

- When you have modules, you can reconfigure their physical connections and reassign their tasks. One of the challenges of a swarm is that specific jobs have to be assigned and even swapped dynamically, as the situation develops, and not ahead of time. When this problem is solved, the whole system is able to adjust as the task unfolds, circumstances change, or some robots fail.

- This is the power of modules. Imagine giving Roomba a module to handle extreme conditions. Or what about a Roomba module for swarm robotics, allowing a team of Roombas to coordinate how they clean a very large space? Or what about a social robotics module for the Roomba, making it more interactive, thereby easier and more fun to take care of? Modules and modular robots expand the limits of what we think of as a robot.

Inspired by Nature and Art

- Where do wonderfully crazy ideas like modular robots and biohybrid robots come from? How might we think of new kinds of bodies, sensors, controllers, and actuators—or new modes of behavior and cognition, action and perception?

- History shows that roboticists are inspired in many different ways. But two paths to the future stand out. The first source of inspiration, from behavior-based robotics to biohybrids, is nature, especially biology.

- The bioinspired future is a wet one. This future involves an even more fundamental shift away from the classic approaches of mechanical and electrical engineering to the bioinspired approaches of tissue engineering and synthetic biology that are helping us create biohybrid systems.

- Classically, we build robots out of dry materials, such as metal, rubber, and plastic. But when we look at animals, they build themselves out of wet materials, such as muscle and bone. A biohybrid system contains materials of both types in order to get the best of both worlds.

- One of the first of biohybrid robots is Medusoid, a robotic jellyfish created by Janna Nawroth and her colleagues at Caltech and Harvard. Medusoid's body is built out of silicon rubber. On top of that body Nawroth deposited proteins. She used those proteins as a scaffold upon which to grow cardiac cells from the heart of a rat.

- After the cells grew and connected, they put Medusoid into a tank and then pulsed electric charges in the water. The cardiac cells responded by contracting, just like they would in a heart stimulated by an emergency defibrillator. Because of the pattern in which they had arranged the protein upon which the cells grew, the contractions actuated the body in a way that allowed the whole robotic jellyfish to swim.

- Medusoid is a great, simple proof of concept that we can begin to grow parts of robots. Because cells are wet and need to be kept wet in order to function, the key to the growth of wet robotics will be either to work in a wet medium or to build bodies that contain that wet medium inside of their bodies.

- The promise of biohybrid systems like Medusoid is that with the growth of tissue, such as muscle, we expand our toolbox of solutions for robots. For example, animals are great at building extracellular tissue, such as tendon and ligament, that have mechanical properties that are tuned to the motion and load of the system in which they are operating. In addition, a system that can grow itself should also be able to heal itself. This would be a terrific ability for any robot on a remote mission.

- A final path to the future, as the best writers and movies make clear, is art. As a society, we sometimes get into a kind of team competition: science versus art. But some of the greatest scientists were on both teams, such as Leonardo da Vinci, who also was arguably the first roboticist. Moreover, the teams of art and science do different and complementary things.

- Art can imagine the impossible and the highly improbable. But art can also offer scenarios for what is possible or even likely. Consider science fiction, for example, as prototyping the future in our imaginations.

Important Terms

cloud robotics: A subdiscipline in which robots are designed to interact with databases via computer networks and then with each other.

cognitive computing: A new style of programming pioneered by IBM's Watson that aims to simulate human cognitive processes, such as understanding natural language, evaluating context, and assessing the reliability of its knowledge.

Suggested Reading

Hickman, Kuffner, Bruce, Gbarpure, Kohler, Poursohi, Francis, and Lewis, 2014, Shared robot knowledge base for use with cloud computing system, U.S. Patent 8,639,644 B1.

van de Molengraft, "Final Project Report."

Nourbakhsh, *Robot Futures*, chap. 6.

Other Resources

The New York Times. "Shooting for a Moon Filled with Robots." www.nytimes.com/interactive/2013/12/04/technology/google-new-generation-robots-videos.html?ref=technology. This web-based article looks at the first wave of robots acquired by Google. Make sure to read the related article at www.nytimes.com/2013/12/14/technology/google-adds-to-its-menagerie-of-robots.html, where you'll find out why this looked at the time like a "moonshot," an all-out effort of time and resources to achieve a goal.

Questions to Consider

1. As we've seen with embedded microprocessors becoming part of every appliance, we are likely to see embedded robotic systems become much more common. Which technological sectors are ripe for a dramatic increase in embedded robotic systems?

   a. Transportation of goods and people.

   b. Assembly and manufacturing.

   c. Medical devices used on and in the bodies of humans and their pets.

   d. Agriculture.

e. Retail services.

f. Personal and home health care.

g. Military and defense systems.

h. First-responder services.

2. Put on your entrepreneurial hat. If you wanted to assemble a team to design, engineer, and produce a new commercial robot, how would you proceed to search for ideas that you could pitch to venture capitalists?

a. Look for a profession that is dominated by humans doing manual labor (see previous question) or by humans providing social interactions and information transfer.

b. Look for any task that is done by humans or for which humans use a tool or machine.

c. Look at forms of entertainment that humans enjoy.

d. Look for any tool or appliance that does not yet possess an embedded robotic system.

e. Look at military and defense systems.

f. Look to nature for life-forms that behave as our machines do not.

Timeline

~1495	Leonardo da Vinci designs a mechanical knight using pulleys and cable, as well as a mechanical cart capable of serving as a programmable controller.
1739	A duck automaton by Jacques de Vaucanson draws crowds to see fake mechanical digestion.
1788	Governor, an automatic speed control, is invented for steam engines by James Watt and Matthew Boulton.
1868	The torpedo, the first autonomous underwater vehicle (AUV), is invented by Robert Whitehead.
1872	*Erewhon*, Samuel Butler's novel, develops the idea that machines might evolve consciousness.
1897	The remote control is first patented and demonstrated on a model ship by Nikola Tesla.
1907	Tik-Tok, a fictional mechanical man with windup keys for thought, movement, and speech, appears in *Ozma of Oz* by L. Frank Baum.
1918	An aerial torpedo, the first unmanned aerial system (UAS), is invented by Elmer Sperry using a modified Curtiss N-9 seaplane.
1920	*R.U.R.* (Rossum's Universal Robots), a play published by Karel Čapek, popularizes the word "robot."
1927	Maria, the first robot to appear in a movie, stars in the silent classic *Metropolis*.

1938	An industrial programmable robot, introduced by Griffith P. Taylor in "An Automatic Block-Setting Crane," *Meccano Magazine*, is the first example fitting later criteria of international standard ISO 8373:1994.
1941	The word *robotics* is first used in the short story "Liar!" by writer Isaac Asimov.
1942	The three laws of robotics are created by Isaac Asimov in his short story "Runaround."
1944	V-1 bombers, unmanned aircraft (UA), are deployed by the German air force against targets in England.
1944	V-2 rockets, the first long-range ballistic missiles, are deployed by the German military against urban targets in England.
1948–1950	Elsie and Elmer, the first biologically inspired autonomous robots, are built by neuroscientist Gray Walter using analog electronics.
1950	*I, Robot*, a collection of robotics short stories by Isaac Asimov, is published.
1950	The Turing test, the so-called imitation game, is proposed by Alan Turing as an operational test of intelligence for computers.
1956	*Artificial intelligence* is coined and launched as a research area by John McCarthy and others at a conference at Dartmouth College.
1956	Unimation, the world's first robot company, is founded by George Devol and Joseph Engelberger.

1958	Numerically controlled (NC) machines are first produced by FANUC in Japan.
1958	Autopilots in cars become the first commercially available speed control on land.
1961	"Programmed Article Transfer" (U.S. Patent 2,988,237), key intellectual property for Unimation, is issued to George C. Devol, Jr.
1962	Unimate, the first industrial robot from Unimation, goes to work unloading die casts at the General Motors factory in Trenton, NJ.
1968–1970	Shakey, the first autonomous mobile robot with digital electronic computer, is built by Stanford Research Institute.
1968–1972	Zero-moment point analysis for motion planning in bipedal robots is developed in papers by Miomir Vukobratović.
1970	Lunokhod 1, a Soviet remote-controlled lunar rover, becomes the first unmanned vehicle to explore the surface of an extraterrestrial planetary body.
1972	The Mowbot patent for the first robotic lawn mower is issued to S. Lawrence Bellinger.
1973	Famulus, the first industrial robot with six electromechanically controlled axes, is introduced by German robotic company KUKA.
1973	WABOT-1, the first full-scale humanoid robot, with systems for vision, conversation, and walking, is developed at Waseda University in Japan.

1973–1974	T3, the first commercially available robot arm controlled by a minicomputer, is designed by Richard Hohn and released by Cincinnati Milacron.
1976	Robot arms are used on Viking 1 and 2 space probes.
1979	The Stanford Cart, designed by Hans Moravec, autonomously navigates a room full of obstacles.
1979	Robert Williams becomes the first human killed by a robot while working in a Ford Motor casting plant.
1981	SCARA industrial robots are introduced in Japan.
1982	KITT, a fictional autonomous talking car, appears in the television series *Knight Rider*.
1982	Fictional "replicants" appear in *Blade Runner*, a science fiction film adapted from Philip K. Dick's *Do Androids Dream of Electric Sheep?* (1968).
1984	*Vehicles*, a book by Otto Braitenberg lays the groundwork for the design of the simplest-possible autonomous robots.
1985	FANUC develops robots to assemble other robots.
1986	A fully autonomous robot car begins test-drives on empty streets in Germany.
1986	Genghis, a behaviorally autonomous hexapod robot, is introduced by Rodney Brooks, Colin Angle, and their team at MIT.
1987	*Star Trek: The Next Generation*, a television show, debuts with android lieutenant commander Data.

1988–1991	HelpMate, the first working hospital courier robot, is developed and debugged at Danbury Hospital by Joseph Engelberger's second robotics company, Transitions Research Corporation (renamed HelpMate Robotics in 1997).
1990	The iRobot Corporation is founded by Rodney Brooks, Colin Angle, and Helen Grenier.
1991	The BEAM (biology, electronics, aesthetics, mechanics) approach to robotics is started by Mark Tilden.
1994	VaMP and VITA-2, driverless cars, operate safely in traffic for more than 600 miles in Germany.
1994	MOTOMAN offers the world's first controller for synchronizing two robots.
1995	The RQ-1 Predator, an unmanned aerial vehicle (UAV) built by General Atomics Aeronautical Systems, is deployed by a U.S. Navy/Army team in the former Yugoslavia for reconnaissance.
1995	Adaptive cruise control for cars, using lasers, is introduced by Mitsubishi.
1997	Sojourner, the semiautonomous rover of NASA's Pathfinder mission, becomes the first robot to land and work on Mars.
1997	RoboCup, a soccer match for autonomous robots, has its first tournament.
1997	Deep Blue, IBM's computer, defeats world chess champion Gary Kasparov.

1998	The LEGO Mindstorms robotics kit is introduced, based on MIT Media Lab's programmable brick technology.
1999	AIBO, the robotic dog, introduced by Sony, sells more than 150,000 units before ceasing production in 2006.
2000	Kismet, a robotic head built to interact socially with humans, is created by Cynthia Breazeal at MIT.
2000	The da Vinci Surgical System, a robot by Intuitive Surgical allowing surgeons to be remotely present and operate inside patients, is approved by the FDA.
2000	ASIMO, a humanoid robot, is introduced by Honda Motor Corporation.
2002	Roomba, an animal-inspired robotic vacuum cleaner, is introduced commercially by iRobot.
2004	Spirit and Opportunity, semiautonomous NASA rovers and geological explorers, land on Mars.
2005	Stanley, a robotic car built by a team from Stanford and Volkswagen, wins the DARPA Grand Challenge by being one of five vehicles to autonomously navigate a 132-mile off-road course through the desert.
2005	Ballbot, a mobile robot that balances on a sphere, is created by Ralph Hollis.
2005	Arduino, an inexpensive open-source microcontroller, is created by Interactive Design Institute in Ivrea, Italy.
2006	Second-generation LEGO Mindstorms NXT robotics kit is released.

2007	Boss, a modified Chevy Tahoe built by Carnegie Mellon and General Motors, wins the DARPA Urban Challenge, navigating a 60-mile course while obeying all traffic laws of California, including avoiding pedestrians.
2007	Predator UAVs fire missiles for the first time in U.S. campaigns, in Iraq and Afghanistan.
2009	RU-27, also known as Scarlet Knight, becomes the first AUV to cross the Atlantic Ocean underwater, a journey that took 221 days.
2012	The Curiosity rover, a robotic geological explorer, lands on Mars.
2012	Nevada and Florida become the first U.S. states to permit testing of autonomous vehicles on ordinary roads; Michigan and California follow in 2013.
2013	The Wave Glider robot Benjamin from Liquid Robotics sets a world record of 7,939 nautical miles, the longest journey by an autonomous surface vessel.
2014	The DEKA prosthetic device, controlled by electromyogram (EMG) signals from muscles, is approved for amputees by the FDA.
2015	The DARPA Robotics Challenge focuses on robots for disaster response.
2015	Jibo and Pepper, the first commercially available social robots, are offered by Jibo and SoftBank/Aldebaran Robotics, respectively.

Glossary

accelerometer: A sensor that converts changes in velocity into electric signals that can be read by the controller.

acoustic sensor: Also known as a microphone, a sensor that converts sound waves into electric signals that can be read by the controller.

Actroid: Humanoid robot, manufactured by Japanese robotics company Kokoro, that is patterned after specific humans and built with high detail for facial features and gestures.

actuator: The moving parts of a robot that allow it to act; any part, appendage, or mechanical system that uses motors to move a robot or manipulate the world through movement.

android: Used mostly in fiction to refer to a humanlike robot. See **humanoid**.

animatronics: Robotic devices used to simulate animals or humans for education and entertainment.

articulated robot: A manipulator or arm that has hinge (revolute) joints.

Arduino: An open-source hardware and software company known for microcontrollers.

ASIMO: Type of humanoid robot emphasizing bipedal mobility over uneven terrain; in development since 2000 by Honda.

automaton: A mechanical machine shaped like a human or animal that works automatically without feedback from sensors or direct control by humans.

Baxter: A two-armed robotic manipulator introduced by Rethink Robotics in 2012 for light manufacturing tasks, featuring rapid reprogramming and safe interactions with humans.

BEAM robotics: Created by Mark Tilden, an approach to building robots that uses analog electric circuits instead of computers for controllers.

behavior-based robotics: Design of robots that eliminates or minimizes the use of internal world models and maximizes the use of reflex-like sense-act modules.

biomechatronics: Applied science that combines mechanical and biological parts into a single device.

biomorph: Any robot modeled after a life-form.

camber: The measure of the front-to-back curvature of a wing that helps create lift.

Cartesian coordinate robot: An industrial robot with three perpendicular, translational degrees of freedom.

caster wheel: Type of unpowered wheel mounted to a robot to provide stability and maneuverability.

chassis: The primary structural support system of a robot's body.

cloud robotics: A subdiscipline in which robots are designed to interact with databases via computer networks and then with each other.

cognitive computing: A new style of programming pioneered by IBM's Watson that aims to simulate human cognitive processes, such as understanding natural language, evaluating context, and assessing the reliability of its knowledge.

compliant actuators: Motors and linkages made of flexible, soft materials.

controller: The mechanical, electronic, or computerized part of a robot that converts information provided by sensors into instructions—whether to adjust actuator behavior or overall motion, update maps and other internal models, or provide information for human operators.

CTD sensor: Used by underwater robots to measure conductivity, temperature, and depth.

cybernetics: The study of how dynamic systems are regulated, focusing on issues of feedback, control, and communication.

cyborg: Short for “cybernetic organism”; term used mostly in fiction to refer to a hybrid of biological and nonbiological parts.

direct current (DC): A type of electric current that is unidirectional.

DC brushed motor: An electric motor that uses a mechanical connection between the rotor and the stator to power and reverse the electromagnetic field that drives the rotor.

dead reckoning: A method of navigation in which current position is calculated based on speed, heading, and time from last known location.

degrees of freedom: The number of independent motions available in a joint, structure, or robot, whether the motion is linear (prismatic, translational), angular (revolute, rotational), or spherical (like a ball joint).

deliberative control: A robotic control architecture that emphasizes planning as part of the ongoing operations of the robot.

differential drive: An actuator design in which a robot is steered by creating a difference in the speeds of two independently driven wheels or propellers.

drivability map: A planning model that is continually updated and used to plot the immediate course for a robot.

drone: Any unmanned aerial vehicle (UAV), especially one that can fly autonomously (using GPS or other navigational data) and beyond the line of sight needed for radio-controlled (RC) aircraft.

ducted thruster: A propeller-based actuator used in torpedo-shaped underwater robots to both propel and steer the vehicle.

dynamic soaring: A type of gliding flight in which the animal or vehicle gains velocity and height by harvesting energy from steep wind gradients located near surfaces.

echolocation: Active sensing in which an animal or robot broadcasts a sound, senses the returned echo, and uses the difference between the signal and its echo to calculate the distance and composition of the target. See **sonar**.

end effector: A tool or other distal element on a robotic manipulator; the part of an actuator that interacts directly with the world.

fin-shear actuator (FSA): A biologically inspired design that converts shear (strain from lateral stress in the layers of the fin) into bending.

flash memory: Erasable, reprogrammable computer memory developed in the early 1980s that is relatively fast, durable, and cheap.

flywheel: A rotating mechanical device used to store energy through rotational inertia or, when its center of mass is located off-axis, create vibrations.

gantry robot: Large type of Cartesian coordinate manipulator robot whose degrees of freedom are in translation and at right angles to each other.

global positioning system (GPS): Navigational system using satellites that locate a receiver in coordinates of the world geodetic system.

gyroscope: Mechanical or electronic rotation sensor that is used for maintaining or measuring orientation; often linked with an accelerometer for more precise location in three-dimensional space.

Hall effect sensor: A sensor that converts changes in an external magnetic field into changes in a voltage or other electric output.

homing: Navigating to a previous location, usually the location where a robot started.

humanoid: A robot designed to look and function like a human. See **Actroid**, **android**, and **cyborg**.

hydraulics: Using pressurized liquid to create mechanical motion.

integrated development environment (IDE): Software interface to facilitate programming in a particular language.

inertial measurement unit (IMU): An integrated sensor that measures orientation, velocity, acceleration, and, sometimes, compass heading.

inertial navigation: Dead reckoning using accelerometers and gyroscopes.

kinematics: The study of motion without regard to the forces generating them.

Kismet: Autonomous robotic head capable of social and emotional interactions with humans that was built by Cynthia Breazeal in the late 1990s.

light-emitting diode (LED): Semiconductor-based light source.

Lunokhod 1: The first robotic planetary rover, deployed on the Moon in 1970 by the Soviet Union.

machine learning: Computer programs written to make adjustments to their code, with or without direct feedback from a human, in order to improve performance of the code itself or the robot that the code controls.

manipulator: A robotic arm that grasps and moves objects; also, any stationary robot that has one or more such arms.

mechanoid: A type of robot built without inspiration from biology.

model-based robotics: Design of robots that maximizes the use of internal world models, ongoing and sophisticated planning algorithms, and complex goals and tasks.

Moravec's paradox: Insight from artificial intelligence of the 1980s that the difficult problems in robotics are the easy problems for animals and humans—namely, sophisticated and skillful movements.

motherboard: The main and largest printed circuit board in a computer.

motor: See DC brushed motor, servomotor, and stepper motor.

multimeter: An instrument used to measure electric voltage, resistance, continuity, and current.

muscle wire: A shape-memory alloy used as a linear or spiral motor for thin or distributed actuators.

navigational fix: Location found using external reference points.

negative feedback control: Regulatory process to maintain stable output of the robot using the difference between the desired output and the actual output.

neuroprosthetic: Type of robotic manipulator used as a replacement limb for human amputees and controlled by muscular or neural signals.

nickel-metal hydride battery: A common type of rechargeable battery.

OpenSCAD: Free software for designing three-dimensional objects.

oscilloscope: An instrument used to measure the dynamic properties of electric signals.

photoresistor: Type of sensor that converts changes in light intensity into changes in electric resistance.

pick and place: Fundamental task for robotic manipulators, involving grasping, moving, and then releasing an object.

piezoelectric crystal: Type of material, used as a sensor or an actuator, that transduces a change of shape into a voltage or a change in voltage into a change in shape.

ping sensor: An ultrasonic range finder.

pneumatics: Using pressurized gas to create mechanical motion.

potentiometer: Type of sensor that converts a mechanical rotation into an electric change in resistance; also known as a variable resistor.

probabilistic robotics: A framework for the design of robots that explicitly models the uncertainty inherent in the signals provided by sensors, the models created by controllers, and the movement instantiated by actuators.

programmed article transfer: Computer-controlled pick-and-place task performed by the first commercially successful robotic manipulator, Unimate.

proprioception: Internal sensing of the motion and force of the robot's joints and motors.

prosody: Properties of speech other than the words, often conveying emotional content.

pulse-width modulation (PWM): Method of encoding information in an electric signal by varying the duration of square-wave pulses.

quadcopter (quadricopter): A rotary-winged aircraft with four propellers.

rack and pinion: A type of actuator that converts a motor's rotational motion to a translational output.

range finder: An active sensor that broadcasts ultrasonic or other sound waves and then uses the time it takes for the echo to return to measure the distance to the reflecting surface. See **sonar**.

robot: A type of machine that can be remote controlled, partially autonomous, or fully autonomous as it moves itself or objects in order to carry out tasks. While robots always have controllers and actuators, remote-controlled robots may lack onboard sensors.

robotics: The field of study and inquiry that develops principles and approaches for the design, fabrication, operation, and control of robots.

robot operating system (ROS): An open-source software environment for professional roboticists that integrates different languages, libraries, and solutions.

Roomba: The floor-cleaning robot introduced in 2002 by iRobot; the first fully autonomous home robot to achieve commercial success.

rotor: The part of a DC motor that spins.

sensor: Any device that detects changes in physical properties or energy patterns in the world or the robot and converts those into electric, chemical, or mechanical signals usable by a controller to adjust actuator behavior and overall motion, update maps and other internal models, or provide information for human operators.

sensor fusion: The process of combining information from multiple sensors to create information that is not available from individual sensors alone.

sensorimotor circuits: Parts of a robot or animal that link the information provided by sensors to the motion produced by actuators.

service robot: Any robot built to assist humans, excluding robots involved in manufacturing.

servomotor: An electric DC motor that uses sensory feedback from an internal potentiometer to precisely control position and movement.

shaft encoder: A sensor that converts the rotations of a wheel into an electric signal that represents the speed of the wheel's rotation.

Shakey: The first mobile, autonomous robot controlled by a reprogrammable digital electronic computer.

simultaneous localization and mapping (SLAM): Subfield of navigation in which the robot uses sensors and communication systems to know its position as it creates a map of its surroundings.

Slocum glider: An underwater robot that cycles its buoyancy from negative to positive and converts the resulting vertical motion into horizontal propulsion.

solder: Metal alloys with low melting points used to fuse together separate elements and, in electronics, create secure electric connections.

sonar: Stands for sound navigation and ranging; typically an active sensor system that broadcasts sound waves and then uses the pattern of the returning echo to measure the distance and size of objects.

spectrometer: Any of a large class of instruments designed to measure how the intensity of a physical property varies across a range of frequencies, energies, or masses.

square-cube law: Mathematical principle that describes the ratio of the surface area to the volume of a class of similarly shaped objects that differ in size alone.

Stanley: A fully autonomous car that won the 2005 DARPA Grand Challenge, created by Stanford Racing Team.

stator: The stationary part of an electric motor.

stepper motor: A brushless DC motor that can rotate precisely without the need for sensory feedback.

stereovision: Visual sensing that uses two cameras focused on the same object or scene to provide information about depth, range, and three-dimensional shape.

stigmergy: Indirect communication by altering the environment to signal other agents.

subsumption architecture: A hierarchy of behaviors in the programming of a robot that depends on input from sensors.

swarm robotics: The coordination and control of multiple robots of similar type, operating at the same time and place.

technological singularity: The speculative idea that artificially intelligent machines will exceed the capacity and control of humans and, by virtue of those properties, radically alter civilization in unpredictable ways.

telepresence: Using sensors and instruments on robots to observe and measure without the need for a human to be physically present.

thermocouple: A sensor that converts external temperature into an electric signal.

three laws of robotics: Introduced by Isaac Asimov in the short story "Runaround" (1942). First law: A robot may not injure a human being or, through inaction, allow a human being to come to harm. Second law: A robot must obey the orders given it by human beings, except where such orders would conflict with the first law. Third law: A robot must protect its own existence as long as such protection does not conflict with the first or second laws.

thrust: A force that propels an animal, vehicle, or robot.

time of flight: Duration for a signal to travel from its source to a sensor.

torque: Force applied through a moment arm to rotate or twist.

trilateration: Method for determining location using the mathematics of circles, spheres, and triangles.

transducer: Any device that converts energy in one form to energy in another. For example, a touch sensor transduces the kinetic energy of movement into electric signals; a motor transduces electric signals into kinetic energy.

Turing test: Also known as the imitation game, a type of behavioral assay proposed by Alan Turing to determine if a machine can demonstrate linguistic behavior indistinguishable from that of humans.

uncanny valley: Hypothesized by Masahiro Mori, a sudden loss of affinity that humans can feel as masks or robots become almost, but not quite, human.

visual odometry: Method for using successive images of a robot's surroundings to measure its displacement.

voltage divider: A simple circuit that uses two resistors, with resistance Z_1 and Z_2, to reduce the input voltage, V_{in}, to a lower output voltage, V_{out}, by the following equation:

$$V_{out} = V_{in} \frac{Z_2}{(Z_1 + Z_2)}$$

voltmeter: A device that measures the voltage across two points in an electric circuit.

wheg: An actuator combining the rotational simplicity of a wheel and the obstacle-clearing capacity of legs.

zero-moment point: Created by Miomir Vukobratović, a mechanical concept that calculates a critical position of legs and body in dynamic motion in order to preserve stability of a humanoid robot.

Answers

Lecture 1

No questions

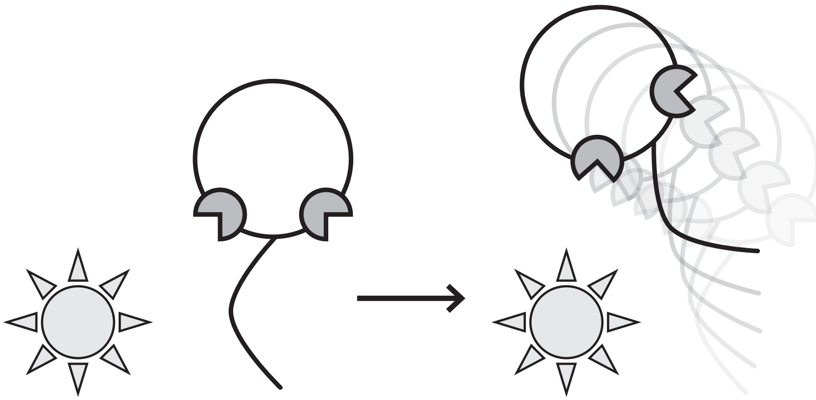
Lecture 2

1. One trade-off of walking on two legs instead of four is that we start to become lousy at moving around in trees. We trade agility in trees for a toolmaking life on the ground. There is no animal—or robot—that can do everything well. A robot may be designed to do almost anything, but no single robot can do everything.

2. The Tadro on the right (with sensors pointed away from the light) would swim away from the light. Calculate direction to the light as follows: Take the difference of the right sensor's intensity and the left sensor's intensity. For the Tadro on the left, with sensors facing forward, a positive difference means that the light is to the right. To move toward the light, that positive difference is then converted into a signal to the motor to turn Tadro to the right. A negative difference means that the light is to the left, and the instruction would be to turn Tadro to the left. A difference of zero means that the light is straight ahead, so Tadro should stay the course.

The Tadro on the left will swim straight as long as both sensors are pointed away from the light. Once it passes the light, the sensor on the left will register higher intensity than the one on the right, and Tadro will turn to the left. But Tadro is moving away from the light and not toward it. So, as it continues to arc left, the right sensor moves into position to receive light that is nearly the same intensity as the left sensor. The difference becomes zero, and this Tadro swims away.

Performing thought experiments like these is the equivalent of using our minds as robotic simulators. Thought experiments are often the first simulation that we run when we start to design a robot. This thought



experiment shows that a reconfiguration of the body can "reprogram" the robot's behavior without touching the computer software on the robot's controller.

Lecture 3

1. A servomotor is a DC motor with a sensor and a controller. What makes servomotors so useful is that the sensor, a potentiometer, feeds back the position of the DC motor to the controller.

2. A DC motor would be sufficient for the tank track drive, "a," while a servomotor would be sufficient for "c." A stepper motor is best for "b," to precisely move and then hold position of a robotic arm. Stepper motors move at slow speeds, provide higher torque than a servomotor, and can be precisely positioned, so that's best for "b."

3. The answer is "c," light-emitting diode (LED). When an LED emits light, it is changing the world by altering the local energy patterns of light. In this course, we use a narrower definition: An actuator is the system that creates movement, such as a wheel and its drive motor and transmission. The broad definition, though, complements the way that sensors work. Sensors detect changes in the energy patterns in the world. Actuators create energy changes in the world.

Lecture 4

1. In answer “a,” the robot would be spinning its wheels, trying to move forward but not making progress because it is up against an object. In answer “c,” the robot would turn into a wall, rather than away from the wall as in the correct answer, “b.”

2. The difference in light intensity between the two sensors indicates the bearing of the light. Tadro calculates the difference between the right and left sensors: right − left = x. The controller on the Tadro is set up to use a positive difference as a signal to turn to the right. With a negative difference, right − left < 0, Tadro will turn to the left. Note that if there is no difference, right = left and right − left = 0, then Tadro keeps swimming straight.

That said, we create a one-eyed Tadro in this course that can work with a single light sensor. The trick with a single sensor is to make the amount of turning proportional to the amount of light reaching the sensor—so keep moving and rotate slowly in a spiral motion.

Lecture 5

1. This is a trick question. All three of these answers are viable options, depending on the task that your robot is trying to accomplish. The answer “a” might be the perfect set of instructions for a robotic plane: Keep flying straight ahead, at this altitude, until further notice. The answer “b” might be perfect for a search-and-rescue robot that is looking for a wandering child: Sit and wait to see if the child wanders within sensor range. The answer “c” could also be useful for a search-and-rescue robot, but the problem with random searches over large spaces is that they don’t systematically cover the area. This would work best in an enclosed space, such as a home, during a fire, when the robot is searching for the family cat, which might be frightened and hiding under the sofa.

2. Build five movement modules, and then combine those modules with sensor triggers to create behaviors. Use IF-THEN-ELSE to have the robot decide when to execute the different behaviors in a fixed-priority hierarchy.

1. Create five movement modules:

   Cruise. Move straight ahead.
   Escape_right. Move straight backward for one second, and then turn to the right 45 degrees and stop.
   Escape_left. Move straight backward for one second, and then turn to the left 45 degrees and stop.
   Avoid_right. Turn 30 degrees to the left.
   Avoid_left. Turn 30 degrees to the right.

2. Create behaviors:

   Escape: If left or right bump sensor is triggered, then escape_left or escape_right.
   Avoid: If left or right IR is triggered, then avoid_left or avoid_right.
   Cruise: If no sensors are triggered, then cruise.

3. Create the fixed-priority decision-making part of the code:

```
If left or right bump sensor is triggered, then escape left or right
    Else
        If left or right IR is triggered, then avoid left or right
            Else
                Cruise.
```

The trick is to nest one IF-THEN-ELSE statement inside another. This creates a fixed-priority hierarchy where one behavior, Escape, overrides any other. As long as Escape isn't triggered, then Avoid overrides Cruise.

Lecture 6

1. Answer "b," a compass heading to the next waypoint, gives the minimum amount of information that you need. In calm waters, if you know the heading you need to sail, then as long as you stay on that course, you'll get to your waypoint. Map coordinates alone won't get you where you need to go unless you know where you are, too.

2. The best answer is usually "c," wait until you run into something that's not on your map and then add it to your map. Ideally, we'd like to keep track of everything in our world (answer "a"), but that isn't efficient: Depending on how quickly your world changes, you might spend all of your time updating the model and very little time getting your work done. There is a trade-off between the accuracy of your model and the time that you have to perform your task. While monitoring a subset of objects and people might be a good compromise, that approach will only work if you've correctly identified the likely objects and people that do not change over time.

3. Model-based systems have the benefit of allowing the robot to undertake tasks that have specific, complicated goals that are specified in the software program. Progress toward that goal can be monitored by an external observer. In contrast, the cost of a pure behavior-based system is that the robot's goals are specified only in terms of how to change behavior in response to changes in sensory input. This benefit-cost pair can be turned on its head. Pure model-based systems have the cost that they cannot respond in a timely fashion to unforeseen circumstances, so they tend to have high runtime failure rates. In contrast, the benefit of a pure behavior-based system is that it is very robust, able to keep on moving in the face of rapidly changing circumstances.

Lecture 7

No questions

Lecture 8

1. In order to close the functional loop to make a behaviorally autonomous robot, you need to add sensors and a controller, and you need all the components working together.

Lecture 9

1. For “a,” you could modify a Roomba by putting a DropCam on it. For “b,” you could modify a quadcopter that has a camera to avoid walls. For “c,” you need a design from scratch. You first have to get through an ice layer that might be miles thick. Once through to the ocean, the robot needs to be able to maneuver in water. Do you use the same robot to do it all, or do you use a team of robots? Do you melt your way to the ocean using a small nuclear reactor? There really is no off-the-shelf solution. You’ll need a team of crack engineers, tons of money, and a ride from NASA.

2. While RayBot is energy efficient, able to glide, and able to stay on station for long periods of time, it doesn’t have the motors or the energy supply to accelerate rapidly or cruise at high speeds.

3. There are other important features, but the following are three.

    (1) Do you have a robot that people want to buy? You need a thorough marketing study. You’d want to see the specific questions that were used on a survey, if they took that approach. A good survey asks the same question in a variety of ways.

    (2) If your robot joins an existing category, how can you differentiate your product? Companies making the next kind of vacuum cleaner, for example, have an identity and shelf-space problem. Why buy their vacuum cleaner when the Roomba is a proven commodity? Why should a vendor stock your new machine when he or she has no idea if someone will buy it? The new company needs a very clear vision about how they will grab attention, induce a sale, and then build loyal customers.

(3) Having a management team that includes an experienced and creative chief of technology and an experienced and creative chief executive officer. Most of the time, it's difficult to find one person who can do both well. Colin Angle, CEO of iRobot, has talked about his difficulties in transferring from being a tech guy into being what he calls a vacuum salesman.

Lecture 10

1. Robots are best for tasks that require extended attention, endurance, and reliability. Robots don't sleep, get hungry, or get distracted.

2. Even though it is very difficult to do, answer "c," running an internal model of oneself, has a number of benefits over the other two answers. First, if you are making plans based on a model of yourself working in the world, then that allows for different conditions to apply each time you attempt the task. For example, when the library is crowded with people before exams, the probability of a robot shelving books in a timely manner might be low because of all the traffic. So, rather than saying that you can never complete that task (answer "a"), you rule it beyond your limits at that time, based on those conditions. Second, your self model can explore different plans and compare their probabilities of success. For example, two plans might vary only in the sequence in which you visit different book drop-off points. Given the state of the world (and perhaps your batteries, or what you are already carrying) at the time, one sequence that didn't work before might offer a solution now.

Lecture 11

1. The robot must have enough kinematic degrees of freedom to reach each block and place it on the display shelf. It must also have a gripper to grasp and then release each block.

Unlike "blind" pick-and-place robots, where the robot doesn't sense the position of the object, this block-spelling robot must know the position of each block and the letter represented, which it does by (1) knowing the angle of each of its joints, and it uses feedback control with angular position sensors; (2) having force-feedback sensors on the grippers to sense if the object has been gripped; (3) having a world model that maps the position of each letter block at all times.

Lecture 12

1. A hands-on approach will give you a very clear answer here: Pick up Roomba while it is on, and manually touch the bump sensor lightly and then more firmly, observing the direction in which the wheels are spinning. Note that the moment you pick Roomba up, it will think that it has detected an edge, so you'll have to put a little masking tape on the infrared sensors that do the edge detection. But now that you are holding it, you can approach the bumper from below with your finger and know that forward-looking infrared detectors aren't at work here.

Lecture 13

1. All of the following would help us build a fully autonomous robotic surgeon: (1) a planning component of the controller that is forwarding the next steps to an independent surgeon or surgical AI before action is taken as extra screening precaution; (2) a model-based controller that uses and updates a three-dimensional map of each patient; and (3) algorithms and new types of sensors to identify tissue, organ, and cell types.

2. You might opt for stronger joint motors, because most small arms like the DEKA Arm are not very powerful. You might want to improve or increase the number or types of sensors on the arm so that the human knows better—beyond visual feedback—what the arm is doing. Or, you could decide to increase the autonomy of the arm so that it interacts with the human in more complex ways. For example, what if the arm, using signals from an eye-tracker system in the frame of your eyeglasses, could anticipate what you are about to do based on the direction of your gaze? Then, it could put itself in a state of preparation. Anticipation and preparation require that the arm have a model of itself, a model of you, and a model of how you interact.

Lecture 14

1. A hybrid controller that combines features and fail-safes from both systems would be best. A behavior-based controller creates important reflexes, such as avoiding obstacles, staying on the road, and reading signs. A model-based controller uses comprehensive road maps coupled with GPS and roadside scanning to navigate carefully.

2. Radar. Here is what Dickmann has to say in an article about the Mercedes robotic car (listed in the Suggested Reading for this lecture): "Unlike optical systems, radar operates well no matter what the weather, working as it does with microwaves."

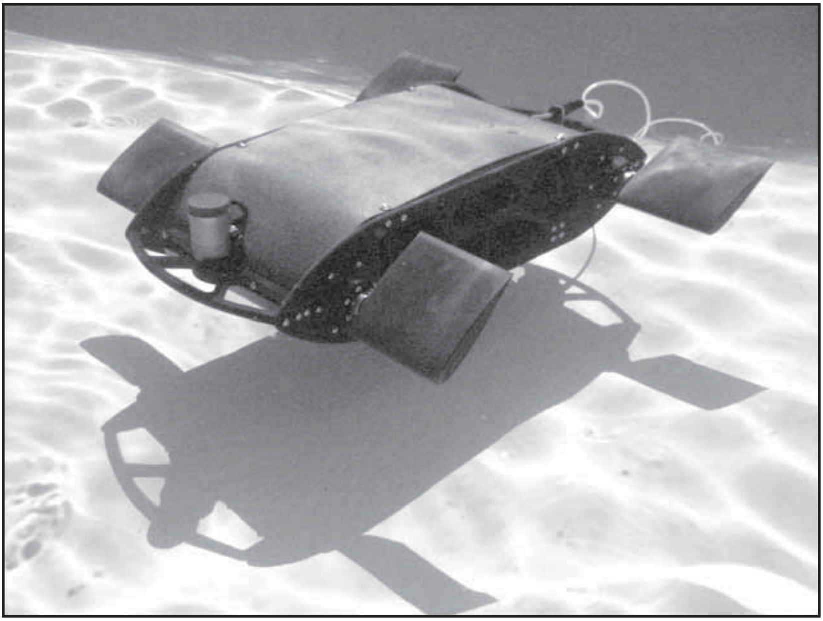
Lecture 15

1. Notice that the question didn't say if the autopilot was engaged. If it is engaged, then you have a closed-loop system. Most commercial aircraft are a classic case of "mixed autonomy," both open-loop *and* closed-loop systems. Keep in mind that mixed autonomy systems can operate in parallel, such as when you steer your car but the cruise control has autonomy over speed, or in series, such as when the plane's pilot steers during takeoff but then hands the steering over to the autopilot.

Lecture 16

1. Most of the time, we try to build underwater robots that are passively "roll stable." This means that answer "b" is better, even though "a" is also feasible. Think about Robot Madeleine (see image).

   She is highly maneuverable in roll, pitch, and yaw. She can roll quickly by having the two flippers on one side generate a downward thrust while the two flippers on the other side generate an upward thrust. She can pitch quickly by having the front and back flippers generate vertical thrust in opposition. She can yaw quickly by



© John Long.

having the left-side flippers generate a forward thrust while the right-side flippers generate a rearward thrust. At the same time, Robot Madeleine's flattened body creates resistance to rotation in roll and pitch that makes her stable in these directions when she isn't actively turning.

2. While all of these are true, "b" is probably the most vexing problem. The challenge is most difficult in saltwater, where the salts in solution increase electric conductance and, hence, the ability to short-circuit electronics. Building a watertight robot is difficult because you need access to the insides of the robot for repairs. The joints and seams that make that possible then become the places most likely to leak.

Lecture 17

1. All three answers offer potential advantages. Complex missions, such as how to explore another planet, benefit from having humans in the loop. Telepresence enhances everything from the entertainment value of recreational robots to the research value of rovers on Mars. Checking on the robot when we do remote exploration at the bottom of the ocean or on another planet is more complete when we keep humans in the loop.

Lecture 18

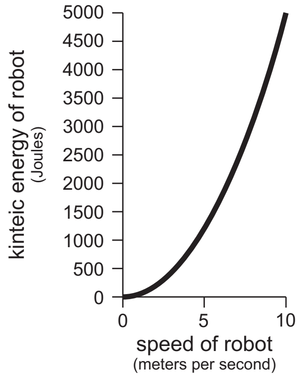
1. While all of these are true, the greatest vulnerability in time of war comes from the humans in the control loop. With weapons, the first rule is to not kill your own troops. In the laws of war, eliminating the deaths of noncombatants is paramount. The navy has an advantage over the army in that at least in open-ocean conditions, the number of noncombatants is likely to be small. The next step might be to have the commanding officer decide the level and type of autonomy to give to the system and when to withdraw each increment of autonomy. For example, you could imagine that if you suddenly were taking heavy fire, you could more quickly mobilize your defenses using the fully autonomous system.

Lecture 19

1. First, calculate the kinetic energy in each case. The microbot has a mass of 0.01 kilograms and a velocity of 1 meter per second. If we square the velocity, we still get 1, and the product of 1 and 0.01 is the kinetic energy of 0.01 joules, where joules are the SI unit for energy.

For the cheetah, its mass is 100 kilograms. Square the velocity of 10 meters per second and we get 100. The product of that square and the mass is 10,000 joules. So, even though the cheetah is only going 10 times as fast as the microbot flyer when they impact, the kinetic energy of the cheetah is 1 million times greater.

2. A robot has to use its onboard energy source (battery or fuel) and convert that potential energy into kinetic energy. This is where actuators get involved and the square of velocity comes back to bite us. To go a little bit faster takes a lot more kinetic energy. More kinetic energy requires more fuel.



Look at the graph. The mass of our robot is 100 kilograms. The speed increases by 2 orders of magnitude, from 0.1 to 10 meters per second. Over that same range of speed, the kinetic energy increases from 0.5 to 5,000 joules, or 5 orders of magnitude.

The trade-off is that when you go faster, you run out of fuel faster. This might seem like common sense, but there's more physics fun here: Because you use more fuel to move faster, you don't move as far. That's the real trade-off. For a given amount of fuel, you can either choose to move fast or far.

Lecture 20

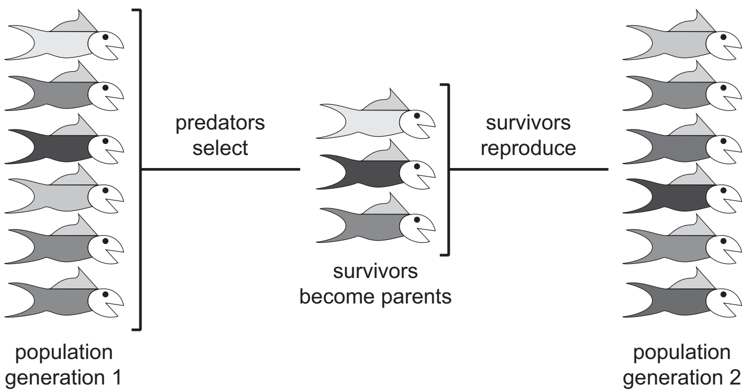
1. The best opportunity for stigmergy is with scenario "c." One of the huge challenges in any search-and-rescue situation is to quickly and systematically cover an area. If these robots are working in snow, and perhaps listening directly below for any signs of life, then the simplest thing to do is to have them keep a set distance away from any other robot's trail. That trail could be the one that they made previously, or it could be one from another robot.

   We can rule out answer "a" because Curiosity is not likely to double-back on its explorations. If you leave yourself a note, you have to go back and read it to make any sense of it. The one-way trip for the fleet of self-driving cars and trucks looks, at first glance, like the same situation, but with many robots on the same road, they could help each other out. For example, suppose that every time a robot encountered a dangerous pothole, it placed a little radio beacon to mark the danger.

Lecture 21

1. This is a tough, open-ended question. If you are physically minded, you could think about life-forms being local dis-entropy machines, able to harvest energy so that they can construct order in the form of complex bodies. If you are biologically minded, you might focus on the idea of reproduction and the transmission of information for self-production from generation to generation. Or maybe you are putting this all together and thinking about the different kinds of autonomy that we see in life-forms: behavior, reproduction, development, and evolution. Would you say that a group of robots fully possessing all those types of autonomy would be functionally indistinguishable from life itself?

2. The correct answer is "b": Designs that perform the best are combined and altered to create a new group of nonidentical robots. This isn't phrased in the usual way that you might find in a biology textbook. Also, the response didn't specify the nature of the judging, but in nature, what happens often is that predators select the winners (see figure). Other kinds of selection forces may occur: Competition for resources (energy) can make some individuals winners and others losers. The winners are the survivors.



Lecture 22

1. The simplest feature to track is the direction of a human's gaze, answer "a." We know from many psychophysical experiments that the eyes of humans automatically track to features and movements in the environment. Social robots bank on this and capture a human's attention using their own movements, sounds of certain types, and even displays of light or images.

2. Carry the pills. Remind the patient. Follow the patient, or wake up the patient. Carry around water. Monitor the dispensing of the pills. Monitor the taking of pills. Communicate with the physician any irregularities in dosages or timing of the pills. Track the patient verbally and with visual record keeping for side effects and progression or retreat of the illness. Wireless monitoring of wearable devices to track body temperature, sleep, hydration, blood pressure, and heart rate.

Lecture 23

1. Automatons from hundreds of years ago had very clever actuators, bodies, and energy sources. But what tended to be missing were sensors (apart from the on/off switch, which wouldn't contribute to behavioral autonomy). The functional loop would be wide open. As far as we know, Leonardo never solved the sensor problem.

But perhaps his biggest challenge was not being aware of electricity—not only a source of energy but also the currency for information throughout the modern robot. All sensor signals are converted to patterns of electricity. All signals to the actuators are given as patterns of electricity. All computations on the controller are carried out by miniaturized electric switches.

Lecture 24

1. Agriculture ("d"), retail services ("e"), personal and home health care ("f"), military and defense systems ("g"), and first-responder services ("h") are all areas that are still dominated by humans performing manual labor. Because the history of technology is in many ways the story of how we use tools and machines to reduce the need for manual labor, chances are that trend will continue. Embedded robotic systems will be the tools of change, because they are psychologically easier for us to accept than a humanoid because we simply don't see them. Out of sight is out of mind. For psychological reasons as well, these embedded robotic systems won't be called robots. Instead, look for these euphemisms: adaptive, assistive, automatic, cognitive, intelligent, and smart.

   The other options—transportation ("a"), assembly and manufacturing ("b"), and medical devices ("c")—are already dominated by embedded robotic systems.

2. Any of the above! Commercial robotics is, fundamentally, like any other business. You have to figure out what people want, what they are willing to purchase, and how much they are willing to pay. That's called market analysis, which you'll have to perform when you create your first business plan.

Bibliography

Angeles, Jorge, and Frank C. Park. "Performance Evaluation and Design Criteria." *Springer Handbook of Robotics*, 229–244. Berlin, Heidelberg: Springer, 2008. This is a formal treatment of the mechanical design of robots. The emphasis on also designing the workspace is very helpful.

Anonymous. "An Automatic Block-Setting Crane." *Meccano Magazine* 23, no. 3 (March 1938): 172. Available at http://www.nzmeccano.com/MMviewer.php. This is the first record of Griffith P. Taylor's "Robot Gargantua." Taylor's robot is often considered to be the first programmable robot.

Banzi, Massimo. *Getting Started with Arduino.* Sebastopol, CA: Make Books, 2011. Within a few minutes of referencing chapters 1 through 4, you can load code onto your Arduino and hook up sensors and motors. The Arduino philosophy gets you to work making and tinkering.

Bekey, G. A. *Autonomous Robots: From Biological Inspiration to Implementation and Control.* Cambridge, MA: MIT Press, 2005. Chapter 1 is an introduction to robotic autonomy that includes explanations of control theory and the parts needed to build an autonomous robot.

Bohm, Harry, and Vickie Jensen. *Build Your Own Underwater Robot, and Other Wet Projects.* Vancouver: Westcoast Words, 1997. This do-it-yourself book will help you avoid the catastrophic failures that only water can bring.

Braitenberg, Valentino. *Vehicles: Experiments in Synthetic Psychology*. Cambridge, MA: MIT Press, 1986. Learn to use the simplest approach possible to understand how robots behave with the famous thought experiments found in chapters 1 through 3. Braitenberg's idea of a "vehicle" is that all you need to create an autonomous agent is a body with a sensor attached to a motor.

Breazeal, Cynthia. “Cognitive Modeling for Biomimetic Robots.” Chapter 9 in *Biologically Inspired Intelligent Robots*. Edited by Yoseph Bar-Cohen and Cynthia Breazeal. Bellingham, WA: SPIE, The International Society for Optical Engineering, 2003. Breazeal is one of the pioneers of social robotics, and in this chapter, she details both the design of Kismet and the basic principles behind building emotions into robots.

Brooks, Rodney. *Flesh and Machines: How Robots Will Change Us.* New York: Pantheon Books, 2002. Brooks is writing for a general audience, so he gives an approachable history of behavior-based robotics. In chapter 3, Brooks explains the design of Grendel, a legged robot built for planetary exploration. Grendel begat Sojourner, the first autonomous rover on Mars.

Burdet, E., D. Franklin, and T. E. Milner. *Human Robotics: Neuromechanics and Motor Control.* Cambridge, MA: MIT Press, 2013. In chapter 1, the authors explain how humans control the motion of their limbs and body as well as how biological sensors and actuators are the basis for engineering neuromechanical arms. In chapter 11, they explore neurorehabilitation and robotics.

Bureau of Ordnance, Department of Navy. *The Whitehead Torpedo, U.S.N., 45 c/m. x 3.55 m. Mark I, Mark II, Mark III, and 45 c/m. x 5 m. Mark I. General Description.* Naval Torpedo Station, document no. 1598, 1898. Available online at http://maritime.org/doc/whitehead/. The first 45 pages take you through Mark I. Make sure to look at the plates, which offer some lovely diagrams of the internal mechanisms and external parts.

Ciuti, G., A. Menciassi, and P. Dario. “Capsule Endoscopy: From Current Achievements to Open Challenges.” *IEEE Reviews in Biomedical Engineering* 4 (2011): 59–72. You’ll have to either pay to purchase this article or get it through your library. The hassle is worth it if you want to read a good overview of this emerging robotics discipline.

Clancey, William. *Working on Mars: Voyages of Scientific Discovery with the Mars Exploration Rovers.* Cambridge, MA: MIT Press, 2012. The scientific mission of Spirit and Opportunity is explained in chapter 4. In chapter 6, the rovers are on Mars and exploring.

Cook, Gerald. *Mobile Robots: Navigation, Control and Remote Sensing*. Hoboken, NJ: John Wiley, 2011. Chapter 4 is a practical exposition of getting your robot to navigate; it includes coordinate systems, GPS, and dead reckoning.

Craig, John. *Introduction to Robotics: Mechanics and Control.* 3rd ed. Upper Saddle River, NJ: Pearson Prentice Hall, 2005. In chapter 1, the basic problems and principles are laid out for kinematics and dynamics of robotic manipulators.

Dawson, Michael, Brian Dupuis, and Michael Wilson. *From Bricks to Brains: The Embodied Cognitive Science of LEGO Robots*. Edmonton, AB: Athabasca University Press, 2010. Chapter 6 is a great analysis of Grey Walter's tortoises. Dawson includes instructions for building a tortoise using the LEGO™ Mindstorms™ robot kit. Chapter 7 includes the computer code for the tortoise.

Devol, George C. 1961. Programmed article transfer. U.S. Patent 2,988,237. This is a landmark patent for industrial robotics. The system Devol describes does not look like a robot, so you'll have to use your detective skills to figure out how sensors and actuators are coordinated.

Dickmann, Jurgen, Nils Appendrot, and Carsten Brenk. "How We Gave Sight to the Mercedes Robotic Car." *IEEE Spectrum*. http://spectrum.ieee.org/transportation/self-driving/how-we-gave-sight-to-the-mercedes-robotic-car. Posted July 24, 2014. Dickmann, one of the pioneers of autonomous navigation in cars, explains the development of radar systems and the importance of combining different types of sensor systems.

Dym, Clive, Patrick Little, and Elizabeth Orwin. *Engineering Design: A Project-Based Introduction*. 4th ed. Hoboken, NJ: John Wiley, 2013. Chapters 1 and 2 emphasize the process of designing, including vocabulary and the importance of asking questions. While not for robotics specifically, the approach to engineering design is completely relevant. Chapters 4 through 6 help us clarify the design by focusing on function. Limits and constraints help frame what is possible.

Floreano, Dario, and Claudio Mattiussi. *Bio-inspired Artificial Intelligence: Theories, Methods, and Technologies*. Cambridge, MA: MIT Press, 2008. This is a good complement to Kernbach's overview of the field ("Introduction to Collective Robotics: Reliability, Flexibility, and Scalability"). Floreano and Mattiussi introduce some mathematical details and provide examples of robotic systems in which collective algorithms have been implemented.

Grundfest, W. S., J. W. Burdick, and A. B. Slatkin. 1994. Robotic endoscopy. U.S. Patent 5,337,723. This patent set the stage for many later robots that go into the body as either miniaturized vehicles or robotic arms.

Hanson, David, Daniela Rus, Steven Canvin, and Gernot Schmierer. "Biologically Inspired Robotics Applications." Chapter 10 in *Biologically Inspired Intelligent Robots*. Edited by Yoseph Bar-Cohen and Cynthia Breazeal. Bellingham, WA: SPIE, The International Society for Optical Engineering, 2003. By focusing on the modules and task-specific applications of robots, this chapter provides several different perspectives on design.

Hickman, Ryan, James Kuffner, James Bruce, Chaitanva Gbarpure, Damon Kohler, Arshan Poursohi, Anthony Francis, and Thor Lewis. 2014. Shared robot knowledge base for use with cloud computing system. U.S. Patent 8,639,644 B1. This patent laid the groundwork for cloud robotics and for a collective robotic "mind" (database).

Hornyak, Timothy. *Loving the Machine: The Art and Science of Japanese Robots.* New York: Kodansha International, 2006. Chapter 9 gets us under the skin of the Actroid androids to understand how they are actuated with pneumatic systems, plus the metaphysical questions that drive creator Hiroshi Ishiguro's development of lifelike robots.

Huang, Hui-Min, Elena Messina, and James Albus. *Autonomy Levels for Unmanned Systems (ALFUS) Framework. Volume II: Framework Models*. Version 1.0. NIST Special Publication 1011-II-1.0, 2007. Prepared for the U.S. military, this document grapples with the complexities of autonomy. Their framework uses three parameters for design decisions: human independence, mission complexity, and environmental complexity.

iRobot. "iRobot© Owner's Manuals and Quick-Start Guides." Owner's Manual for Roomba. http://homesupport.irobot.com/app/answers/detail/a_id/843/~/irobot%C2%AE-owners-manuals-and-quick-start-guides. While this might not seem like exciting reading, it actually is pretty cool to look at the capacities of these robots, how they work, and how you can do some simple troubleshooting and repair.

Jarnot, Charles. "History." Chapter 1 in *Introduction to Unmanned Aircraft Systems*. Edited by Richard Barnhart, Stephen Hottman, Douglas Marshall, and Eric Shappee. Boca Raton, FL: CRC Press, 2012. Jarnot provides a concise, detailed history of UAVs, including some of the context behind Sperry's aerial torpedo patent of 1917.

Jones, J. L., A. M. Flynn, and B. A. Seiger. *Mobile Robots: Inspiration to Implementation*. 2nd ed. Wellesley, MA: AK Peters, 1999. Chapter 2 contains the instructions for building your own Tutebot and a great explanation of the workings of the electronics. Chapter 5 reviews a range of sensors and explains how they work. Chapter 6 offers insights into actuators built to move robots around, including wheels and legs. Chapter 7 explains how DC and servomotors work and how to size them properly for your robot.

Karvinen, Kimmo, and Tero Karvinen. *Make: Getting Started with Sensors.* Sebastopol, CA: Maker Media, 2014. Chapters 1 and 2 will get you building simple sensor circuits immediately. And you will understand them by building them.

Kernbach, Serge. "Introduction to Collective Robotics: Reliability, Flexibility, and Scalability." Chapter 1 in *Handbook of Collective Robotics: Fundamentals and Challenges.* Edited by Serge Kernback. Singapore: Pan Stanford Publishing, 2013. Collective robotics, the overarching field that includes swarm robotics, examines systems that are cooperative, networked, swarming, and organized around the principle of small-world networks.

Long, John H., Jr. “Biomimetics: Robotics Based on Fish Swimming.” In *Encyclopedia of Fish Physiology: From Genome to Environment*, edited by A. P. Farrell, 603–612. San Diego: Academic Press, 2011. Long takes you through the detailed process of designing and building a fishlike robot and reviews some of the recent history in biomimetic robotics.

———. *Darwin’s Devices: What Evolving Robots Can Teach Us About the History of Life and the Future of Technology.* New York: Basic Books, 2012. Long explains why and how to evolve robots, focusing on how this helps biologists answer questions about how extinct animals evolved. The book explores notions of intelligence in animals and robots and how you can design robots that mimic life-forms.

Monk, Simon. *Hacking Electronics: An Illustrated DIY Guide for Makers and Hobbyists*. New York: McGraw-Hill Education, 2013. Chapters 1 and 2 instruct how to assemble a kit with tools and electronic parts. Stripping wire and soldering, invaluable skills for building or hacking, are also introduced. Chapter 2 introduces electricity and circuit diagrams. In Chapter 2, Monk explains how to read a circuit diagram. Chapter 3 gets us hacking and making circuits. Monk’s push-light hack can be modified to make a light-sensitive circuit that turns a tank track actuator into a light-sensitive robot.

———. *Programming Arduino: Getting Started with Sketches.* New York: McGraw-Hill. With more detail than Banzi’s *Getting Started with Arduino*, this is a great companion. Chapters 1 through 3 do the important work of teaching you some of the formal structures of programming in general and the C computer languages in particular.

Mori, Mashimori. “The Uncanny Valley.” *IEEE Robotics & Automation Magazine* (June 2012): 98–100. Translated by Karl F. MacDorman and Norri Kageki. Available at http://spectrum.ieee.org/automaton/robotics/humanoids/the-uncanny-valley. This is an influential paper in robotics, with its ideas spreading throughout the worlds of entertainment and psychology. Social robotics, and recognition of the importance of our emotional relationship with robots, begins here.

Murphy, Robin. *Introduction to AI Robotics.* Cambridge, MA: MIT Press, 2000. Chapter 9 presents technical details about the "where am I?" problem of topological navigation.

Nourbakhsh, Illah. *Robot Futures*. Cambridge, MA: MIT Press, 2013. Chapter 6 is the bright spot in the book; it focuses on the possibilities that robotics provides to empower individuals.

Pfeifer, Rolf, and Josh Bongard. *How the Body Shapes the Way We Think: A New View of Intelligence.* Cambridge, MA: MIT Press, 2007. Chapters 1 and 2 offer a great introduction to the concept of embodiment: Behavior results from the physical interactions of robots and their environments.

Pinheiro, A. V., D. Han, W. M. Shih, and H. Yan. "Challenges and Opportunities for Structural DNA Nanotechnology." *Nature Nanotechology* 6 (2011): 763–772. This review provides the background to understand how we can build DNA spiders and molecular "robots."

Reynolds, Craig. "Flocks, Herds and Schools: A Distributed Behavioral Model." *SIGGRAPH '87: Procs. 14 Ann. Conf. Computer Graphics Interactive Tech*. (1987): 25–34. Available at http://www.cs.toronto.edu/~dt/siggraph97-course/cwr87/. While written for workers in computer graphics, this seminal paper in swarm robotics enumerates the three principles for which Reynolds and his Boids have become famous: (1) collision avoidance, (2) velocity matching, and (3) centering.

Rosheim, Mark. *Leonardo's Lost Robots*. Berlin: Springer Science & Business Media, 2006. In chapter 3, Rosheim reconstructs Leonardo's robot knight. This reads like a detective story, and it's a great way to appreciate Leonardo's brilliance and Rosheim's creativity.

———. *Robot Evolution: The Development of Anthrobotics*. New York: John Wiley & Sons, 1994. Even though much has changed since this book was published in 1994, the problems of building humanoids have not. As Rosheim steps through the history of humanoids, which he calls "anthrobots," he also spells out design goals and principles.

Scherz, Paul, and Simon Monk. *Practical Electronics for Inventors.* 3rd ed. New York: McGraw-Hill Education, 2013. The front flyleaves and chapters 1 and 2 provide more detail than Monk's *Hacking Electronics*. Chapter 2 is a serious introduction to electricity and electronics. Chapter 7, "Hands-On Electronics," is a must-read if you are going to build robots from scratch. Foremost are important precautions about safety. Learn how multimeters and oscilloscopes work. This is full of details about tools and designing your workspace that can keep you occupied for hours.

Singer, P. W. *Wired for War: The Robotics Revolution and Conflict in the 21st Century.* New York: Penguin Press, 2009. Chapters 1 through 5 provide an excellent introduction to robotics in war, including a brief history.

Smithers, T. "Autonomy in Robots and Other Agents." *Brain and Cognition* 34, no. 1 (1997): 88–106. This paper reviews uses of the term "autonomous." While we use this term to mean that a robot has self-control, Smithers argues that true autonomy requires a robot that can learn. He calls this type of autonomy "self-ruling."

Sperry, Elmer. 1931. Wireless-controlled aerial torpedo. U.S. Patent 1,792, 937. Filed in 1917, this patent covers what today we would call radio-controlled planes. Introduced with military applications in mind, Sperry's other innovation was the gyroscopic sensors that could be used for feedback to have the plane fly autonomously.

Stamp, Jimmy. "A Brief History of Robot Birds: The Early Greeks and Renaissance Artists Had Birds on Their Brains." *Smithsonian Magazine*, May 22, 2013. http://www.smithsonianmag.com/arts-culture/a-brief-history-of-robot-birds-77235415/#qlk1X53P8jVIDUpX.99. Stamp takes us back to 50 B.C.E., when the Greek mathematician Archytas of Tarentum is purported to have built a flying dove. We learn of Hero of Alexandria's designs for pneumatically powered mechanical animals.

Thrun, S., W. Burgard, and D. Fox. *Probabilistic Robotics*. Cambridge, MA: MIT Press, 2005. Available at http://mitpress.mit.edu/books/probabilistic-robotics. The central mission of this new approach in robotics is to quantify uncertainty. Chapter 1, available for free, sets the stage.

Thrun, Sebastian, et al. “Stanley: The Robot That Won the DARPA Grand Challenge.” *Journal of Field Robotics* 23, no. 9 (2006): 661–692. Available at www.robotics.usc.edu/~maja/teaching/cs584/papers/thrun-stanley05.pdf. Written shortly after Thrun and Stanford Racing won the 2005 DARPA challenge, this paper is now a classic in robotics.

U.S. Department of Defense. *Unmanned Systems Integrated Roadmap FY2013–2038.* Open Publication Reference Number: 14-S-0553. This offers a fascinating look at what the U.S. Department of Defense is making public about their plans for robots in defense systems.

van de Molengraft, Rene. “Final Project Report: RoboEarth.” Available at http://roboearth.org/wp-content/uploads/2011/03/document.pdf. RoboEarth can be credited with creating, in a formal and systematic way, the idea of cloud robotics.

Vernon, David. *Artificial Cognitive Systems: A Primer.* Cambridge, MA: MIT Press, 2014. Chapter 9 offers a concise and formal overview of cooperating, with a focus on social interaction, helping and being helped, collaboration, joint action, and attention.

Vogel, Steven. *Comparative Biomechanics: Life’s Physical World.* 2nd ed. Princeton, NJ: Princeton University Press, 2014. When we talk about extremes of size, we invoke the mathematics of scaling. In appendix 3, Vogel gives a concise introduction to the essential mathematics and some of the strange things that happen when nature starts to compensate for physics.

Walter, W. Grey. “An Imitation of Life.” *Scientific American* 182, no. 5 (1950): 42–45. Walter talks about “electromechanical evolution.” We meet Elsie and Elmer, his “synthetic animals,” and Walter describes their design and behavior.

Werfel, Justin, Kirstin Peterson, and Nagpal Radhika. “Designing Collective Behavior in a Termite-Inspired Robot Construction Team.” *Science* 343 (2014): 745–758. This is an exciting project that puts the concept of stigmergy to work, showing that robots can leave behind a trail for one another using the structure that they are building—plus some simple traffic rules they enact that allow the group to coordinate its activities.

Wood, Gaby. *Edison's Eve: A Magical History of the Quest for Mechanical Life*. New York: Alfred A. Knopf, 2002. This history reminds us of the deep desire of humans to invent living machines. Wood brings to life the ancient inventors, such as Descartes and Vaucanson, who, working in clockwork, were laying the groundwork for modern robotics and trying to build living machines.

Wood, Robert, Radhika Nagpal, and Gu-Yeon Wei. "The Robobee Project Is Building Flying Robots the Size of Insects." *Scientific American* 303, no. 3 (2013). Wood and his colleagues explain the design and function of Robobees.

Xu, Yangsheng, Huihan Qian, and Xinyu Wu. *Household Service Robotics.* San Diego, CA: Elsevier, 2015. Chapter 3 introduces the process of building a map and then using the map for planning a path and avoiding objects.

Zaloga, Steven. *Unmanned Aerial Vehicles: Robotic Air Warfare 1917–2007*. Oxford, UK: Osprey Publishing, 2008. This is a brief history of drones with great pictures and illustrations. It's almost like a field guide to historic UAVs—concise and with explanatory visuals.