CHAPTER 1

Fully Actuated vs. Underactuated Systems

Robots today move far too conservatively, and accomplish only a fraction of the tasks and achieve a fraction of the performance that they are mechanically capable of. In many cases, we are still fundamentally limited by control technology which matured on rigid robotic arms in structured factory environments. The study of underactuated robotics focuses on building control systems which use the natural dynamics of the machines in an attempt to achieve extraordinary performance in terms of speed, efficiency, or robustness.

1.1 MOTIVATION

Let's start with some examples, and some videos.

1.1.1 Honda's ASIMO vs. Passive Dynamic Walkers

The world of robotics changed when, in late 1996, Honda Motor Co. announced that they had been working for nearly 15 years (behind closed doors) on walking robot technology. Their designs have continued to evolve over the last 12 years, resulting in a humanoid robot they call ASIMO (Advanced Step in Innovative MObility). Honda's ASIMO is widely considered to be the state of the art in walking robots, although there are now many robots with designs and performance very similar to ASIMO's. We will dedicate effort to understanding a few of the details of ASIMO in chapter 5... for now I just want you to become familiar with the look and feel of ASIMO's movements [watch asimo video now[1]].

I hope that your first reaction is to be incredibly impressed with the quality and versatility of ASIMO's movements. Now take a second look. Although the motions are very smooth, there is something a little unnatural about ASIMO's gait. It feels a little like an astronaut encumbered by a heavy space suit. In fact this is a reasonable analogy... ASIMO is walking like somebody that is unfamiliar with his/her dynamics. It's control system is using high-gain feedback, and therefore considerable joint torque, to cancel out the natural dynamics of the machine and strictly follow a desired trajectory. This control approach comes with a stiff penalty. ASIMO uses roughly 20 times the energy (scaled) that a human uses to walk on the flat (measured by cost of transport)[25]. Also, control stabilization in this approach only works in a relatively small portion of the state space (when the stance foot is flat on the ground), so ASIMO can't move nearly as quickly as a human, and cannot walk on unmodelled or uneven terrain.

For contrast, let's now consider a very different type of walking robot, called a passive dynamic walker. This "robot" has no motors, no controllers, no computer, but is still capable of walking stably down a small ramp, powered only by gravity. Most people will agree that the passive gait of this machine is more natural than ASIMO's; it is certainly

[1] `http://world.honda.com/ASIMO/`

more efficient. [watch PDW videos now[2]]. Passive walking machines have a long history - there are patents for passively walking toys dating back to the mid 1800's. We will discuss, in detail, what people know about the dynamics of these machines and what has been accomplished experimentally. This most impressive passive dynamic walker to date was built by Steve Collins in Andy Ruina's lab at Cornell.

Passive walkers demonstrate that the high-gain, dynamics-cancelling feedback approach taken on ASIMO is not a necessary one. In fact, the dynamics of walking is beautiful, and should be exploited - not cancelled out.

1.1.2 Birds vs. modern aircraft

The story is surprisingly similar in a very different type of machine. Modern airplanes are extremely effective for steady-level flight in still air. Propellers produce thrust very efficiently, and today's cambered airfoils are highly optimized for speed and/or efficiency. It would be easy to convince yourself that we have nothing left to learn from birds. But, like ASIMO, these machines are mostly confined to a very conservative, low angle-of-attack flight regime where the aerodynamics on the wing are well understood. Birds routinely execute maneuvers outside of this flight envelope (for instance, when they are landing on a perch), and are considerably more effective than our best aircraft at exploiting energy (eg, wind) in the air.

As a consequence, birds are extremely efficient flying machines; some are capable of migrating thousands of kilometers with incredibly small fuel supplies. The wandering albatross can fly for hours, or even days, without flapping its wings - these birds exploit the shear layer formed by the wind over the ocean surface in a technique called dynamic soaring. Remarkably, the metabolic cost of flying for these birds is indistinguishable from the baseline metabolic cost[6], suggesting that they can travel incredible distances (upwind or downwind) powered almost completely by gradients in the wind. Other birds achieve efficiency through similarly rich interactions with the air - including formation flying, thermal soaring, and ridge soaring. Small birds and large insects, such as butterflies and locusts, use 'gust soaring' to migrate hundreds or even thousands of kilometers carried primarily by the wind.

Birds are also incredibly maneuverable. The roll rate of a highly acrobatic aircraft (e.g, the A-4 Skyhawk) is approximately 720 deg/sec[73]; a barn swallow has a roll rate in excess of 5000 deg/sec[73]. Bats can be flying at full-speed in one direction, and completely reverse direction while maintaining forward speed, all in just over 2 wing-beats and in a distance less than half the wingspan[86]. Although quantitative flow visualization data from maneuvering flight is scarce, a dominant theory is that the ability of these animals to produce sudden, large forces for maneuverability can be attributed to unsteady aerodynamics, e.g., the animal creates a large suction vortex to rapidly change direction[87]. These astonishing capabilities are called upon routinely in maneuvers like flared perching, prey-catching, and high speed flying through forests and caves. Even at high speeds and high turn rates, these animals are capable of incredible agility - bats sometimes capture prey on their wings, Peregrine falcons can pull 25 G's out of a 240 mph dive to catch a sparrow in mid-flight[89], and even the small birds outside our building can be seen diving through a chain-link fence to grab a bite of food.

Although many impressive statistics about avian flight have been recorded, our un-

[2] `http://www-personal.engin.umich.edu/~shc/robots.html`

derstanding is partially limited by experimental accessibility - it's is quite difficult to carefully measure birds (and the surrounding airflow) during their most impressive maneuvers without disturbing them. The dynamics of a swimming fish are closely related, and can be more convenient to study. Dolphins have been known to swim gracefully through the waves alongside ships moving at 20 knots[87]. Smaller fish, such as the bluegill sunfish, are known to possess an escape response in which they propel themselves to full speed from rest in less than a body length; flow visualizations indeed confirm that this is accomplished by creating a large suction vortex along the side of the body[90] - similar to how bats change direction in less than a body length. There are even observations of a dead fish swimming upstream by pulling energy out of the wake of a cylinder; this passive propulsion is presumably part of the technique used by rainbow trout to swim upstream at mating season[10].

1.1.3 The common theme

Classical control techniques for robotics are based on the idea that feedback can be used to override the dynamics of our machines. These examples suggest that to achieve outstanding dynamic performance (efficiency, agility, and robustness) from our robots, we need to understand how to design control system which take advantage of the dynamics, not cancel them out. That is the topic of this course.

Surprisingly, there are relatively few formal control ideas that consider "exploiting" the dynamics. In order to convince a control theorist to consider the dynamics (efficiency arguments are not enough), you have to do something drastic, like taking away his control authority - remove a motor, or enforce a torque-limit. These issues have created a formal class of systems, the underactuated systems, for which people have begun to more carefully consider the dynamics of their machines in the context of control.

1.2 DEFINITIONS

According to Newton, the dynamics of mechanical systems are second order ($F = ma$). Their state is given by a vector of positions, $\mathbf{q}$, and a vector of velocities, $\dot{\mathbf{q}}$, and (possibly) time. The general form for a second-order controllable dynamical system is:

$$\ddot{\mathbf{q}} = \mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{u}, t),$$

where $\mathbf{u}$ is the control vector. As we will see, the forward dynamics for many of the robots that we care about turn out to be affine in commanded torque, so let's consider a slightly constrained form:

$$\ddot{\mathbf{q}} = \mathbf{f}_1(\mathbf{q}, \dot{\mathbf{q}}, t) + \mathbf{f}_2(\mathbf{q}, \dot{\mathbf{q}}, t)\mathbf{u}; . \tag{1.1}$$

DEFINITION 1 (Fully-Actuated)**.** A control system described by equation 1.1 is fully-actuated in configuration $(\mathbf{q}, \dot{\mathbf{q}}, t)$ if it is able to command an instantaneous acceleration in an arbitrary direction in $\mathbf{q}$:

$$\text{rank}\left[\mathbf{f}_2(\mathbf{q}, \dot{\mathbf{q}}, t)\right] = \dim\left[\mathbf{q}\right]. \tag{1.2}$$

DEFINITION 2 (Underactuated)**.** A control system described by equation 1.1 is underactuated in configuration $(\mathbf{q}, \dot{\mathbf{q}}, t)$ if it is not able to command an instantaneous acceleration in an arbitrary direction in $\mathbf{q}$:

$$\text{rank}\left[\mathbf{f}_2(\mathbf{q}, \dot{\mathbf{q}}, t)\right] < \dim\left[\mathbf{q}\right]. \tag{1.3}$$

Notice that whether or not a control system is underactuated may depend on the state of the system, although for most systems (including all of the systems in this book) underactuation is a global property of the system.

In words, underactuated control systems are those in which the control input cannot accelerate the state of the robot in arbitrary directions. As a consequence, unlike fully-actuated systems, underactuated system cannot be commanded to follow arbitrary trajectories.

EXAMPLE 1.1 Robot Manipulators

Consider the simple robot manipulator illustrated in Figure 1.1. As described in Appendix A, the equations of motion for this system are quite simple to derive, and take the form of the standard "manipulator equations":

$$\mathbf{H}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) = \mathbf{B}(\mathbf{q})\mathbf{u}.$$

It is well known that the inertial matrix, $\mathbf{H}(\mathbf{q})$ is (always) uniformly symmetric and positive definite, and is therefore invertible. Putting the system into the form of equation 1.1 yields:

$$\begin{aligned}\ddot{\mathbf{q}} =& \mathbf{H}^{-1}(\mathbf{q})\left[\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q})\right] \\ &+ \mathbf{H}^{-1}(\mathbf{q})\mathbf{B}(\mathbf{q})\mathbf{u}.\end{aligned}$$

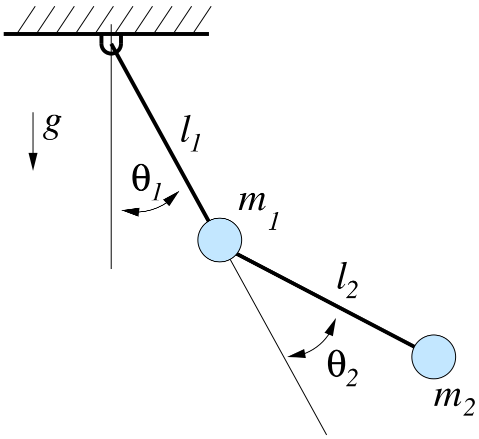

FIGURE 1.1 Simple double pendulum

Because $\mathbf{H}^{-1}(\mathbf{q})$ is always full rank, we find that a system described by the manipulator equations is fully-actuated if and only if $\mathbf{B}(\mathbf{q})$ is full row rank.

For this particular example, $\mathbf{q} = [\theta_1, \theta_2]^T$ and $\mathbf{u} = [\tau_1, \tau_2]^T$, and $\mathbf{B}(\mathbf{q}) = \mathbf{I}_{2\times2}$. The system is fully actuated. Now imagine the somewhat bizarre case that we have a motor to provide torque at the elbow, but no motor at the shoulder. In this case, we have $\mathbf{u} = \tau_2$, and $\mathbf{B}(\mathbf{q}) = [0, 1]^T$. This system is clearly underactuated. While it may sound like a contrived example, it turns out that it is exactly the dynamics we will use to study the compass gait model of walking in chapter 5.

The matrix $\mathbf{f}_2$ is equation 1.1 always has dim[$\mathbf{q}$] rows, and dim[$\mathbf{u}$] columns. Therefore, as in the example, one of the most common cases for underactuation, which trivially implies that $\mathbf{f}_2$ is not full row rank, is dim[$\mathbf{u}$] $<$ dim[$\mathbf{q}$]. But this is not the only case. The human body, for instance, has an incredible number of actuators (muscles), and in many cases has multiple muscles per joint; despite having more actuators that position variables, when I jump through the air, there is no combination of muscle inputs that can change the ballistic trajectory of my center of mass (barring aerodynamic effects). That control system is underactuated.

A quick note about notation. Throughout this class I will try to be consistent in using $\mathbf{q}$ and $\dot{\mathbf{q}}$ for positions and velocities, respectively, and reserve $\mathbf{x}$ for the full state ($\mathbf{x} = [\mathbf{q}, \dot{\mathbf{q}}]^T$). Unless otherwise noted, vectors are always treated as column vectors. Vectors and matrices are bold (scalars are not).

1.3 FEEDBACK LINEARIZATION

Fully actuated systems are dramatically easier to control than underactuated systems. The key observation is that, for fully-actuated systems with known dynamics (e.g., $\mathbf{f}_1$ and $\mathbf{f}_2$ are known), it is possible to use feedback to effectively change a nonlinear control problem into a linear control problem. The field of linear control is incredibly advanced, and there are many well-known solutions for controlling linear systems.

The trick is called feedback linearization. When $\mathbf{f}_2$ is full row rank, it is invertible. Consider the nonlinear feedback law:

$$\mathbf{u} = \pi(\mathbf{q}, \dot{\mathbf{q}}, t) = \mathbf{f}_2^{-1}(\mathbf{q}, \dot{\mathbf{q}}, t)\left[\mathbf{u}' - \mathbf{f}_1(\mathbf{q}, \dot{\mathbf{q}}, t)\right],$$

where $\mathbf{u}'$ is some additional control input. Applying this feedback controller to equation 1.1 results in the linear, decoupled, second-order system:

$$\ddot{\mathbf{q}} = \mathbf{u}'.$$

In other words, if $\mathbf{f}_1$ and $\mathbf{f}_2$ are known and $\mathbf{f}_2$ is invertible, then we say that the system is "feedback equivalent" to $\ddot{\mathbf{q}} = \mathbf{u}'$. There are a number of strong results which generalize this idea to the case where $\mathbf{f}_1$ and $\mathbf{f}_2$ are estimated, rather than known (e.g, [75]).

EXAMPLE 1.2 Feedback-Linearized Double Pendulum

Let's say that we would like our simple double pendulum to act like a simple single pendulum (with damping), whose dynamics are given by:

$$\begin{aligned}\ddot{\theta}_1 &= -\frac{g}{l}\cos\theta_1 - b\dot{\theta}_1 \\ \ddot{\theta}_2 &= 0.\end{aligned}$$

This is easily achieved[3] using

$$\mathbf{u} = \mathbf{B}^{-1}\left[\mathbf{C}\dot{\mathbf{q}} + \mathbf{G} + \mathbf{H}\begin{bmatrix} -\frac{g}{l}c_1 - b\dot{q}_1 \\ 0 \end{bmatrix}\right].$$

This idea can, and does, make control look easy - for the special case of a fully-actuated deterministic system with known dynamics. For example, it would have been just as easy for me to invert gravity. Observe that the control derivations here would not have been any more difficult if the robot had 100 joints.

[3] Note that our chosen dynamics do not actually stabilize θ_2 - this detail was left out for clarity, but would be necessary for any real implementation.

The underactuated systems are not feedback linearizable. Therefore, unlike fully-actuated systems, the control designer has no choice but to reason about the nonlinear dynamics of the plant in the control design. This dramatically complicates feedback controller design.

1.4 INPUT AND STATE CONSTRAINTS

Although dynamic constraints due to underactuation are the constraints which embody the spirit of this course, many of the systems we care about could be subject to other constraints, as well. For example, the actuators on our machines may only be mechanically capable of producing some limited amount of torque, or their may be a physical obstacle in the free space which we cannot permit our robot to come into contact with.

> **Definition 3** (Input and State Constraints)**.** A dynamical system described by $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}, t)$ may be subject to one or more constraints described by $\phi(\mathbf{x}, \mathbf{u}, t) \leq 0$.

In practice it can useful to separate out constraints which depend only on the input, e.g. $\phi(\mathbf{u}) \leq 0$, such as actuator limits, as they can often be easier to handle than state constraints. An obstacle in the environment might manifest itself as one or more constraints that depend only on position, e.g. $\phi(\mathbf{q}) \leq 0$.

Although distinct from underactuation constraints, these constraints can complicate control design in similar ways (i.e., the robot cannot follow arbitrary trajectories), and often require similar tools to find a control solution.

1.5 UNDERACTUATED ROBOTICS

The control of underactuated systems is an open and interesting problem in controls - although there are a number of special cases where underactuated systems have been controlled, there are relatively few general principles. Now here's the rub... most of the interesting problems in robotics are underactuated:

- Legged robots are underactuated. Consider a legged machine with N internal joints and N actuators. If the robot is not bolted to the ground, then the degrees of freedom of the system include both the internal joints and the six degrees of freedom which define the position and orientation of the robot in space. Since $\mathbf{u} \in \Re^N$ and $\mathbf{q} \in \Re^{N+6}$, equation 1.3 is satisfied.

- (Most) Swimming and flying robots are underactuated. The story is the same here as for legged machines. Each control surface adds one actuator and one DOF. And this is already a simplification, as the true state of the system should really include the (infinite-dimensional) state of the flow.

- Robot manipulation is (often) underactuated. Consider a fully-actuated robotic arm. When this arm is manipulating an object w/ degrees of freedom (even a brick has six), it can become underactuated. If force closure is achieved, and maintained, then we can think of the system as fully-actuated, because the degrees of freedom of the object are constrained to match the degrees of freedom of the hand. That is, of course, unless the manipulated object has extra DOFs. Note that the force-closure analogy has an interesting parallel in legged robots.

Even fully-actuated control systems can be improved using the lessons from underactuated systems, particularly if there is a need to increase the efficiency of their motions or reduce the complexity of their designs.

1.6 GOALS FOR THE COURSE

This course is based on the observation that there are new tools from computer science which be used to design feedback control for underactuated systems. This includes tools from numerical optimal control, motion planning, machine learning. The goal of this class is to develop these tools in order to design robots that are more dynamic and more agile than the current state-of-the-art.

The target audience for the class includes both computer science and mechanical/aero students pursuing research in robotics. Although I assume a comfort with linear algebra, ODEs, and MATLAB, the course notes will provide most of the material and references required for the course.