1 INTRODUCTION

In which we try to explain why we consider artificial intelligence to be a subject most worthy of study, and in which we try to decide what exactly it is, this being a good thing to decide before embarking.

We call ourselves *Homo sapiens—man* the wise—because our mental capacities are so important to us. For thousands of years, we have tried to understand *how we think;* that is, how a mere handful of stuff can perceive, understand, predict, and manipulate a world far larger and more complicated than itself. The field of artificial intelligence, or **AI**, goes further still: it attempts not just to understand but also to *build* intelligent entities.

ARTIFICIAL INTELLIGENCE

AI is one of the newest sciences. Work started in earnest soon after World War II, and the name itself was coined in 1956. Along with molecular biology, AI is regularly cited as the "field I would most like to be in" by scientists in other disciplines. A student in physics might reasonably feel that all the good ideas have already been taken by Galileo, Newton, Einstein, and the rest. AI, on the other hand, still has openings for several full-time Einsteins.

AI currently encompasses a huge variety of subfields, ranging from general-purpose areas, such as learning and perception to such specific tasks as playing chess, proving mathematical theorems, writing poetry, and diagnosing diseases. AI systematizes and automates intellectual tasks and is therefore potentially relevant to any sphere of human intellectual activity. In this sense, it is truly a universal field.

1.1 WHAT IS AI?

We have claimed that AI is exciting, but we have not said what it *is*. Definitions of artificial intelligence according to eight textbooks are shown in Figure 1.1. These definitions vary along two main dimensions. Roughly, the ones on top are concerned with *thought processes* and *reasoning,* whereas the ones on the bottom address *behavior*. The definitions on the left measure success in terms of fidelity to *human* performance, whereas the ones on the right measure against an *ideal* concept of intelligence, which we will call rationality. A system is rational if it does the "right thing," given what it knows.

RATIONALITY

Systems that think like humans	Systems that think rationally
"The exciting new effort to make computers think ... *machines with minds,* in the full and literal sense." (Haugeland, 1985)	"The study of mental faculties through the use of computational models." (Chamiak and McDermott, 1985)
"[The automation of] activities that we associate with human thinking, activities such as decision-making, problem solving, learning ..." (Bellman, 1978)	"The study of the computations that make it possible to perceive, reason, and act." (Winston, 1992)
Systems that act like humans	**Systems that act rationally**
"The art of creating machines that perform functions that require intelligence when performed by people." (Kurzweil, 1990)	"Computational Intelligence is the study of the design of intelligent agents." (Poole *et al.*, 1998)
"The study of how to make computers do things at which, at the moment, people are better." (Rich and Knight, 1991)	"AI ...is concerned with intelligent behavior in artifacts." (Nilsson, 1998)

Figure 1.1 Some definitions of artificial intelligence, organized into four categories.

Historically, all four approaches to AI have been followed. As one might expect, a tension exists between approaches centered around humans and approaches centered around rationality.[1] A human-centered approach must be an empirical science, involving hypothesis and experimental confirmation. A rationalist approach involves a combination of mathematics and engineering. Each group has both disparaged and helped the other. Let us look at the four approaches in more detail.

Acting humanly: The Turing Test approach

TURING TEST

The **Turing Test,** proposed by Alan Turing (1950), was designed to provide a satisfactory operational definition of intelligence. Rather than proposing a long and perhaps controversial list of qualifications required for intelligence, he suggested a test based on indistinguishability from undeniably intelligent entities —human beings. The computer passes the test if a human interrogator, after posing some written questions, cannot tell whether the written responses come from a person or not. Chapter 26 discusses the details of the test and whether a computer is really intelligent if it passes. For now, we note that programming a computer to pass the test provides plenty to work on. The computer would need to possess the following capabilities:

NATURAL LANGUAGE PROCESSING

◇ **natural language processing** to enable it to communicate successfully in English.

[1] We should point out that, by distinguishing between *human* and rational behavior, we are not suggesting that humans are necessarily "irrational" in the sense of "emotionally unstable" or "insane." One merely need note that we are not perfect: we are not all chess grandmasters, even those of us who know all the rules of chess; and, unfortunately, not everyone gets an A on the exam. Some systematic errors in human reasoning are cataloged by Kahneman *et al.* (1982).

KNOWLEDGE REPRESENTATION
◇ **knowledge representation** to store what it knows or hears;

AUTOMATED REASONING
◇ **automated reasoning** to use the stored information to answer questions and to draw new conclusions;

MACHINE LEARNING
◇ **machine learning** to adapt to new circumstances and to detect and extrapolate patterns.

Turing's test deliberately avoided direct physical interaction between the interrogator and the computer, because physical simulation of a person is unnecessary for intelligence. However,

TOTAL TURING TEST
the so-called **total Turing Test** includes a video signal so that the interrogator can test the subject's perceptual abilities, as well as the opportunity for the interrogator to pass physical objects "through the hatch." To pass the total Turing Test, the computer will need

COMPUTER VISION
◇ **computer vision** to perceive objects, and

ROBOTICS
◇ **robotics** to manipulate objects and move about.

These six disciplines compose most of AI, and Turing deserves credit for designing a test that remains relevant 50 years later. Yet AI researchers have devoted little effort to passing the Turing test, believing that it is more important to study the underlying principles of intelligence than to duplicate an exemplar. The quest for "artificial flight" succeeded when the Wright brothers and others stopped imitating birds and learned about aerodynamics. Aeronautical engineering texts do not define the goal of their field as making "machines that fly so exactly like pigeons that they can fool even other pigeons."

Thinking humanly: The cognitive modeling approach

If we are going to say that a given program thinks like a human, we must have some way of determining how humans think. We need to get *inside* the actual workings of human minds. There are two ways to do this: through introspection—trying to catch our own thoughts as they go by—and through psychological experiments. Once we have a sufficiently precise theory of the mind, it becomes possible to express the theory as a computer program. If the program's input/output and timing behaviors match corresponding human behaviors, that is evidence that some of the program's mechanisms could also be operating in humans. For example, Allen Newell and Herbert Simon, who developed GPS, the "General Problem Solver" (Newell and Simon, 1961), were not content to have their program solve problems correctly. They were more concerned with comparing the trace of its reasoning steps to traces of human

COGNITIVE SCIENCE
subjects solving the same problems. The interdisciplinary field of **cognitive science** brings together computer models from AI and experimental techniques from psychology to try to construct precise and testable theories of the workings of the human mind.

Cognitive science is a fascinating field, worthy of an encyclopedia in itself (Wilson and Keil, 1999). We will not attempt to describe what is known of human cognition in this book. We will occasionally comment on similarities or differences between AI techniques and human cognition. Real cognitive science, however, is necessarily based on experimental investigation of actual humans or animals, and we assume that the reader has access only to a computer for experimentation.

In the early days of AI there was often confusion between the approaches: an author would argue that an algorithm performs well on a task and that it is *therefore* a good model

of human performance, or vice versa. Modern authors separate the two kinds of claims; this distinction has allowed both AI and cognitive science to develop more rapidly. The two fields continue to fertilize each other, especially in the areas of vision and natural language. Vision in particular has recently made advances via an integrated approach that considers neurophysiological evidence and computational models.

Thinking rationally: The "laws of thought" approach

SYLLOGISMS

The Greek philosopher Aristotle was one of the first to attempt to codify "right thinking," that is, irrefutable reasoning processes. His **syllogisms** provided patterns for argument structures that always yielded correct conclusions when given correct premises—for example, "Socrates is a man; all men are mortal; therefore, Socrates is mortal." These laws of thought were supposed to govern the operation of the mind; their study initiated the field called **logic.**

LOGIC

Logicians in the 19th century developed a precise notation for statements about all kinds of things in the world and about the relations among them. (Contrast this with ordinary arithmetic notation, which provides mainly for equality and inequality statements about numbers.) By 1965, programs existed that could, in principle, solve *any* solvable problem described in logical notation.[2] The so-called **logicist** tradition within artificial intelligence hopes to build on such programs to create intelligent systems.

LOGICIST

There are two main obstacles to this approach. First, it is not easy to take informal knowledge and state it in the formal terms required by logical notation, particularly when the knowledge is less than 100% certain. Second, there is a big difference between being able to solve a problem "in principle" and doing so in practice. Even problems with just a few dozen facts can exhaust the computational resources of any computer unless it has some guidance as to which reasoning steps to try first. Although both of these obstacles apply to *any* attempt to build computational reasoning systems, they appeared first in the logicist tradition.

Acting rationally: The rational agent approach

AGENT

An **agent** is just something that acts (*agent* comes from the Latin *agere*, to do). But computer agents are expected to have other attributes that distinguish them from mere "programs," such as operating under autonomous control, perceiving their environment, persisting over a prolonged time period, adapting to change, and being capable of taking on another's goals. A

RATIONAL AGENT

rational agent is one that acts so as to achieve the best outcome or, when there is uncertainty, the best expected outcome.

In the "laws of thought" approach to AI, the emphasis was on correct inferences. Making correct inferences is sometimes *part* of being a rational agent, because one way to act rationally is to reason logically to the conclusion that a given action will achieve one's goals and then to act on that conclusion. On the other hand, correct inference is not all of rationality, because there are often situations where there is no provably correct thing to do, yet something must still be done. There are also ways of acting rationally that cannot be said to involve inference. For example, recoiling from a hot stove is a reflex action that is usually more successful than a slower action taken after careful deliberation.

[2] If there is no solution, the program might never stop looking for one.

All the skills needed for the Turing Test are there to allow rational actions. Thus, we need the ability to represent knowledge and reason with it because this enables us to reach good decisions in a wide variety of situations. We need to be able to generate comprehensible sentences in natural language because saying those sentences helps us get by in a complex society. We need learning not just for erudition, but because having a better idea of how the world works enables us to generate more effective strategies for dealing with it. We need visual perception not just because seeing is fun, but to get a better idea of what an action might achieve—for example, being able to see a tasty morsel helps one to move toward it.

For these reasons, the study of AI as rational-agent design has at least two advantages. First, it is more general than the "laws of thought" approach, because correct inference is just one of several possible mechanisms for achieving rationality. Second, it is more amenable to scientific development than are approaches based on human behavior or human thought because the standard of rationality is clearly defined and completely general. Human behavior, on the other hand, is well-adapted for one specific environment and is the product, in part, of a complicated and largely unknown evolutionary process that still is far from producing perfection. *This book will therefore concentrate on general principles of rational agents and on components for constructing them.* We will see that despite the apparent simplicity with which the problem can be stated, an enormous variety of issues come up when we try to solve it. Chapter 2 outlines some of these issues in more detail.

One important point to keep in mind: We will see before too long that achieving perfect rationality—always doing the right thing—is not feasible in complicated environments. The computational demands are just too high. For most of the book, however, we will adopt the working hypothesis that perfect rationality is a good starting point for analysis. It simplifies the problem and provides the appropriate setting for most of the foundational material in the field. Chapters 6 and 17 deal explicitly with the issue of limited rationality—acting appropriately when there is not enough time to do all the computations one might like.

LIMITED RATIONALITY

1.2 THE FOUNDATIONS OF ARTIFICIAL INTELLIGENCE

In this section, we provide a brief history of the disciplines that contributed ideas, viewpoints, and techniques to AI. Like any history, this one is forced to concentrate on a small number of people, events, and ideas and to ignore others that also were important. We organize the history around a series of questions. We certainly would not wish to give the impression that these questions are the only ones the disciplines address or that the disciplines have all been working toward AI as their ultimate fruition.

Philosophy (428 B.C.–present)

- Can formal rules be used to draw valid conclusions?
- How does the mental mind arise from a physical brain?
- Where does knowledge come from?
- How does knowledge lead to action?

Aristotle (384-322 B.C.) was the first to formulate a precise set of laws governing the rational part of the mind. He developed an informal system of syllogisms for proper reasoning, which in principle allowed one to generate conclusions mechanically, given initial premises. Much later, Ramon Lull (d. 1315) had the idea that useful reasoning could actually be carried out by a mechanical artifact. His "concept wheels" are on the cover of this book. Thomas Hobbes (1588–1679) proposed that reasoning was like numerical computation, that "we add and subtract in our silent thoughts." The automation of computation itself was already well under way; around 1500, Leonardo da Vinci (1452–1519) designed but did not build a mechanical calculator; recent reconstructions have shown the design to be functional. The first known calculating machine was constructed around 1623 by the German scientist Wilhelm Schickard (1592–1635), although the Pascaline, built in 1642 by Blaise Pascal (1623–1662), is more famous. Pascal wrote that "the arithmetical machine produces effects which appear nearer to thought than all the actions of animals." Gottfried Wilhelm Leibniz (1646–1716) built a mechanical device intended to carry out operations on concepts rather than numbers, but its scope was rather limited.

Now that we have the idea of a set of rules that can describe the formal, rational part of the mind, the next step is to consider the mind as a physical system. René Descartes (1596–1650) gave the first clear discussion of the distinction between mind and matter and of the problems that arise. One problem with a purely physical conception of the mind is that it seems to leave little room for free will: if the mind is governed entirely by physical laws, then it has no more free will than a rock "deciding" to fall toward the center of the earth. Although a strong advocate of the power of reasoning, Descartes was also a proponent of **dualism.** He held that there is a part of the human mind (or soul or spirit) that is outside of nature, exempt from physical laws. Animals, on the other hand, did not possess this dual quality; they could be treated as machines. An alternative to dualism is **materialism,** which holds that the brain's operation according to the laws of physics *constitutes* the mind. Free will is simply the way that the perception of available choices appears to the choice process.

DUALISM

MATERIALISM

Given a physical mind that manipulates knowledge, the next problem is to establish the source of knowledge. The **empiricism** movement, starting with Francis Bacon's (1561–1626) *Novum Organum*,[3] is characterized by a dictum of John Locke (1632–1704): "Nothing is in the understanding, which was not first in the senses." David Hume's (1711–1776) *A Treatise of Human Nature* (Hume, 1739) proposed what is now known as the principle of **induction:** that general rules are acquired by exposure to repeated associations between their elements. Building on the work of Ludwig Wittgenstein (1889–1951) and Bertrand Russell (1872–1970), the famous Vienna Circle, led by Rudolf Carnap (1891–1970), developed the doctrine of **logical positivism.** This doctrine holds that all knowledge can be characterized by logical theories connected, ultimately, to **observation sentences** that correspond to sensory inputs.[4] The **confirmation theory** of Carnap and Carl Hempel (1905–1997) attempted to understand how knowledge can be acquired from experience. Carnap's book *The Logical Structure of*

EMPIRICISM

INDUCTION

LOGICAL POSITIVISM

OBSERVATION SENTENCES

CONFIRMATION THEORY

[3] An update of Aristotle's *Organon*, or instrument of thought.

[4] In this picture, all meaningful statements can be verified or falsified either by analyzing the meaning of the words or by carrying out experiments. Because this rules out most of metaphysics, as was the intention, logical positivism was unpopular in some circles.

the World (1928) defined an explicit computational procedure for extracting knowledge from elementary experiences. It was probably the first theory of mind as a computational process.

The final element in the philosophical picture of the mind is the connection between knowledge and action. This question is vital to AI, because intelligence requires action as well as reasoning. Moreover, only by understanding how actions are justified can we understand how to build an agent whose actions are justifiable (or rational). Aristotle argued that actions are justified by a logical connection between goals and knowledge of the action's outcome (the last part of this extract also appears on the front cover of this book):

> But how does it happen that thinking is sometimes accompanied by action and sometimes not, sometimes by motion, and sometimes not? It looks as if almost the same thing happens as in the case of reasoning and making inferences about unchanging objects. But in that case the end is a speculative proposition . . . whereas here the conclusion which results from the two premises is an action. . . . I need covering; a cloak is a covering. I need a cloak. What I need, I have to make; I need a cloak. I have to make a cloak. And the conclusion, the "I have to make a cloak:' is an action. (Nussbaum, 1978, p. 40)

In the *Nicomachean Ethics* (Book III. *3,* 1112b), Aristotle further elaborates on this topic, suggesting an algorithm:

> We deliberate not about ends, but about means. For a doctor does not deliberate whether he shall heal, nor an orator whether he shall persuade, . . . They assume the end and consider how and by what means it is attained, and if it seems easily and best produced thereby; while if it is achieved by one means only they consider ***how*** it will be achieved by this and by what means ***this*** will be achieved, till they come to the first cause, . . . and what is last in the order of analysis seems to be first in the order of becoming. And if we come on an impossibility, we give up the search, e.g. if we need money and this cannot be got; but if a thing appears possible we try to do it.

Aristotle's algorithm was implemented 2300 years later by Newell and Simon in their GPS program. We would now call it a regression planning system. (See Chapter 11.)

Goal-based analysis is useful, but does not say what to do when several actions will achieve the goal, or when no action will achieve it completely. Antoine Arnauld (1612–1694) correctly described a quantitative formula for deciding what action to take in cases like this (see Chapter 16). John Stuart Mill's (1806–1873) book *Utilitarianism* (Mill, 1863) promoted the idea of rational decision criteria in all spheres of human activity. The more formal theory of decisions is discussed in the following section.

Mathematics (c. 800–present)

- What are the formal rules to draw valid conclusions?
- What can be computed?
- How do we reason with uncertain information?

Philosophers staked out most of the important ideas of AI, but the leap to a formal science required a level of mathematical formalization in three fundamental areas: logic, computation, and probability.

The idea of formal logic can be traced back to the philosophers of ancient Greece (see Chapter 7), but its mathematical development really began with the work of George Boole

(1815–1864), who worked out the details of propositional, or Boolean, logic (Boole, 1847). In 1879, Gottlob Frege (1848–1925) extended Boole's logic to include objects and relations, creating the first-order logic that is used today as the most basic knowledge representation system.[5] Alfred Tarski (1902–1983) introduced a theory of reference that shows how to relate the objects in a logic to objects in the real world. The next step was to determine the limits of what could be done with logic and computation.

ALGORITHM

The first nontrivial **algorithm** is thought to be Euclid's algorithm for computing greatest common denominators. The study of algorithms as objects in themselves goes back to al-Khowarazmi, a Persian mathematician of the 9th century, whose writings also introduced Arabic numerals and algebra to Europe. Boole and others discussed algorithms for logical deduction, and, by the late 19th century, efforts were under way to formalize general mathematical reasoning as logical deduction. In 1900, David Hilbert (1862–1943) presented a list of 23 problems that he correctly predicted would occupy mathematicians for the bulk of the century. The final problem asks whether there is an algorithm for deciding the truth of any logical proposition involving the natural numbers—the famous *Entscheidungsproblem*, or decision problem. Essentially, Hilbert was asking whether there were fundamental limits to the power of effective proof procedures. In 1930, Kurt Gödel (1906–1978) showed that there exists an effective procedure to prove any true statement in the first-order logic of Frege and Russell, but that first-order logic could not capture the principle of mathematical induction needed to characterize the natural numbers. In 1931, he showed that real limits do exist.

INCOMPLETENESS THEOREM

His **incompleteness theorem** showed that in any language expressive enough to describe the properties of the natural numbers, there are true statements that are undecidable in the sense that their truth cannot be established by any algorithm.

This fundamental result can also be interpreted as showing that there are some functions on the integers that cannot be represented by an algorithm—that is, they cannot be computed. This motivated Alan Turing (1912–1954) to try to characterize exactly which functions are capable of being computed. This notion is actually slightly problematic, because the notion of a computation or effective procedure really cannot be given a formal definition. However, the Church–Turing thesis, which states that the Turing machine (Turing, 1936) is capable of computing any computable function, is generally accepted as providing a sufficient definition. Turing also showed that there were some functions that no Turing machine can compute. For example, no machine can tell in general whether a given program will return an answer on a given input or run forever.

INTRACTABILITY

Although undecidability and noncomputability are important to an understanding of computation, the notion of **intractability** has had a much greater impact. Roughly speaking, a problem is called intractable if the time required to solve instances of the problem grows exponentially with the size of the instances. The distinction between polynomial and exponential growth in complexity was first emphasized in the mid-1960s (Cobham, 1964; Edmonds, 1965). It is important because exponential growth means that even moderately large instances cannot be solved in any reasonable time. Therefore, one should strive to divide

[5] Frege's proposed notation for first-order logic never became popular, for reasons that are apparent immediately from the example on the front cover.

the overall problem of generating intelligent behavior into tractable subproblems rather than intractable ones.

NP-COMPLETENESS

How can one recognize an intractable problem? The theory of **NP-completeness,** pioneered by Steven Cook (1971) and Richard Karp (1972), provides a method. Cook and Karp showed the existence of large classes of canonical combinatorial search and reasoning problems that are NP-complete. Any problem class to which the: class of NP-complete problems can be reduced is likely to be intractable. (Although it has not been proved that NP-complete problems are necessarily intractable, most theoreticians believe it.) These results contrast with the optimism with which the popular press greeted the first computers — "Electronic Super-Brains" that were "Faster than Einstein!" Despite the increasing speed of computers, careful use of resources will characterize intelligent systems. Put crudely, the world is an *extremely* large problem instance! In recent years, AI has helped explain why some instances of NP-complete problems are hard, yet others are easy (Cheeseman *et al.,* 1991).

PROBABILITY

Besides logic and computation, the third great contribution of mathematics to AI is the theory of **probability.** The Italian Gerolamo Cardano (1501–1576) first framed the idea of probability, describing it in terms of the possible outcomes of gambling events. Probability quickly became an invaluable part of all the quantitative sciences, helping to deal with uncertain measurements and incomplete theories. Pierre Fermat (1601–1665), Blaise Pascal (1623–1662), James Bernoulli (1654–1705), Pierre Laplace (1749–1827), and others advanced the theory and introduced new statistical methods. Thomas Bayes (1702–1761) proposed a rule for updating probabilities in the light of new evidence. Bayes' rule and the resulting field called Bayesian analysis form the basis of most modern approaches to uncertain reasoning in AI systems.

Economics (1776–present)

- How should we make decisions so as to maximize payoff?
- How should we do this when others may not go along?
- How should we do this when the payoff may be far in the future?

The science of economics got its start in 1776, when Scottish philosopher Adam Smith (1723–1790) published *An Inquiry into the Nature and Causes of the Wealth of Nations.* While the ancient Greeks and others had made contributions to economic thought, Smith was the first to treat it as a science, using the idea that economies can be thought of as consisting of individual agents maximizing their own economic well-being. Most people think of economics as being about money, but economists will say that they are really studying how people make choices that lead to preferred outcomes. The mathematical treatment of "preferred outcomes" or **utility** was first formalized by Léon Walras (pronounced "Valrasse") (1834-1910) and was improved by Frank Ramsey (1931) and later by John von Neumann and Oskar Morgenstern in their book *The Theory of Games and Economic Behavior* (1944).

DECISION THEORY

Decision theory, which combines probability theory with utility theory, provides a formal and complete framework for decisions (economic or otherwise) made under uncertainty—that is, in cases where probabilistic descriptions appropriately capture the decision-maker's environment. This is suitable for "large" economies where each agent need pay no attention

to the actions of other agents as individuals. For "small" economies, the situation is much more like a game: the actions of one player can significantly affect the utility of another (either positively or negatively). Von Neumann and Morgenstern's development of game theory (see also Luce and Raiffa, 1957) included the surprising result that, for some games, a rational agent should act in a random fashion, or at least in a way that appears random to the adversaries.

GAMETHEORY

For the most part, economists did not address the third question listed above, namely, how to make rational decisions when payoffs from actions are not immediate but instead result from several actions taken in *sequence*. This topic was pursued in the field of operations research, which emerged in World War II from efforts in Britain to optimize radar installations, and later found civilian applications in complex management decisions. The work of Richard Bellman (1957) formalized a class of sequential decision problems called Markov decision processes, which we study in Chapters 17 and 21.

OPERATIONS RESEARCH

Work in economics and operations research has contributed much to our notion of rational agents, yet for many years AI research developed along entirely separate paths. One reason was the apparent complexity of making rational decisions. Herbert Simon (1916–2001), the pioneering AI researcher, won the Nobel prize in economics in 1978 for his early work showing that models based on satisficing–making decisions that are "good enough," rather than laboriously calculating an optimal decision–gave a better description of actual human behavior (Simon, 1947). In the 1990s, there has been a resurgence of interest in decision-theoretic techniques for agent systems (Wellman, 1995).

SATISFICING

Neuroscience (1861–present)

- How do brains process information?

NEUROSCIENCE

Neuroscience is the study of the nervous system, particularly the brain. The exact way in which the brain enables thought is one of the great mysteries of science. It has been appreciated for thousands of years that the brain is somehow involved in thought, because of the evidence that strong blows to the head can lead to mental incapacitation. It has also long been known that human brains are somehow different; in about 335 B.C. Aristotle wrote, "Of all the animals, man has the largest brain in proportion to his size." [6] Still, it was not until the middle of the 18th century that the brain was widely recognized as the seat of consciousness. Before then, candidate locations included the heart, the spleen, and the pineal gland.

Paul Broca's (1824–1880) study of aphasia (speech deficit) in brain-damaged patients in 1861 reinvigorated the field and persuaded the medical establishment of the existence of localized areas of the brain responsible for specific cognitive functions. In particular, he showed that speech production was localized to a portion of the left hemisphere now called Broca's area.[7] By that time, it was known that the brain consisted of nerve cells or neurons, but it was not until 1873 that Camillo Golgi (1843–1926) developed a staining technique allowing the observation of individual neurons in the brain (see Figure 1.2). This technique

NEURONS

[6] Since then, it has been discovered that some species of dolphins and whales have relatively larger brains. The large size of human brains is now thought to be enabled in part by recent improvements in its cooling system.

[7] Many cite Alexander Hood (1824) as a possible prior source.

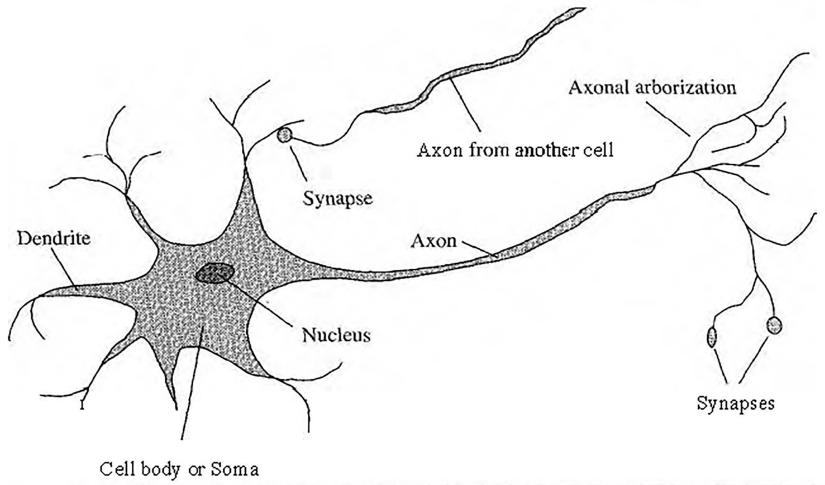


Figure 1.2 The parts of a nerve cell or neuron. Each neuron consists of a cell body, or soma, that contains a cell nucleus. Branching out from the cell body are a number of fibers called dendrites and a single long fiber called the axon. The axon stretches out for a long distance, much longer than the scale in this diagram indicates. Typically they are 1 cm long (100 times the diameter of the cell body), but can reach up to 1 meter. A neuron makes connections with 10 to 100,000 other neurons at junctions called synapses. Signals are propagated from neuron to neuron by a complicated electrochemical reaction. The signals control brain activity in the short term, and also enable long-term changes in the position and connectivity of neurons. These mechanisms are thought to form the basis for learning in the brain. Most information processing goes on in the cerebral cortex, the outer layer of the brain. The basic organizational unit appears to be a column of tissue about 0.5 mm in diameter, extending the full depth of the cortex, which is about 4 mm in humans. A column contains about 20,000 neurons.

was used by Santiago Ramon y Cajal (1852–1934) in his pioneering studies of the brain's neuronal structures.[8]

We now have some data on the mapping between areas of the brain and the parts of the body that they control or from which they receive sensory input. Such mappings are able to change radically over the course of a few weeks, and some animals seem to have multiple maps. Moreover, we do not fully understand how other areas can take over functions when one area is damaged. There is almost no theory on how an individual memory is stored.

The measurement of intact brain activity began in 1929 with the invention by Hans Berger of the electroencephalograph (EEG). The recent development of functional magnetic resonance imaging (fMRI) (Ogawa *et al.*, 1990) is giving neuroscientists unprecedentedly detailed images of brain activity, enabling measurements that correspond in interesting ways to ongoing cognitive processes. These are augmented by advances in single-cell recording of

[8] Golgi persisted in his belief that the brain's functions were carried out primarily in a continuous medium in which neurons were embedded, whereas Cajal propounded the "neuronal doctrine." The two shared the Nobel prize in 1906 but gave rather antagonistic acceptance speeches.

	Computer	Human Brain
Computational units	1 CPU, 10^8 gates	10^{11} neurons
Storage units	10^{10} bits RAM	10^{11} neurons
	10^{11} bits disk	10^{14} synapses
Cycle time	10^{-9} sec	10^{-3} sec
Bandwidth	10^{10} bits/sec	10^{14} bits/sec
Memory updates/sec	10^9	10^{14}

Figure 1.3 A crude comparison of the raw computational resources available to computers (**circa** 2003) and brains. The computer's numbers have all increased by at least a factor of 10 since the first edition of this book, and are expected to do so again this decade. The brain's numbers have not changed in the last 10,000 years.

neuron activity. Despite these advances, we are still a long way from understanding how any of these cognitive processes actually work.

The truly amazing conclusion is that *a collection of simple cells can lead to thought, action, and consciousness* or, in other words, that *brains cause minds* (Searle, 1992). The only real alternative theory is mysticism: that there is some mystical realm in which minds operate that is beyond physical science.

Brains and digital computers perform quite different tasks and have different properties. Figure 1.3 shows that there are 1000 times more neurons in the typical human brain than there are gates in the CPU of a typical high-end computer. Moore's Law[9] predicts that the CPU's gate count will equal the brain's neuron count around 2020. Of course, little can be inferred from such predictions; moreover, the difference in storage capacity is minor compared to the difference in switching speed and in parallelism. Computer chips can execute an instruction in a nanosecond, whereas neurons are millions of times slower. Brains more than make up for this, however, because all the neurons and synapses are active simultaneously, whereas most current computers have only one or at most a few CPUs. Thus, *even though a computer is a million times faster in raw switching speed, the brain ends up being 100,000 times faster at what it does.*

Psychology (1879–present)

- How do humans and animals think and act?

The origins of scientific psychology are usually traced to the work of the German physicist Hermann von Helmholtz (1821–1894) and his student Wilhelm Wundt (1832–1920). Helmholtz applied the scientific method to the study of human vision, and his *Handbook of Physiological Optics* is even now described as "the single most important treatise on the physics and physiology of human vision" (Nalwa, 1993, p.15). In 1879, Wundt opened the first laboratory of experimental psychology at the University of Leipzig. Wundt insisted on carefully controlled experiments in which his workers would perform a perceptual or associa-

[9] Moore's Law says that the number of transistors per square inch doubles every 1 to 1.5 years. Human brain capacity doubles roughly every 2 to 4 million years.

tive task while introspecting on their thought processes. The careful controls went a long way toward making psychology a science, but the subjective nature of the data made it unlikely that an experimenter would ever disconfirm his or her own theories. Biologists studying animal behavior, on the other hand, lacked introspective data and developed an objective methodology, as described by H. S. Jennings (1906) in his influential work *Behavior of the Lower Organisms*. Applying this viewpoint to humans, the **behaviorism** movement, led by John Watson (1878–1958), rejected *any* theory involving mental processes on the grounds that introspection could not provide reliable evidence. Behaviorists insisted on studying only objective measures of the percepts (or *stimulus*) given to an animal and its resulting actions (or *response)*. Mental constructs such as knowledge, beliefs, goals, and reasoning steps were dismissed as unscientific "folk psychology." Behaviorism discovered a lot about rats and pigeons, but had less success at understanding humans. Nevertheless, it exerted a strong hold on psychology (especially in the United States) from about 1920 to 1960.

BEHAVIORISM

The view of the brain as an information-processing device, which is a principal characteristic of **cognitive psychology**, can be traced back at least to the works of William James[10] (1842–1910). Helmholtz also insisted that perception involved a form of unconscious logical inference. The cognitive viewpoint was largely eclipsed by behaviorism in the United States, but at Cambridge's Applied Psychology Unit, directed by Frederic Bartlett (1886–1969), cognitive modeling was able to flourish. *The Nature of Explanation,* by Bartlett's student and successor Kenneth Craik (1943), forcefully reestablished the legitimacy of such "mental" terms as beliefs and goals, arguing that they are just as scientific as, say, using pressure and temperature to talk about gases, despite their being made of molecules that have neither. Craik specified the three key steps of a knowledge-based agent: (1) the stimulus must be translated into an internal representation, (2) the representation is manipulated by cognitive processes to derive new internal representations, and *(3)* these are in turn retranslated back into action. He clearly explained why this was a good design for an agent:

COGNITIVE PSYCHOLOGY

> If the organism carries a "small-scale model" of external reality and of its own possible actions within its head, it is able to try out various alternatives, conclude which is the best of them, react to future situations before they arise, utilize the knowledge of past events in dealing with the present and future, and in every way to react in a much fuller, safer, and more competent manner to the emergencies which face it. (Craik, 1943)

After Craik's death in a bicycle accident in 1945, his work was continued by Donald Broadbent, whose book *Perception and Communication* (1958) included some of the first information-processing models of psychological phenomena. Meanwhile, in the United States, the development of computer modeling led to the creation of the field of **cognitive science.** The field can be said to have started at a workshop in September 1956 at MIT. (We shall see that this is just two months after the conference at which AI itself was "born.") At the workshop, George Miller presented *The Magic Number Seven,* Noam Chomsky presented *Three Models of Language,* and Allen Newell and Herbert Simon presented *The Logic Theory Machine.* These three influential papers showed how coniputer models could be used to

COGNITIVE SCIENCE

[10] William James was the brother of novelist Henry James. It is said that Henry wrote fiction as if it were psychology and William wrote psychology as if it were fiction.

address the psychology of memory, language, and logical thinlung, respectively. It is now a common view among psychologists that "a cognitive theory should be like a computer program" (Anderson, 1980), that is, it should describe a detailed information-processing mechanism whereby some cognitive function might be implemented.

Computer engineering (1940–present)

- How can we build an efficient computer?

For artificial intelligence to succeed, we need two things: intelligence and an artifact. The computer has been the artifact of choice. The modern digital electronic computer was invented independently and almost simultaneously by scientists in three countries embattled in World War II. The first *operational* computer was the electromechanical Heath Robinson,[11] built in 1940 by Alan Turing's team for a single purpose: deciphering German messages. In 1943, the same group developed the Colossus, a powerful general-purpose machine based on vacuum tubes.[12] The first operational *programmable* computer was the Z-3, the invention of Konrad Zuse in Germany in 1941. Zuse also invented floating-point numbers and the first high-level programming language, Plankalkiil. The first *electronic* computer, the ABC, was assembled by John Atanasoff and his student Clifford Berry between 1940 and 1942 at Iowa State University. Atanasoff's research received little support or recognition; it was the ENIAC, developed as part of a secret military project at the University of Pennsylvania by a team including John Mauchly and John Eckert, that proved to be the most influential forerunner of modern computers.

In the half-century since then, each generation of computer hardware has brought an increase in speed and capacity and a decrease in price. Performance doubles every 18 months or so, with a decade or two to go at this rate of increase. After that, we will need molecular engineering or some other new technology.

Of course, there were calculating devices before the electronic computer. The earliest automated machines, dating from the 17th century, were discussed on page 6. The first *programmable* machine was a loom devised in 1805 by Joseph Marie Jacquard (1752–1834) that used punched cards to store instructions for the pattern to be woven. In the mid-19th century, Charles Babbage (1792–1871) designed two machines, neither of which he completed. The "Difference Engine," which appears on the cover of this book, was intended to compute mathematical tables for engineering and scientific projects. It was finally built and shown to work in 1991 at the Science Museum in London (Swade, 1993). Babbage's "Analytical Engine" was far more ambitious: it included addressable memory, stored programs, and conditional jumps and was the first artifact capable of universal computation. Babbage's colleague Ada Lovelace, daughter of the poet Lord Byron, was perhaps the world's first programmer. (The programming language Ada is named after her.) She wrote programs for the unfinished Analytical Engine and even speculated that the machine could play chess or compose music.

[11] Heath Robinson was a cartoonist famous for his depictions of whimsical and absurdly complicated contraptions for everyday tasks such as buttering toast.

[12] In the postwar period, Turing wanted to use these computers for AI research—for example, one of the first chess programs (Turing *et al.*, 1953). His efforts were blocked by the British government.

AI also owes a debt to the software side of computer science, which has supplied the operating systems, programming languages, and tools needed to write modern programs (and papers about them). But this is one area where the debt has been repaid: work in AI has pioneered many ideas that have made their way back to mainstream computer science, including time sharing, interactive interpreters, personal computers with windows and mice, rapid development environments, the linked list data type, automatic storage management, and key concepts of symbolic, functional, dynamic, and object-oriented programming.

Control theory and Cybernetics (1948–present)

- How can artifacts operate under their own control?

Ktesibios of Alexandria (c. 250 B.C.) built the first self-controlling machine: a water clock with a regulator that kept the flow of water running through it at a constant, predictable pace. This invention changed the definition of what an artifact could do. Previously, only living things could modify their behavior in response to changes in the environment. Other examples of self-regulating feedback control systems include the steam engine governor, created by James Watt (1736–1819), and the thermostat, invented by Cornelis Drebbel (1572–1633), who also invented the submarine. The mathematical theory of stable feedback systems was developed in the 19th century.

CONTROL THEORY

The central figure in the creation of what is now called **control theory** was Norbert Wiener (1894–1964). Wiener was a brilliant mathematician who worked with Bertrand Russell, among others, before developing an interest in biological and mechanical control systems and their connection to cognition. Like Craik (who also used control systems as psychological models), Wiener and his colleagues Arturo Rosenblueth and Julian Bigelow challenged the behaviorist orthodoxy (Rosenblueth *et al.,* 1943). They viewed purposive behavior as arising from a regulatory mechanism trying to minimize "error"—the difference between current state and goal state. In the late 1940s, Wiener, along with Warren McCulloch, Walter Pitts, and John von Neumann, organized a series of conferences that explored the new mathematical and computational models of cognition and influenced many other researchers in the behavioral sciences. Wiener's book Cybernetics (1948) became a bestseller and awoke the public to the possibility of artificially intelligent machines.

CYBERNETICS

Modern control theory, especially the branch known as stochastic optimal control, has as its goal the design of systems that maximize an **objective function** over time. This roughly matches our view of AI designing systems that behave optimally. Why, then, are AI and control theory two different fields, especially given the close connections among their founders? The answer lies in the close coupling between the mathematical techniques that were familiar to the participants and the corresponding sets of problems that were encompassed in each world view. Calculus and matrix algebra, the tools of control theory, lend themselves to systems that are describable by fixed sets of continuous variables; furthermore, exact analysis is typically feasible only for linear systems. AI was founded in part as a way to escape from the limitations of the mathematics of control theory in the 1950s. The tools of logical inference and computation allowed AI researchers to consider some problems such as language, vision, and planning, that fell completely outside the control theorist's purview.

OBJECTIVE FUNCTION

Linguistics (1957–present)

- How does language relate to thought?

In 1957, B. F. Skinner published *Verbal Behavior*. This was a comprehensive, detailed account of the behaviorist approach to language learning, written by the foremost expert in the field. But curiously, a review of the book became as well known as the book itself, and served to almost kill off interest in behaviorism. The author of the review was Noam Chomsky, who had just published a book on his own theory, *Syntactic Structures*. Chomsky showed how the behaviorist theory did not address the notion of creativity in language—it did not explain how a child could understand and make up sentences that he or she had never heard before. Chomsky's theory—based on syntactic models going back to the Indian linguist Panini (c. 350 B.C.)—could explain this, and unlike previous theories, it was formal enough that it could in principle be programmed.

COMPUTATIONAL LINGUISTICS

Modem linguistics and AI, then, were "born" at about the same time, and grew up together, intersecting in a hybrid field called **computational linguistics** or **natural language processing.** The problem of understanding language soon turned out to be considerably more complex than it seemed in 1957. Understanding language requires an understanding of the subject matter and context, not just an understanding of the structure of sentences. This might seem obvious, but it was not widely appreciated until the 1960s. Much of the early work in **knowledge representation** (the study of how to put knowledge into a form that a computer can reason with) was tied to language and informed by research in Linguistics, which was connected in turn to decades of work on the philosophical analysis of language.

1.3 The History of Artificial Intelligence

With the background material behind us, we are ready to cover the development of AI itself.

The gestation of artificial intelligence (1943–1955)

The first work that is now generally recognized as AI was done by Warren McCulloch and Walter Pitts (1943). They drew on three sources: knowledge of the basic physiology and function of neurons in the brain; a formal analysis of propositional logic due to Russell and Whitehead; and Turing's theory of computation. They proposed a model of artificial neurons in which each neuron is characterized as being "on" or "off," with a switch to "on" occurring in response to stimulation by a sufficient number of neighboring neurons. The state of a neuron was conceived of as "factually equivalent to a proposition which proposed its adequate stimulus." They showed, for example, that any computable function could be computed by some network of connected neurons, and that all the logical connectives (and, or, not, etc.) could be implemented by simple net structures. McCulloch and Pitts also suggested that suitably defined networks could learn. Donald Hebb (1949) demonstrated a simple updating rule for modifying the connection strengths between neurons. His rule, now called **Hebbian learning,** remains an influential model to this day.

Two undergraduate students at Harvard, Marvin Minsky and Dean Edmonds, built the first neural network computer in 1950. The SNARC, as it was called, used 3000 vacuum tubes and a surplus automatic pilot mechanism from a B-24 bomber to simulate a network of 40 neurons. Later, at Princeton, Minsky studied universal computation in neural networks. His Ph.D. committee was skeptical about whether this kind of work should be considered mathematics, but von Neumann reportedly said, "If it isn't now, it will be someday." Minsky was later to prove influential theorems showing the limitations of neural network research.

There were a number of early examples of work that can be characterized as AI, but it was Alan Turing who first articulated a complete vision of AI in his 1950 article "Computing Machinery and Intelligence." Therein, he introduced the Turing test, machine learning, genetic algorithms, and reinforcement learning.

The birth of artificial intelligence (1956)

Princeton was home to another influential figure in AI, John McCarthy. After graduation, McCarthy moved to Dartmouth College, which was to become the official birthplace of the field. McCarthy convinced Minsky, Claude Shannon, and Nathaniel Rochester to help him bring together U.S. researchers interested in automata theory, neural nets, and the study of intelligence. They organized a two-month workshop at Dartmouth in the summer of 1956. There were 10 attendees in all, including Trenchard More from Princeton, Arthur Samuel from IBM, and Ray Solomonoff and Oliver Selfridge from MIT.

Two researchers from Carnegie Tech,[13] Allen Newell and Herbert Simon, rather stole the show. Although the others had ideas and in some cases programs for particular applications such as checkers, Newell and Simon already had a reasoning program, the Logic Theorist (LT), about which Simon claimed, "We have invented a computer program capable of thinking non-numerically, and thereby solved the venerable mind–body problem."[14] Soon after the workshop, the program was able to prove most of the theorems in Chapter 2 of Russell and Whitehead's Principia *Mathematica*. Russell was reportedly delighted when Simon showed him that the program had come up with a proof for one theorem that was shorter than the one in Principia. The editors of the Journal of Symbolic Logic were less impressed; they rejected a paper coauthored by Newell, Simon, and Logic Theorist.

The Dartmouth workshop did not lead to any new breakthroughs, but it did introduce all the major figures to each other. For the next 20 years, the field would be dominated by these people and their students and colleagues at MIT, CMU, Stanford, and IBM. Perhaps the longest-lasting thing to come out of the workshop was an agreement to adopt McCarthy's new name for the field: **artificial intelligence.** Perhaps "computational rationality" would have been better, but "AI" has stuck.

Looking at the proposal for the Dartmouth workshop (McCarthy *et al.*, 1955), we can see why it was necessary for AI to become a separate field. Why couldn't all the work done

[13] Now Carnegie Mellon University (CMU).

[14] Newell and Simon also invented a list-processing language, IPL, to write LT. They had no compiler, and translated it into machine code by hand. To avoid errors, they worked in parallel, calling out binary numbers to each other as they wrote each instruction to make sure they agreed.

in AI have taken place under the name of control theory, or operations research, or decision theory, which, after all, have objectives similar to those of AI? Or why isn't AI a branch of mathematics? The first answer is that AI from the start embraced the idea of duplicating human faculties like creativity, self-improvement, and language use. None of the other fields were addressing these issues. The second answer is methodology. AI is the only one of these fields that is clearly a branch of computer science (although operations research does share an emphasis on computer simulations), and AI is the only field to attempt to build machines that will function autonomously in complex, changing environments.

Early enthusiasm, great expectations (1952–1969)

The early years of AI were full of successes—in a limited way. Given the primitive computers and programming tools of the time, and the fact that only a few years earlier computers were seen as things that could do *arithmetic* and no more, it was astonishing whenever a computer did anything remotely clever. The intellectual establishment, by and large, preferred to believe that "a machine can never do X." (See Chapter 26 for a long list of X's gathered by Turing.) AI researchers naturally responded by demonstrating one X after another. John McCarthy referred to this period as the "Look, Ma, no hands!" era.

Newell and Simon's early success was followed up with the General Problem Solver, or GPS. Unlike Logic Theorist, this program was designed from the start to imitate human problem-solving protocols. Within the limited class of puzzles it could handle, it turned out that the order in which the program considered subgoals and possible actions was similar to that in which humans approached the same problems. Thus, GPS was probably the first program to embody the "thinking humanly" approach. The success of GPS and subsequent programs as models of cognition led Newell and Simon (1976) to formulate the famous **physical symbol system** hypothesis, which states that "a physical symbol system has the necessary and sufficient means for general intelligent action." What they meant is that any system (human or machine) exhibiting intelligence must operate by manipulating data structures composed of symbols. We will see later that this hypothesis has been challenged from many directions.

PHYSICAL SYMBOL SYSTEM

At IBM, Nathaniel Rochester and his colleagues produced some of the first AI programs. Herbert Gelernter (1959) constructed the Geometry Theorem Prover, which was able to prove theorems that many students of mathematics would find quite tricky. Starting in 1952, Arthur Samuel wrote a series of programs for checkers (draughts) that eventually learned to play at a strong amateur level. Along the way, he disproved the idea that computers can do only what they are told to: his program quickly learned to play a better game than its creator. The program was demonstrated on television in February 1956, creating a very strong impression. Like Turing, Samuel had trouble finding computer time. Working at night, he used machines that were still on the testing floor at IBM's manufacturing plant. Chapter 6 covers game playing, and Chapter 21 describes and expands on the learning techniques used by Samuel.

John McCarthy moved from Dartmouth to MIT and there made three crucial contributions in one historic year: 1958. In MIT AI Lab Memo No. 1, McCarthy defined the high-level language **Lisp**, which was to become the dominant AI programming language. Lisp is the

second-oldest major high-level language in current use, one year younger than FORTRAN. With Lisp, McCarthy had the tool he needed, but access to scarce and expensive computing resources was also a serious problem. In response, he and others at MIT invented time sharing. Also in 1958, McCarthy published a paper entitled *Programs with Common Sense,* in which he described the Advice Taker, a hypothetical program that can be seen as the first complete AI system. Like the Logic Theorist and Geometry Theorem Prover, McCarthy's program was designed to use knowledge to search for solutions to problems. But unlike the others, it was to embody general knowledge of the world. For example, he showed how some simple axioms would enable the program to generate a plan to drive to the airport to catch a plane. The program was also designed so that it could accept new axioms in the normal course of operation, thereby allowing it to achieve competence in new areas *without being reprogrammed.* The Advice Taker thus embodied the central principles of knowledge representation and reasoning: that it is useful to have a formal, explicit representation of the world and of the way an agent's actions affect the world and to be able to manipulate these representations with deductive processes. It is remarkable how much of the 1958 paper remains relevant even today.

1958 also marked the year that Marvin Minsky moved to MIT. His initial collaboration with McCarthy did not last, however. McCarthy stressed representation and reasoning in formal logic, whereas Minsky was more interested in getting programs to work and eventually developed an anti-logical outlook. In 1963, McCarthy started the AI lab at Stanford. His plan to use logic to build the ultimate Advice Taker was advanced by J. A. Robinson's discovery of the resolution method (a complete theorem-proving algorithm for first-order logic; see Chapter 9). Work at Stanford emphasized general-purpose methods for logical reasoning. Applications of logic included Cordell Green's question-answering and planning systems (Green, 1969b) and the Shakey robotics project at the new Stanford Research Institute (SRI). The latter project, discussed further in Chapter 25, was the first to demonstrate the complete integration of logical reasoning and physical activity.

MICROWORLDS

Minsky supervised a series of students who chose limited problems that appeared to require intelligence to solve. These limited domains became known as **microworlds.** James Slagle's SAINT program (1963a) was able to solve closed-form calculus integration problems typical of first-year college courses. Tom Evans's ANALOGY program (1968) solved geometric analogy problems that appear in IQ tests, such as the one in Figure 1.4. Daniel Bobrow's STUDENT program (1967) solved algebra story problems, such as the following:

> If the number of customers Tom gets is twice the square of 20 percent of the number of advertisements he runs, and the number of advertisements he runs is 45, what is the number of customers Tom gets?

The most famous microvvorld was the blocks world, which consists of a set of solid blocks placed on a tabletop (or more often, a simulation of a tabletop), as shown in Figure 1.5. A typical task in this world is to rearrange the blocks in a certain way, using a robot hand that can pick up one block at a time. The blocks world was home to the vision project of David Huffman (1971), the vision and constraint-propagation work of David Waltz (1975), the learning theory of Patrick Winston (1970), the natural language understanding program

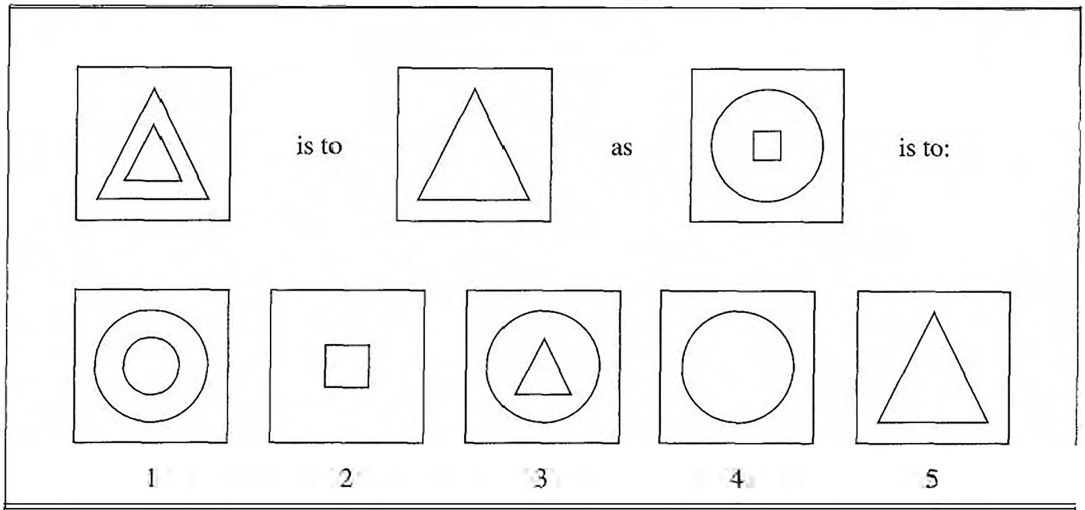


Figure 1.4 An example problem solved by Evans's ANALOGY program.

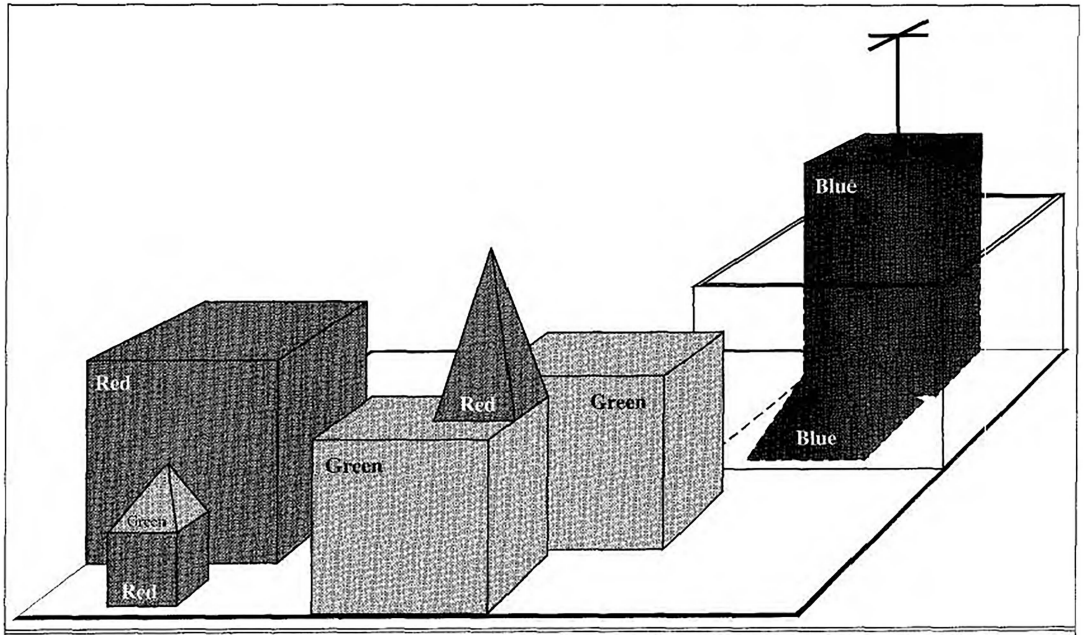


Figure 1.5 A scene from the blocks world. SHRDLU (Winograd, 1972) has just completed the command, "Find a block which is taller than the one you are holding and put it in the box."

of Terry Winograd (1972), and the planner of Scott Fahlman (1974).

Early work building on the neural networks of McCulloch and Pitts also flourished. The work of Winograd and Cowan (1963) showed how a large number of elements could collectively represent an individual concept, with a corresponding increase in robustness and parallelism. Hebb's learning methods were enhanced by Bernie Widrow (Widrow and Hoff,

1960; Widrow, 1962), who called his networks adalines, and by Frank Rosenblatt (1962) with his perceptrons. Rosenblatt proved the perceptron **convergence** theorem, showing that his learning algorithm could adjust the connection strengths of a perceptron to match any input data, provided such a match existed. These topics are covered in Chapter 20.

A dose of reality (1966–1973)

From the beginning, AI researchers were not shy about making predictions of their coming successes. The following statement by Herbert Simon in 1957 is often quoted:

> It is not my aim to surprise or shock you—but the simplest way I can summarize is to say that there are now in the world machines that think, that learn and that create. Moreover, their ability to do these things is going to increase rapidly until—in a visible future—the range of problems they can handle will be coextensive with the range to which the human mind has been applied.

Terms such as "visible future" can be interpreted in various ways, but Simon also made a more concrete prediction: that within 10 years a computer would be chess champion, and a significant mathematical theorem would be proved by machine. These predictions came true (or approximately true) within 40 years rather than 10. Simon's over-confidence was due to the promising performance of early AI systems on simple examples. In almost all cases. however, these early systems turned out to fail miserably when tried out on wider selections of problems and on more difficult problems.

The first kind of difficulty arose because most early programs contained little or no knowledge of their subject matter; they succeeded by means of simple syntactic manipulations. A typical story occurred in early machine translation efforts, which were generously funded by the U.S. National Research Council in an attempt to speed up the translation of Russian scientific papers in the wake of the Sputnik launch in 1957. It was thought initially that simple syntactic transformations based on the grammars of Russian and English, and word replacement using an electronic dictionary, would suffice to preserve the exact meanings of sentences. The fact is that translation requires general knowledge of the subject matter in order to resolve ambiguity and establish the content of the sentence. The famous re-translation of "the spirit is willing but the flesh is weak" as "the vodka is good but the meat is rotten" illustrates the difficulties encountered. In 1966, a report by an advisory committee found that "there has been no machine translation of general scientific text, and none is in immediate prospect." All U.S. government funding for academic translation projects was canceled. Today, machine translation is an imperfect but widely used tool for technical, commercial, government, and Internet documents.

The second kind of difficulty was the intractability of many of the problems that AI was attempting to solve. Most of the early AI programs solved problems by trying out different combinations of steps until the solution was found. This strategy worked initially because microworlds contained very few objects and hence very few possible actions and very short solution sequences. Before the theory of computational complexity was developed, it was widely thought that "scaling up" to larger problems was simply a matter of faster hardware and larger memories. The optimism that accompanied the development of resolution theorem

proving, for example, was soon dampened when researchers failed to prove theorems involving more than a few dozen facts. *The fact that a program can find a solution in principle does not mean that the program contains any of the mechanisms needed to find it in practice.*

MACHINE EVOLUTION

The illusion of unlimited computational power was not confined to problem-solving programs. Early experiments in machine evolution (now called genetic algorithms) (Friedberg, 1958; Friedberg *et al.*, 1959) were based on the undoubtedly correct belief that by making an appropriate series of small mutations to a machine code program, one can generate a program with good performance for any particular simple task. The idea, then, was to try random mutations with a selection process to preserve mutations that seemed useful. Despite thousands of hours of CPU time, almost no progress was demonstrated. Modern genetic algorithms use better representations and have shown more success.

Failure to come to grips with the "combinatorial explosion" was one of the main criticisms of AI contained in the Lighthill report (Lighthill, 1973), which formed the basis for the decision by the British government to end support for AI research in all but two universities. (Oral tradition paints a somewhat different and more colorful picture, with political ambitions and personal animosities whose description is beside the point.)

A third difficulty arose because of some fundamental limitations on the basic structures being used to generate intelligent behavior. For example, Minsky and Papert's book *Perceptrons* (1969) proved that, although perceptrons (a simple form of neural network) could be shown to learn anything they were capable of representing, they could represent very little. In particular, a two-input perceptron could not be trained to recognize when its two inputs were different. Although their results did not apply to more complex, multilayer networks, research funding for neural-net research soon dwindled to almost nothing. Ironically, the new back-propagation learning algorithms for multilayer networks that were to cause an enormous resurgence in neural-net research in the late 1980s were actually discovered first in 1969 (Bryson and Ho, 1969).

Knowledge-based systems: The key to power? (1969–1979)

The picture of problem solving that had arisen during the first decade of AI research was of a general-purpose search mechanism trying to string together elementary reasoning steps to find complete solutions. Such approaches have been called weak methods, because, although general, they do not scale up to large or difficult problem instances. The alternative to weak methods is to use more powerful, domain-specific knowledge that allows larger reasoning steps and can more easily handle typically occurring cases in narrow areas of expertise. One might say that to solve a hard problem, you have to almost know the answer already.

WEAK METHODS

The DENDRAL program (Buchanan *et al.*, 1969) was an early example of this approach. It was developed at Stanford, where Ed Feigenbaum (a former student of Herbert Simon), Bruce Buchanan (a philosopher turned computer scientist), and Joshua Lederberg (a Nobel laureate geneticist) teamed up to solve the problem of inferring molecular structure from the information provided by a mass spectrometer. The input to the program consists of the elementary formula of the molecule (e.g., $C_6H_{13}NO_2$) and the mass spectrum giving the masses of the various fragments of the molecule generated when it is bombarded by an electron beam.

For example, the mass spectrum might contain a peak at m = 15, corresponding to the mass of a methyl (CH_3) fragment.

The naive version of the program generated all possible structures consistent with the formula, and then predicted what mass spectrum would be observed for each, comparing this with the actual spectrum. As one might expect, this is intractable for decent-sized molecules. The DENDRAL researchers consulted analytical chemists and found that they worked by looking for well-known patterns of peaks in the spectrum that suggested common substructures in the molecule. For example, the following rule is used to recognize a ketone (C=O) subgroup (which weighs 28):

> if there are two peaks at x_1 and x_2 such that
> (a) $x_1 + x_2 = M + 28$ (M is the mass of the whole molecule);
> (b) $x_1 - 28$ is a high peak;
> (c) $x_2 - 28$ is a high peak;
> (d) At least one of x_1 and x_2 is high.
> *then* there is a ketone subgroup

Recognizing that the molecule contains a particular substructure reduces the number of possible candidates enormously. DENDRAL was powerful because

> All the relevant theoretical knowledge to solve these problems has been mapped over from its general form in the [spectrum prediction component] ("first principles") to efficient special forms ("cookbook recipes"). (Feigenbaum *et* al., 1971)

The significance of DENDRAL was that it was the first successful *knowledge-intensive* system: its expertise derived from large numbers of special-purpose rules. Later systems also incorporated the main theme of McCarthy's Advice Taker approach—the clean separation of the knowledge (in the form of rules) from the reasoning component.

With this lesson in mind, Feigenbaum and others at Stanford began the Heuristic Programming Project (HPP), to investigate the extent to which the new methodology of **expert systems** could be applied to other areas of human expertise. The next major effort was in the area of medical diagnosis. Feigenbaum, Buchanan, and Dr. Edward Shortliffe developed MYCIN to diagnose blood infections. With about 450 rules, MYCIN was able to perform as well as some experts, and considerably better than junior doctors. It also contained two major differences from DENDRAL. First, unlike the DENDRAL rules, no general theoretical model existed from which the MYCIN rules could be deduced. They had to be acquired from extensive interviewing of experts, who in turn acquired them from textbooks, other experts, and direct experience of cases. Second, the rules had to reflect the uncertainty associated with medical knowledge. MYCIN incorporated a calculus of uncertainty called **certainty factors** (see Chapter 14), which seemed (at the time) to fit well with how doctors assessed the impact of evidence on the diagnosis.

EXPERTSYSTEMS

The importance of domain knowledge was also apparent in the area of understanding natural language. Although Winograd's SHRDLU system for understanding natural language had engendered a good deal of excitement, its dependence on syntactic analysis caused some of the same problems as occurred in the early machine translation work. It was able to overcome ambiguity and understand pronoun references, but this was mainly because it was

designed specifically for one area—the blocks world. Several researchers, including Eugene Charniak, a fellow graduate student of Winograd's at MIT, suggested that robust language understanding would require general knowledge about the world and a general method for using that knowledge.

At Yale, the linguist-turned-AI-researcher Roger Schank emphasized this point, claiming, "There is no such thing as syntax," which upset a lot of linguists, but did serve to start a useful discussion. Schank and his students built a series of programs (Schank and Abelson, 1977; Wilensky, 1978; Schank and Riesbeck, 1981; Dyer, 1983) that all had the task of understanding natural language. The emphasis, however, was less on language per *se* and more on the problems of representing and reasoning with the knowledge required for language understanding. The problems included representing stereotypical situations (Cullingford, 1981), describing human memory organization (Rieger, 1976; Kolodner, 1983), and understanding plans and goals (Wilensky, 1983).

The widespread growth of applications to real-world problems caused a concurrent increase in the demands for workable knowledge representation schemes. A large number of different representation and reasoning languages were developed. Some were based on logic—for example, the Prolog language became popular in Europe, and the PLANNER family in the United States. Others, following Minsky's idea of **frames** (1975), adopted a more structured approach, assembling facts about particular object and event types and arranging the types into a large taxonomic hierarchy analogous to a biological taxonomy.

FRAMES

AI becomes an industry (1980–present)

The first successful commercial expert system, R1, began operation at the Digital Equipment Corporation (McDermott, 1982). The program helped configure orders for new computer systems; by 1986, it was saving the company an estimated $40 million a year. By 1988, DEC's AI group had 40 expert systems deployed, with more on the way. Du Pont had 100 in use and 500 in development, saving an estimated $10 million a year. Nearly every major U.S. corporation had its own AI group and was either using or investigating expert systems.

In 1981, the Japanese announced the "Fifth Generation" project, a 10-year plan to build intelligent computers running Prolog. In response the United States formed the Microelectronics and Computer Technology Corporation (MCC) as a research consortium designed to assure national competitiveness. In both cases, AI was part of a broad effort, including chip design and human-interface research. However, the AI components of MCC and the Fifth Generation projects never met their ambitious goals. In Britain, the Alvey report reinstated the funding that was cut by the Lighthill report.[15]

Overall, the AI industry boomed from a few million dollars in 1980 to billions of dollars in 1988. Soon after that came a period called the "AI Winter," in which many companies suffered as they failed to deliver on extravagant promises.

[15] To save embarrassment, a new field called IKBS (Intelligent Knowledge-Based Systems) was invented because Artificial Intelligence had been officially canceled.

The return of neural networks (1986–present)

Although computer science had largely abandoned the field of neural networks in the late 1970s, work continued in other fields. Physicists such as John Hopfield (1982) used techniques from statistical mechanics to analyze the storage and optimization properties of networks, treating collections of nodes like collections of atoms. Psychologists including David Rumelhart and Geoff Hinton continued the study of neural-net models of memory. As we discuss in Chapter 20, the real impetus came in the mid-1980s when at least four different groups reinvented the back-propagation learning algorithm first found in 1969 by Bryson and Ho. The algorithm was applied to many learning problems in computer science and psychology, and the widespread dissemination of the results in the collection *Parallel Distributed Processing* (Rumelhart and McClelland, 1986) caused great excitement.

CONNECTIONIST

These so-called **connectionist** models of intelligent systems were seen by some as direct competitors both to the symbolic models promoted by Newell and Simon and to the logicist approach of McCarthy and others (Smolensky, 1988). It might seem obvious that at some level humans manipulate symbols—in fact, Terrence Deacon's book *The Symbolic Species* (1997) suggests that this is the *defining characteristic* of humans, but the most ardent connectionists questioned whether symbol manipulation had any real explanatory role in detailed models of cognition. This question remains unanswered, but the current view is that connectionist and symbolic approaches are complementary, not competing.

AI becomes a science (1987–present)

Recent years have seen a revolution in both the content and the methodology of work in artificial intelligence.[16] It is now more common to build on existing theories than to propose brand new ones, to base claims on rigorous theorems or hard experimental evidence rather than on intuition, and to show relevance to real-world applications rather than toy examples.

AI was founded in part as a rebellion against the limitations of existing fields like control theory and statistics, but now it is embracing those fields. As David McAllester (1998) put it,

> In the early period of AI it seemed plausible that new forms of symbolic computation, e.g., frames and semantic networks, made much of classical theory obsolete. This led to a form of isolationism in which AI became largely separated from the rest of computer science. This isolationism is currently being abandoned. There is a recognition that machine learning should not be isolated from information theory, that uncertain reasoning should not be isolated from stochastic modeling, that search should not be isolated from classical optimization and control, and that automated reasoning should not be isolated from formal methods and static analysis.

In terms of methodology, AI has finally come firmly under the scientific method. To be accepted, hypotheses must be subjected to rigorous empirical experiments, and the results must

[16] Some have characterized this change as a victory of the **neats**—those who think that AI theories should be grounded in mathematical rigor—over the **scruffies**—those who would rather try out lots of ideas, write some programs, and then assess what seems to be working. Both approaches are important. A shift toward neatness implies that the field has reached a level of stability and maturity. Whether that stability will be disrupted by a new scruffy idea is another question.

be analyzed statistically for their importance (Cohen, 1995). Through the use of the Internet and shared repositories of test data and code, it is now possible to replicate experiments.

The field of speech recognition illustrates the pattern. In the 1970s, a wide variety of different architectures and approaches were tried. Many of these were rather *ad hoc* and fragile, and were demonstrated on only a few specially selected examples. In recent years, approaches based on **hidden Markov models** (HMMs) have come to dominate the area. Two aspects of HMMs are relevant. First, they are based on a rigorous mathematical theory. This has allowed speech researchers to build on several decades of mathematical results developed in other fields. Second, they are generated by a process of training on a large corpus of real speech data. This ensures that the performance is robust, and in rigorous blind tests the HMMs have been improving their scores steadily. Speech technology and the related field of handwritten character recognition are already making the transition to widespread industrial and consumer applications.

Neural networks also fit this trend. Much of the work on neural nets in the 1980s was done in an attempt to scope out what could be done and to learn how neural nets differ from "traditional" techniques. Using improved methodology and theoretical frameworks, the field arrived at an understanding in which neural nets can now be compared with corresponding techniques from statistics, pattern recognition, and machine learning, and the most promising technique can be applied to each application. As a result of these developments, so-called **data mining** technology has spawned a vigorous new industry.

DATA MINING

Judea Pearl's (1988) *Probabilistic Reasoning in Intelligent Systems* led to a new acceptance of probability and decision theory in AI, following a resurgence of interest epitomized by Peter Cheeseman's (1985) article "In Defense of Probability." The **Bayesian network** formalism was invented to allow efficient representation of, and rigorous reasoning with, uncertain knowledge. This approach largely overcomes many problems of the probabilistic reasoning systems of the 1960s and 1970s; it now dominates AI research on uncertain reasoning and expert systems. The approach allows for learning from experience, and it combines the best of classical AI and neural nets. Work by Judea Pearl (1982a) and by Eric Horvitz and David Heckerman (Horvitz and Heckerman, 1986; Horvitz et *al.*, 1986) promoted the idea of *normative* expert systems: ones that act rationally according to the laws of decision theory and do not try to imitate the thought steps of human experts. The windows^{TM} operating system includes several normative diagnostic expert systems for correcting problems. Chapters 13 to 16 cover this area.

Similar gentle revolutions have occurred in robotics, computer vision, and knowledge representation. A better understanding of the problems and their complexity properties, combined with increased mathematical sophistication, has led to workable research agendas and robust methods. In many cases, formalization and specialization have also led to fragmentation: topics such as vision and robotics are increasingly isolated from "mainstream" AI work. The unifying view of AI as rational agent design is one that can bring unity back to these disparate fields.

The emergence of intelligent agents (1995–present)

Perhaps encouraged by the progress in solving the subproblems of AI, researchers have also started to look at the "whole agent" problem again. The work of Allen Newell, John Laird, and Paul Rosenbloom on SOAR (Newell, 1990; Laird *et* al., 1987) is the best-known example of a complete agent architecture. The so-called situated movement aims to understand the workings of agents embedded in real environments with continuous sensory inputs. One of the most important environments for intelligent agents is the Internet. AI systems have become so common in web-based applications that the "-bot" suffix has entered everyday language. Moreover, AI technologies underlie many Internet tools, such as search engines, recommender systems, and Web site construction systems.

Besides the first edition of this text (Russell and Norvig, 1995), other recent texts have also adopted the agent perspective (Poole *et* al., 1998; Nilsson, 1998). One consequence of trying to build complete agents is the realization that the previously isolated subfields of AI might need to be reorganized somewhat when their results are to be tied together. In particular, it is now widely appreciated that sensory systems (vision, sonar, speech recognition, etc.) cannot deliver perfectly reliable information about the environment. Hence, reasoning and planning systems must be able to handle uncertainty. A second major consequence of the agent perspective is that AI has been drawn into much closer contact with other fields, such as control theory and economics, that also deal with agents.

1.4 THE STATE OF THE ART

What can AI do today? A concise answer is difficult, because there are so many activities in so many subfields. Here we sample a few applications; others appear throughout the book.

Autonomous planning and scheduling: A hundred million miles from Earth, NASA's Remote Agent program became the first on-board autonomous planning program to control the scheduling of operations for a spacecraft (Jonsson *et* al., 2000). Remote Agent generated plans from high-level goals specified from the ground, and it monitored the operation of the spacecraft as the plans were executed—detecting, diagnosing, and recovering from problems as they occurred.

Game playing: IBM's Deep Blue became the first computer program to defeat the world champion in a chess match when it bested Garry Kasparov by a score of 3.5 to 2.5 in an exhibition match (Goodman and Keene, 1997). Kasparov said that he felt a "new kind of intelligence" across the board from him. *Newsweek* magazine described the match as "The brain's last stand." The value of IBM's stock increased by $18 billion.

Autonomous control: The ALVINN computer vision system was trained to steer a car to keep it following a lane. St was placed in CMU's NAVLAB computer-controlled minivan and used to navigate across the United States—for 2850 miles it was in control of steering the vehicle 98% of the time. A human took over the other 2%, mostly at exit ramps. NAVLAB has video cameras that transmit road images to ALVINN, which then computes the best direction to steer, based on experience from previous training runs.

Diagnosis: Medical diagnosis programs based on probabilistic analysis have been able to perform at the level of an expert physician in several areas of medicine. Heckerman (1991) describes a case where a leading expert on lymph-node pathology scoffs at a program's diagnosis of an especially difficult case. The creators of the program suggest he ask the computer for an explanation of the diagnosis. The machine points out the major factors influencing its decision and explains the subtle interaction of several of the symptoms in this case. Eventually, the expert agrees with the program.

Logistics Planning: During the Persian Gulf crisis of 1991, U.S. forces deployed a Dynamic Analysis and Replanning Tool, DART (Cross and Walker, 1994), to do automated logistics planning and scheduling for transportation. This involved up to 50,000 vehicles, cargo, and people at a time, and had to account for starting points, destinations, routes, and conflict resolution among all parameters. The AI planning techniques allowed a plan to be generated in hours that would have taken weeks with older methods. The Defense Advanced Research Project Agency (DARPA) stated that this single application more than paid back DARPA's 30-year investment in AI.

Robotics: Many surgeons now use robot assistants in microsurgery. HipNav (DiGioia *et* al., 1996) is a system that uses computer vision techniques to create a three-dimensional model of a patient's internal anatomy and then uses robotic control to guide the insertion of a hip replacement prosthesis.

Language understanding and problem solving: PROVERB (Littman *et al.*, 1999) is a computer program that solves crossword puzzles better than most humans, using constraints on possible word fillers, a large database of past puzzles, and a variety of information sources including dictionaries and online databases such as a list of movies and the actors that appear in them. For example, it determines that the clue "Nice Story" can be solved by "ETAGE because its database includes the clue/solution pair "Story in France/ETAGE" and because it recognizes that the patterns "Nice *X*" and "X in France" often have the same solution. The program does not know that Nice is a city in France, but it can solve the puzzle.

These are just a few examples of artificial intelligence systems that exist today. Not magic or science fiction—but rather science, engineering, and mathematics, to which this book provides an introduction.

1.5 SUMMARY

This chapter defines AI and establishes the cultural background against which it has developed. Some of the important points are as follows:

- Different people think of AI differently. Two important questions to ask are: Are you concerned with thinking or behavior? Do you want to model humans or work from an ideal standard?
- In this book, we adopt the view that intelligence is concerned mainly with **rational action.** Ideally, an **intelligent agent** takes the best possible action in a situation. We will study the problem of building agents that are intelligent in this sense.

- Philosophers (going back to 400 B.C.) made AI conceivable by considering the ideas that the mind is in some ways like a machine, that it operates on knowledge encoded in some internal language, and that thought can be used to choose what actions to take.
- Mathematicians provided the tools to manipulate statements of logical certainty as well as uncertain, probabilistic statements. They also set the groundwork for understanding computation and reasoning about algorithms.
- Economists formalized the problem of making decisions that maximize the expected outcome to the decision-maker.
- Psychologists adopted the idea that humans and animals can be considered information-processing machines. Linguists showed that language use fits into this model.
- Computer engineers provided the artifacts that make AI applications possible. AI programs tend to be large, and they could not work without the great advances in speed and memory that the computer industry has provided.
- Control theory deals with designing devices that act optimally on the basis of feedback from the environment. Initially, the mathematical tools of control theory were quite different from AI, but the fields are coming closer together.
- The history of AI has had cycles of success, misplaced optimism, and resulting cutbacks in enthusiasm and funding. There have also been cycles of introducing new creative approaches and systematically refining the best ones.
- AI has advanced more rapidly in the past decade because of greater use of the scientific method in experimenting with and comparing approaches.
- Recent progress in understanding the theoretical basis for intelligence has gone hand in hand with improvements in the capabilities of real systems. The subfields of AI have become more integrated, and AI has found common ground with other disciplines.

BIBLIOGRAPHICAL AND HISTORICAL NOTES

The methodological status of artificial intelligence is investigated in *The Sciences of the Artificial*, by Herb Simon (1981), which discusses research areas concerned with complex artifacts. It explains how AI can be viewed as both science and mathematics. Cohen (1995) gives an overview of experimental methodology within AI. Ford and Hayes (1995) give an opinionated view of the usefulness of the Turing Test.

Artificial Intelligence: The Very Idea, by John Haugeland (1985) gives a readable account of the philosophical and practical problems of AI. Cognitive science is well described by several recent texts (Johnson-Laird, 1988; Stillings *et al.,* 1995; Thagard, 1996) and by the *Encyclopedia of the Cognitive Sciences* (Wilson and Keil, 1999). Baker (1989) covers the syntactic part of modern linguistics, and Chierchia and McConnell-Ginet (1990) cover semantics. Jurafsky and Martin (2000) cover computational linguistics.

Early AI is described in Feigenbaum and Feldman's *Computers and Thought* (1963), Minsky's *Semantic Information Processing* (1968), and the *Machine Intelligence* series edited by Donald Michie. A large number of influential papers have been anthologized by Webber

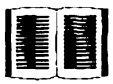
and Nilsson (1981) and by Luger (1995). Early papers on neural networks are collected in *Neurocomputing* (Anderson and Rosenfeld, 1988). The *Encyclopedia of AI* (Shapiro, 1992) contains survey articles on almost every topic in AI. These articles usually provide a good entry point into the research literature on each topic.

The most recent work appears in the proceedings of the major AI conferences: the biennial International Joint Conference on AI (IJCAI), the annual European Conference on AI (ECAI), and the National Conference on AI, more often known as AAAI, after its sponsoring organization. The major journals for general AI are *Artificial Intelligence, Computational Intelligence,* the *IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE Intelligent Systems,* and the electronic *Journal of Artificial Intelligence Research.* There are also many conferences and journals devoted to specific areas, which we cover in the appropriate chapters. The main professional societies for AI are the American Association for Artificial Intelligence (AAAI), the ACM Special Interest Group in Artificial Intelligence (SIGART), and the Society for Artificial Intelligence and Simulation of Behaviour (AISB). AAAI's *AI Magazine* contains many topical and tutorial articles, and its website, aaai.org, contains news and background information.

EXERCISES

These exercises are intended to stimulate discussion, and some might be set as term projects. Alternatively, preliminary attempts can be made now, and these attempts can be reviewed after the completion of the book.

1.1 Define in your own words: (a) intelligence, (b) artificial intelligence, (c) agent.



1.2 Read Turing's original paper on AI (Turing, 1950). In the paper, he discusses several potential objections to his proposed enterprise and his test for intelligence. Which objections still carry some weight? Are his refutations valid? Can you think of new objections arising from developments since he wrote the paper? In the paper, he predicts that, by the year 2000, a computer will have a 30% chance of passing a five-minute Turing Test with an unskilled interrogator. What chance do you think a computer would have today? In another 50 years?

1.3 Every year the Loebner prize is awarded to the program that comes closest to passing a version of the Turing test. Research and report on the latest winner of the Loebner prize. What techniques does it use? How does it advance the state of the art in AI?

1.4 There are well-known classes of problems that are intractably difficult for computers, and other classes that are provably undecidable. Does this mean that AI is impossible?

1.5 Suppose we extend Evans's ANALOGY program so that it can score 200 on a standard IQ test. Would we then have a program more intelligent than a human? Explain.

1.6 How could introspection—reporting on one's inner thoughts—be inaccurate? Could I be wrong about what I'm thinking? Discuss.

1.7 Examine the AI literature to discover whether the following tasks can currently be solved by computers:

a. Playing a decent game of table tennis (ping-pong).
b. Driving in the center of Cairo.
c. Buying a week's worth of groceries at the market.
d. Buying a week's worth of groceries on the web.
e. Playing a decent game of bridge at a competitive level.
f. Discovering and proving new mathematical theorems.
g. Writing an intentionally funny story.
h. Giving competent legal advice in a specialized area of law.
i. Translating spoken English into spoken Swedish in real time.
j. Performing a complex surgical operation.

For the currently infeasible tasks, try to find out what the difficulties are and predict when, if ever, they will be overcome.

1.8 Some authors have claimed that perception and motor skills are the most important part of intelligence, and that "higher level" capacities are necessarily parasitic — simple add-ons to these underlying facilities. Certainly, most of evolution and a large part of the brain have been devoted to perception and motor skills, whereas AI has found tasks such as game playing and logical inference to be easier, in many ways, than perceiving and acting in the real world. Do you think that AI's traditional focus on higher-level cognitive abilities is misplaced?

1.9 Why would evolution tend to result in systems that act rationally? What goals are such systems designed to achieve?

1.10 Are reflex actions (such as moving your hand away from a hot stove) rational? Are they intelligent?

1.11 "Surely computers cannot be intelligent — they can do only what their programmers tell them." Is the latter statement true, and does it imply the former?

1.12 "Surely animals cannot be intelligent — they can do only what their genes tell them." Is the latter statement true, and does it imply the former?

1.13 "Surely animals, humans, and computers cannot be intelligent — they can do only what their constituent atoms are told to do by the laws of physics." Is the latter statement true, and does it imply the former?

26 PHILOSOPHICAL FOUNDATIONS

In which we consider what it means to think and whether artifacts could and should ever do so.

As we mentioned in Chapter 1, philosophers have been around for much longer than computers and have been trying to resolve some questions that relate to AI: How do minds work? Is it possible for machines to act intelligently in the way that people do, and if they did, would they have minds? What are the ethical implications of intelligent machines? For the first 25 chapters of this book, we have considered questions from AI itself; now we consider the philosopher's agenda for one chapter.

First, some terminology: the assertion that machines could possibly act intelligently (or, perhaps better, act *as if* they were intelligent) is called the **weak AI** hypothesis by philosophers, and the assertion that machines that do so are *actually* thinking (as opposed to *simulating* thinking) is called the **strong AI** hypothesis.

WEAK AI

STRONG AI

Most AI researchers take the weak AI hypothesis for granted, and don't care about the strong AI hypothesis—as long as their program works, they don't care whether you call it a simulation of intelligence or real intelligence. All AI researchers should be concerned with the ethical implications of their work.

26.1 WEAK AI: CAN MACHINES ACT INTELLIGENTLY?

Some philosophers have tried to prove that AI is impossible; that machines cannot possibly act intelligently. Some have used their arguments to call for a stop to AI research:

> Artificial intelligence *pursued within the cult of computationalism* stands not even a ghost of a chance of producing durable results ... it is time to divert the efforts of AI researchers—and the considerable monies made available for their support—into avenues other than the computational approach. (Sayre, 1993)

Clearly, whether AI is impossible depends on how it is defined. In essence, AI is the quest for the best agent program on a given architecture. With this formulation, AI is by definition possible: for any digital architecture consisting of k bits of storage there are exactly 2^k agent

programs, and all we have to do to find the best one is enumerate and test them all. This might not be feasible for large k, but philosophers deal with the theoretical, not the practical.

Our definition of AI works well for the engineering problem of finding a good agent, given an architecture. Therefore, we're tempted to end this section right now, answering the title question in the affirmative. But philosophers are interested in the problem of comparing two architectures—human and machine. Furthermore, they have traditionally posed the question as, "**Can** machines think?" Unfortunately, this question is ill-defined. To see why, consider the following questions:

CAN MACHINES THINK?

- Can machines fly?
- Can machines swim?

Most people agree that the answer to the first question is yes, airplanes can fly, but the answer to the second is no; boats and submarines do move through the water, but we do not call that swimming. However, neither the questions nor the answers have any impact at all on the working lives of aeronautic and naval engineers or on the users of their products. The answers have very little to do with the design or capabilities of airplanes and submarines, and much more to do with the way we have chosen to use words. The word "swim" in English has come to mean "to move along in the water by movement of body parts," whereas the word "fly" has no such limitation on the means of locomotion.[1] The practical possibility of "thinking machines" has been with us for only 50 years or so, not long enough for speakers of English to settle on a meaning for the word "think."

Alan Turing, in his famous paper "Computing Machinery and Intelligence" (Turing, 1950), suggested that instead of asking whether machines can think, we should ask whether machines can pass a behavioral intelligence test, which has come to be called the Turing Test. The test is for a program to have a conversation (via online typed messages) with an interrogator for 5 minutes. The interrogator then has to guess if the conversation is with a program or a person; the program passes the test if it fools the interrogator 30% of the time. Turing conjectured that, by the year 2000, a computer with a storage of 10^9 units could be programmed well enough to pass the test, but he was wrong. Some people **have** been fooled for 5 minutes; for example, the ELIZA program and the Internet chatbot called MGONZ have fooled humans who didn't realize they might be talking to a program, and the program ALICE fooled one judge in the 2001 Loebner Prize competition. But no program has come close to the 30% criterion against trained judges, and the field of AI as a whole has paid little attention to Turing tests.

Turing also examined a wide variety of possible objections to the possibility of intelligent machines, including virtually all of those that have been raised in the half century since his paper appeared. We will look at some of them.

The argument from disability

The "argument from disability" makes the claim that "a machine can never do X" As examples of X, Turing lists the following:

[1] In Russian, the equivalent of "swim" ***does*** apply to ships.

> Be kind, resourceful, beautiful, friendly, have initiative, have a sense of humor, tell right from wrong, make mistakes, fall in love, enjoy strawberries and cream, make someone fall in love with it, learn from experience, use words properly, be the subject of its own thought, have as much diversity of behavior as man, do something really new.

Turing had to use his intuition to guess what would be possible in the future, but we have the luxury of looking back at what computers have already done. It is undeniable that computers now do many things that previously were the domain of humans alone. Programs play chess, checkers and other games, inspect parts on assembly lines, check the spelling of word processing documents, steer cars and helicopters, diagnose diseases, and do hundreds of other tasks as well as or better than humans. Computers have made small but significant discoveries in astronomy, mathematics, chemistry, mineralogy, biology, computer science, and other fields. Each of these required performance at the level of a human expert.

Given what we now know about computers, it is not surprising that they do well at combinatorial problems such as playing chess. But algorithms also perform at human levels on tasks that seemingly involve human judgment, or as Turing put it, "learning from experience" and the ability to "tell right from wrong." As far back as 1955, Paul Meehl (see also Grove and Meehl, 1996) studied the decision-making processes of trained experts at subjective tasks such as predicting the success of a student in a training program, or the recidivism of a criminal. In 19 out of the 20 studies he looked at, Meehl found that simple statistical learning algorithms (such as linear regression or naive Bayes) predict better than the experts. The Educational Testing Service has used an automated program to grade millions of essay questions on the GMAT exam since 1999. The program agrees with human graders 97% of the time, about the same level that two human graders agree (Burstein *et al.*, 2001).

It is clear that computers can do many things as well as or better than humans, including things that people believe require great human insight and understanding. This does not mean, of course, that computers use insight and understanding in performing these tasks—those are not part of behavior, and we address such questions elsewhere—but the point is that one's first guess about the mental processes required to produce a given behavior is often wrong. It is also true, of course, that there are many tasks at which computers do not yet excel (to put it mildly), including Turing's task of carrying on an open-ended conversation.

The mathematical objection

It is well known, through the work of Turing (1936) and Gödel (1931), that certain mathematical questions are in principle unanswerable by particular formal systems. Gödel's incompleteness theorem (see Section 9.5) is the most famous example of this. Briefly, for any formal axiomatic system $\mathbf{F}$ powerful enough to do arithmetic, it is possible to construct a so-called "Gödel sentence" $G(F)$ with the following properties:

- $G(F)$ is a sentence of $\mathbf{F}$, but cannot be proved within $\mathbf{F}$.
- If F is consistent, then $G(F)$ is true.

Philosophers such as J. R. Lucas (1961) have claimed that this theorem shows that machines are mentally inferior to humans, because machines are formal systems that are limited by the incompleteness theorem—they cannot establish the truth of their own Gödel sentence—while

humans have no such limitation. This claim has caused decades of controversy, spawning a vast literature including two books by the mathematician Sir Roger Penrose (1989, 1994) that repeat the claim with some fresh twists (such as the hypothesis that humans are different because their brains operate by quantum gravity). We will examine only three of the problems with the claim.

First, Godel's incompleteness theorem applies only to formal systems that are powerful enough to do arithmetic. This includes Turing machines, and Lucas's claim is in part based on the assertion that computers are Turing machines. This is a good approximation, but is not quite true. Turing machines are infinite, whereas computers are finite, and any computer can therefore be described as a (very large) system in propositional logic, which is not subject to Gödel's incompleteness theorem.

Second, an agent should not be too ashamed that it cannot establish the truth of some sentence while other agents can. Consider the sentence

J. R. Lucas cannot consistently assert that this sentence is true.

If Lucas asserted this sentence then he would be contradicting himself, so therefore Lucas cannot consistently assert it, and hence it must be true. (The sentence cannot be false, because if it were then Lucas could not consistently assert it, so it would be true.) We have thus demonstrated that there is a sentence that Lucas cannot consistently assert while other people (and machines) can. But that does not make us think less of Lucas. To take another example, no human could compute the sum of 10 billion 10 digit numbers in his or her lifetime, but a computer could do it in seconds. Still, we do not see this as a fundamental limitation in the human's ability to think. Humans were behaving intelligently for thousands of years before they invented mathematics, so it is unlikely that mathematical reasoning plays more than a peripheral role in what it means to be intelligent.

Third, and most importantly, even if we grant that computers have limitations on what they can prove, there is no evidence that humans are immune from those limitations. It is all too easy to show rigorously that a formal system cannot do X, and then claim that humans can do X using their own informal method, without giving any evidence for this claim. Indeed, it is impossible to prove that humans are not subject to Gödel's incompleteness theorem, because any rigorous proof would itself contain a formalization of the claimed unformalizable human talent, and hence refute itself. So we are left with an appeal to intuition that humans can somehow perform superhuman feats of mathematical insight. This appeal is expressed with arguments such as "we must assume our own consistency, if thought is to be possible at all" (Lucas, 1976). But if anything, humans are known to be inconsistent. This is certainly true for everyday reasoning, but it is also true for careful mathematical thought. A famous example is the four-color map problem. Alfred Kempe published a proof in 1879 that was widely accepted and contributed to his election as a Fellow of the Royal Society. In 1890, however, Percy Heawood pointed out a flaw and the theorem remained unproved until 1977.

The argument from informality

One of the most influential and persistent criticisms of AI as an enterprise was raised by Turing as the "argument from informality of behavior." Essentially, this is the claim that human

behavior is far too complex to be captured by any simple set of rules and that because computers can do no more than follow a set of rules, they cannot generate behavior as intelligent as that of humans. The inability to capture everything in a set of logical rules is called the **qualification problem** in AI. (See Chapter 10.)

The principal proponent of this view has been the philosopher Hubert Dreyfus, who has produced a series of influential critiques of artificial intelligence: *What Computers Can't Do* (1972), *What Computers Still Can't Do* (1992), and, with his brother Stuart, *Mind Over Machine* (1986).

The position they criticize came to be called "Good (Old-FashionedAI," or GOFAI, a term coined by Haugeland (1985). GOFAI is supposed to claim that all intelligent behavior can be captured by a system that reasons logically from a set of facts and rules describing the domain. It therefore corresponds to the simplest logical agent described in Chapter 7. Dreyfus is correct in saying that logical agents are vulnerable to the qualification problem. As we saw in Chapter 13, probabilistic reasoning systems are more appropriate for open-ended domains. The Dreyfus critique therefore is not addressed against computers *per se,* but rather against one particular way of programming them. It is reasonable to suppose, however, that a book called *What First-Order Logical Rule-Based Systems Without Learning Can't Do* might have had less impact.

Under Dreyfus's view, human expertise does include knowledge of some rules, but only as a "holistic context" or "background" within which humans operate. He gives the example of appropriate social behavior in giving and receiving gifts: "Normally one simply responds in the appropriate circumstances by giving an appropriate gift." One apparently has "a direct sense of how things are done and what to expect." The same claim is made in the context of chess playing: "A mere chess master might need to figure out what to do, but a grandmaster just sees the board as demanding a certain move . . .the right response just pops into his or her head." It is certainly true that much of the thought processes of a present-giver or grandmaster is done at a level that is not open to introspection by the conscious mind. But that does not mean that the thought processes do not exist. The important question that Dreyfus does not answer is *how* the right move gets into the grandmaster's head. One is reminded of Daniel Dennett's (1984) comment,

> It is rather as if philosophers were to proclaim themselves expert explainers of the methods of stage magicians, and then, when we ask how the magician does the sawing-the-lady-in-half trick. they explain that it is really quite obvious: the magician doesn't really saw her in half; he simply makes it appear that he does. "But how does he do *that?'* we ask. "Not our department," say the philosophers.

Dreyfus and Dreyfus (1986) propose a five-stage process of acquiring expertise, beginning with rule-based processing (of the sort proposed in GOFAI) and ending with the ability to select correct responses instantaneously. In making this proposal, Dreyfus and Dreyfus in effect move from being AI critics to AI theorists—they propose a neural network architecture organized into a vast "case library," but point out several problems. Fortunately, all of their problems have been addressed, some with partial success and some with total success. Their problems include:

1. Good generalization from examples cannot be achieved without background knowledge. They claim no one has any idea how to incorporate background knowledge into the neural network learning process. In fact, we saw in Chapter 19 that there are techniques for using prior knowledge in learning algorithms. Those techniques, however, rely on the availability of knowledge in explicit form, something that Dreyfus and Dreyfus strenuously deny. In our view, this is a good reason for a serious redesign of current models of neural processing so that they *can* take advantage of previously learned knowledge in the way that other learning algorithms do.
2. Neural network learning is a form of supervised learning (see Chapter 18), requiring the prior identification of relevant inputs and correct outputs. Therefore, they claim, it cannot operate autonomously without the help of a human trainer. In fact, learning without a teacher can be accomplished by unsupervised learning (Chapter 20) and reinforcement learning (Chapter 21).
3. Learning algorithms do not perform well with many features, and if we pick a subset of features, "there is no known way of adding new features should the current set prove inadequate to account for the learned facts." In fact, new methods such as support vector machines handle large feature sets very well. As we saw in Chapter 19, there are also principled ways to generate new features, although much more work is needed.
4. The brain is able to direct its sensors to seek relevant information and to process it to extract aspects relevant to the current situation. But, they claim, "Currently, no details of this mechanism are understood or even hypothesized in a way that could guide AI research." In fact, the field of active vision, underpinned by the theory of information value (Chapter 16), is concerned with exactly the problem of directing sensors, and already some robots have incorporated the theoretical results obtained.

In sum, many of the issues Dreyfus has focused on—background commonsense knowledge, the qualification problem, uncertainty, learning, compiled forms of decision making, the importance of considering situated agents rather than disembodied inference engines—have by now been incorporated into standard intelligent agent design. In our view, this is evidence of AI's progress, not of its impossibility.

26.2 Strong AI: Can Machines Really Think?

Many philosophers have claimed that a machine that passes the Turing Test would still not be *actually* thinking, but would be only a *simulation* of thinking. Again, the objection was foreseen by Turing. He cites a speech by Professor Geoffrey Jefferson (1949):

> Not until a machine could write a sonnet or compose a concerto because of thoughts and emotions felt, and not by the chance fall of symbols, could we agree that machine equals brain—that is, not only write it but know that it had written it.

Turing calls this the argument from consciousness—the machine has to be aware of its own mental states and actions. While consciousness is an important subject, Jefferson's key point

actually relates to phenomenology, or the study of direct experience—the machine has to actually feel emotions. Others focus on intentionality—that is, the question of whether the machine's purported beliefs, desires, and other representations are actually "about" something in the real world.

Turing's response to the objection is interesting. He could have presented reasons that machines can in fact be conscious (or have phenomenology, or have intentions). Instead, he maintains that the question is just as ill-defined as asking, "Can machines think?" Besides, why should we insist on a higher standard for machines than we do for humans? After all, in ordinary life we never have any direct evidence about the internal mental states of other humans. Nevertheless, Turing says, "Instead of arguing continually over this point, it is usual to have the polite convention that everyone thinks."

POLITE CONVENTION

Turing argues that Jefferson would be willing to extend the polite convention to machines if only he had experience with ones that act intelligently. He cites the following dialog, which has become such a part of AI's oral tradition that we simply have to include it:

> HUMAN: In the first line of your sonnet which reads "shall I compare thee to a summer's day," would not a "spring day" do as well or better?
> MACHINE: It wouldn't scan.
> HUMAN: How about "a winter's day." That would scan all right.
> MACHINE: Yes, but nobody wants to be compared to a winter's day.
> HUMAN: Would you say Mr. Pickwick reminded you of Christmas?
> MACHINE: In a way.
> HUMAN: Yet Christmas is a winter's day, and I do not think Mr. Pickwick would mind the comparison.
> MACHINE: I don't think you're serious. By a winter's day one means a typical winter's day, rather than a special one like Christmas.

Turing concedes that the question of consciousness is a difficult one, but denies that it has much relevance to the practice of AI: "I do not wish to give the impression that I think there is no mystery about consciousness ... But I do not think these mysteries necessarily need to be solved before we can answer the question with which we are concerned in this paper." We agree with Turing—we are interested in creating programs that behave intelligently, not in whether someone else pronounces them to be real or simulated. On the other hand, many philosophers are keenly interested in the question. To help understand it, we will consider the question of whether other artifacts are considered real.

In 1848, artificial urea was synthesized for the first time, by Frederick Wohler. This was important because it proved that organic and inorganic chemistry could be united, a question that had been hotly debated. Once the synthesis was accomplished, chemists agreed that artificial urea ***was*** urea, because it had all the right physical properties. Similarly, artificial sweeteners are undeniably sweeteners, and artificial insemination (the other AI) is undeniably insemination. On the other hand, artificial flowers are not flowers, and Daniel Dennett points out that artificial Chateau Latour wine would not be Chateau Latour wine, even if it was chemically indistinguishable, simply because it was not made in the right place in the right way. Nor is an artificial Picasso painting a Picasso painting, no matter what it looks like.

We can conclude that in some cases, the behavior of an artifact is important, while in others it is the artifact's pedigree that matters. Which one is important in which case seems to be a matter of convention. But for artificial minds, there is no convention, and we are left to rely on intuitions. The philosopher John Searle (1980) has a strong one:

> No one supposes that a computer simulation of a storm will leave us all wet ... Why on earth would anyone in his right mind suppose a computer simulation of mental processes actually had mental processes? (pp. 37–38)

While it is easy to agree that computer simulations of storms do not make us wet, it is not clear how to carry this analogy over to computer simulations of mental processes. After all, a Hollywood simulation of a storm using sprinklers and wind machines *does* make the actors wet. Most people are comfortable saying that a computer simulation of addition is addition, and a computer simulation of a chess game is a chess game. Are mental processes more like storms, or more like addition or chess? Like Chateau Latour and Picasso, or like urea? That all depends on your theory of mental states and processes.

FUNCTIONALISM

The theory of **functionalism** says that a mental state is any intermediate causal condition between input and output. Under functionalist theory, any two systems with isomorphic causal processes would have the same mental states. Therefore, a computer program could have the same mental states as a person. Of course, we have not yet said what "isomorphic" really means, but the assumption is that there is some level of abstraction below which the specific implementation does not matter; as long as the processes are isomorphic down to the this level, the same mental states will occur.

BIOLOGICAL NATURALISM

In contrast, the **biological naturalism** theory says that mental states are high-level emergent features that are caused by low-level neurological processes *in the neurons,* and it is the (unspecified) properties of the neurons that matter. Thus, mental states cannot be duplicated just on the basis of some program having the same functional structure with the same input–output behavior; we would require that the program be running on an architecture with the same causal power as neurons. The theory does not say why neurons have this causal power, nor what other physical instantiations might or might not have it.

To investigate these two viewpoints we will first look at one of the oldest problems in the philosophy of mind, and then turn to three thought experiments.

The mind–body problem

MIND-BODY PROBLEM

The **mind–body problem** asks how mental states and processes are related to bodily (specifically, brain) states and processes. As if that wasn't hard enough, we will generalize the problem to the "mind–architecture" problem, to allow us to talk about the possibility of machines having minds.

Why is the mind–body problem a problem? The first difficulty goes back to René Descartes, who considered how an immortal soul interacts with a mortal body and concluded

DUALISM

that the soul and body are two distinct types of things—a **dualist** theory. The **monist** theory,

MONISM

often called **materialism,** holds that there are no such things as immaterial souls; only mate-

MATERIALISM

rial objects. Consequently, mental states—such as being in pain, knowing that one is riding a horse, or believing that Vienna is the capital of Austria—are brain states. John Searle pithily

sums up the idea with the slogan, *"Brains cause minds."*

The materialist must face at least two serious obstacles. The first is the problem of

FREE WILL

free will: how can it be that a purely physical mind, whose every transformation is governed strictly by the laws of physics, still retains any freedom of choice? Most philosophers regard this problem as requiring a careful reconstitution of our naive notion of free will, rather than presenting any threat to materialism. The second problem concerns the general issue of **con-**

CONSCIOUSNESS

sciousness (and related, but not identical, questions of **understanding** and **self-awareness).** Put simply, why is it that it *feels* like something to have certain brain states, whereas it presumably does not feel like anything to have other physical states (e.g., being a rock).

To begin to answer such questions, we need ways to talk about brain states at levels more abstract than specific configurations of all the atoms of the brain of a particular person at a particular time. For example, as I think about the capital of Austria, my brain undergoes myriad tiny changes from one picosecond to the next, but these do not constitute a *qualitative* change in brain state. To account for this, we need a notion of brain state *types,* under which we can judge whether two brain states belong to the same or different types. Various authors have various positions on what one means by *type* in this case. Almost everyone believes that if one takes a brain and replaces some of the carbon atoms by a new set of carbon atoms,[2] the mental state will not be affected. This is a good thing because real brains are continually replacing their atoms through metabolic processes, and yet this in itself does not seem to cause major mental upheavals.

Now let's consider a particular kind of mental state: the **propositional attitudes** (first

INTENTIONAL STATE

discussed in Chapter 10), which are also known as **intentional states.** These are states, such as believing, knowing, desiring, fearing, and so on, that refer to some aspect of the external world. For example, the belief that Vienna is the capital of Austria is a belief about a particular city and its status. We will be asking whether it is possible for computers to have intentional states, so it helps to understand how to characterize such stales. For example, one might say that the mental state in which I desire a hamburger differs from the state in which I desire a pizza because hamburger and pizza are different things in the real world. That is to say, intentional states have a necessary connection to their objects in the external world. On the other hand, we argued just a few paragraphs back that mental states are brain states; hence the identity or non-identity of mental states should be determined by staying completely "inside the head," without reference to the real world. To examine this dilemma we turn to a thought experiment that attempts to separate intentional states from their external objects.

The "brain in a vat" experiment

Imagine, if you will, that your brain was removed from your body at birth and placed in a marvelously engineered vat. The vat sustains your brain, allowing it to grow and develop. At the same time, electronic signals are fed to your brain from a computer simulation of an entirely fictitious world, and motor signals from your brain are intercepted and used to modify the simulation as appropriate.[3] Then the brain could have the mental state

[2] Perhaps even atoms of a different isotope of carbon, as is sometimes done in brain-scanning experiments.

[3] This situation may be familiar to those who have seen the 1999 film, *The* **Matrix.**

DyingFor(*Me*, Hamburger) even though it has no body to feel hunger and no taste buds to experience taste, and there may be no hamburger in the real world. In that case, would this be the same mental state as one held by a brain in a body?

WIDE CONTENT

One way to resolve the dilemma is to say that the content of mental states can be interpreted from two different points of view. The **"wide content"** view interprets it from the point of view of an omniscient outside observer with access to the whole situation, who can distinguish differences in the world. So under wide content the brain-in-a-vat beliefs are different from those of a "normal" person. **Narrow content** considers only the internal subjective point of view, and under this view the beliefs would all be the same.

NARROW CONTENT

QUALIA

The belief that a hamburger is delicious has a certain intrinsic nature—there is something that it is like to have this belief. Now we get into the realm of **qualia**, or intrinsic experiences (from the Latin word meaning, roughly, "such things"). Suppose, through some accident of retinal and neural wiring, that person X experiences as red the color that person Y perceives as green, and vice-versa. Then when both see the same traffic light they will act the same way, but the **experience** they have will be in some way different. Both may agree that the name for their experience is "the light is red," but the experiences feel different. It is not clear whether that means they are the same or different mental states.

We now turn to another thought experiment that gets at the question of whether physical objects other than human neurons can have mental states.

The brain prosthesis experiment

The brain prosthesis experiment was introduced in the mid-1970s by Clark Glymour and was touched on by John Searle (1980), but is most commonly associated with the work of Hans Moravec (1988). It goes like this: Suppose neurophysiology has developed to the point where the input–output behavior and connectivity of all the neurons in the human brain are perfectly understood. Suppose further that we can build microscopic electronic devices that mimic this behavior and can be smoothly interfaced to neural tissue. Lastly, suppose that some miraculous surgical technique can replace individual neurons with the corresponding electronic devices without interrupting the operation of the brain as a whole. The experiment consists of gradually replacing all the neurons in someone's head with electronic devices and then reversing the process to return the subject to his or her normal biological state.

We are concerned with both the external behavior and the internal experience of the subject, during and after the operation. By the definition of the experiment, the subject's external behavior must remain unchanged compared with what would be observed if the operation were not carried out.[4] Now although the presence or absence of consciousness cannot easily be ascertained by a third party, the subject of the experiment ought at least to be able to record any changes in his or her own conscious experience. Apparently, there is a direct clash of intuitions as to what would happen. Moravec, a robotics researcher and functionalist, is convinced his consciousness would remain unaffected. Searle, a philosopher and biological naturalist, is equally convinced his consciousness would vanish:

[4] One can imagine using an identical "control" subject who is given a placebo operation, so that the two behaviors can be compared.

> You find, to your total amazement, that you are indeed losing control of your external behavior. You find, for example, that when doctors test your vision, you hear them say "We are holding up a red object in front of you; please tell us what you see." You want to cry out "I can't see anything. I'm going totally blind." But you hear your voice saying in a way that is completely out of your control, "I see a red object in front of me." ... [Y]our conscious experience slowly shrinks to nothing, while your externally observable behavior remains the same. (Searle, 1992)

But one can do more than argue from intuition. First, note that, in order for the external behavior to remain the same while the subject gradually becomes unconscious, it must be the case that the subject's volition is removed instantaneously and totally; otherwise the shrinking of awareness would be reflected in external behavior—"Help, I'm shrinking!" or words to that effect. This instantaneous removal of volition as a result of gradual neuron-at-a-time replacement seems an unlikely claim to have to make.

Second, consider what happens if we do ask the subject questions concerning his or her conscious experience during the period when no real neurons remain. By the conditions of the experiment, we will get responses such as "I feel fine. I must say I'm a bit surprised because I believed Searle's argument." Or we might poke the subject with a pointed stick and observe the response, "Ouch, that hurt." Now, in the normal course of affairs, the skeptic can dismiss such outputs from AI programs as mere contrivances. Certainly, it is easy enough to use a rule such as "If sensor 12 reads 'High' then output 'Ouch.' " But the point here is that, because we have replicated the functional properties of a normal human brain, we assume that the electronic brain contains no such contrivances. Then we must have an explanation of the manifestations of consciousness produced by the electronic brain that appeals only to the functional properties of the neurons. *And this explanation must also apply to the real brain, which has the same functional properties.* There are, it seems, only two possible conclusions:

1. The causal mechanisms of consciousness that generate these kinds of outputs in normal brains are still operating in the electronic version, which is therefore conscious.
2. The conscious mental events in the normal brain have no causal connection to behavior, and are missing from the electronic brain, which is therefore not conscious.

EPIPHENOMENON

Although we cannot rule out the second possibility, it reduces consciousness to what philosophers call an **epiphenomenal** role—something that happens, but casts no shadow, as it were, on the observable world. Furthermore, if consciousness is indeed epiphenomenal, then the brain must contain a second, unconscious mechanism that is responsible for the "Ouch."

Third, consider the situation after the operation has been reversed and the subject has a normal brain. Once again, the subject's external behavior must, by definition, be as if the operation had not occurred. In particular, we should be able to ask, "What was it like during the operation? Do you remember the pointed stick?" The subject must have accurate memories of the actual nature of his or her conscious experiences, including the qualia, despite the fact that, according to Searle there were no such experiences.

Searle might reply that we have not defined the experiment properly. If the real neurons are, say, put into suspended animation between the time they are extracted and the time they are replaced in the brain, then of course they will not "remember" the experiences during

the operation. To deal with this eventuality, we need to make sure that the neurons' state is updated to reflect the internal state of the artificial neurons they are replacing. If the supposed "nonfunctional" aspects of the real neurons then result in functionally different behavior from that observed with artificial neurons still in place, then we have a simple *reductio ad absurdum,* because that would mean that the artificial neurons are not functionally equivalent to the real neurons. (See Exercise 26.3 for one possible rebuttal to this argument.)

Patricia Churchland (1986) points out that the functionalist arguments that operate at the level of the neuron can also operate at the level of any larger functional unit—a clump of neurons, a mental module, a lobe, a hemisphere, or the whole brain. That means that if you accept the notion that the brain prosthesis experiment shows that the replacement brain is conscious, then you should also believe that consciousness is maintained when the entire brain is replaced by a circuit that maps from inputs to outputs via a huge lookup table. This is disconcerting to many people (including Turing himself), who have the intuition that lookup tables are not conscious—or at least, that the conscious experiences generated during table lookup are not the same as those generated during the operation of a system that might be described (even in a simple-minded, computational sense) as accessing and generating beliefs, introspections, goals, and so on. This would suggest that the brain prosthesis experiment cannot use whole-brain-at-once replacement if it is to be effective in guiding intuitions, but it does not mean that it must use one-atom-at-a-time replacement as Searle have us believe.

The Chinese room

Our final thought experiment is perhaps the most famous of all. It is due to John Searle (1980), who describes a hypothetical system that is clearly running a program and passes the Turing Test, but that equally clearly (according to Searle) does not *understand* anything of its inputs and outputs. His conclusion is that running the appropriate program (i.e., having the right outputs) is not a *sufficient* condition for being a mind.

The system consists of a human, who understands only English, equipped with a rule book, written in English, and various stacks of paper, some blank, some with indecipherable inscriptions. (The human therefore plays the role of the CPU, the rule book is the program, and the stacks of paper are the storage device.) The system is inside a room with a small opening to the outside. Through the opening appear slips of paper with indecipherable symbols. The human finds matching symbols in the rule book, and follows the instructions. The instructions may include writing symbols on new slips of paper, finding symbols in the stacks, rearranging the stacks, and so on. Eventually, the instructions will cause one or more symbols to be transcribed onto a piece of paper that is passed back to the outside world.

So far, so good. But from the outside, we see a system that is taking input in the form of Chinese sentences and generating answers in Chinese that are as obviously "intelligent" as those in the conversation imagined by Turing.[5] Searle then argues as follows: the person in the room does not understand Chinese (given). The rule book and the stacks of paper, being

[5] The fact that the stacks of paper might well be larger than the entire planet and the generation of answers would take millions of years has no bearing on the logical structure of the argument. One aim of philosophical training is to develop a finely honed sense of which objections are germane and which are not.

just pieces of paper, do not understand Chinese. Therefore, there is no understanding of Chinese going on. *Hence, according to Searle, running the right program does not necessarily generate understanding.*

Like Turing, Searle considered and attempted to rebuff a number of replies to his argument. Several commentators, including John McCarthy and Robert Wilensky, proposed what Searle calls the systems reply. The objection is that, although one can ask if the human in the room understands Chinese, this is analogous to asking if the CPU can take cube roots. In both cases, the answer is no, and in both cases, according to the systems reply, the entire system *does* have the capacity in question. Certainly, if one asks the Chinese room whether it understands Chinese, the answer would be affirmative (in fluent Chinese). By Turing's polite convention, this should be enough. Searle's response is to reiterate the point that the understanding is not in the human and cannot be in the paper, so there cannot be any understanding. He further suggests that one could imagine the human memorizing the rule book and the contents of all the stacks of paper, so that there would be nothing to have understanding *except* the human; and again, when one asks the human (in English), the reply will be in the negative.

Now we are down to the real issues. The shift from paper to memorization is a red herring, because both forms are simply physical instantiations of a running program. The real claim made by Searle rests upon the following four axioms (Searle, 1990):

1. Computer programs are formal, syntactic entities.
2. Minds have mental contents, or semantics.
3. Syntax by itself is not sufficient for semantics.
4. Brains cause minds.

From the first three axioms he concludes that programs are not sufficient for minds. In other words, an agent running a program might be a mind, but it is not necessarily a mind just by virtue of running the program. From the fourth axiom he concludes "Any other system capable of causing minds would have to have causal powers (at least) equivalent to those of brains." From there he infers that any artificial brain would have to duplicate the causal powers of brains, not just run a particular program, and that human brains do not produce mental phenomena solely by virtue of running a program.

The conclusions that programs are not sufficient for minds *does* follow from the axioms, if you are generous in interpreting them. But the conclusion is unsatisfactory—all Searle has shown is that if you explicitly deny functionalism (that is what his axiom (3) does) then you can't necessarily conclude that non-brains are minds. This is reasonable enough, so the whole argument comes down to whether axiom (3) can be accepted. According to Searle, the point of the Chinese room argument is to provide intuitions for axiom (3). But the reaction to his argument shows that it provides intuitions only to those who were already inclined to accept the idea that mere programs cannot generate true understanding.

To reiterate, the aim of the Chinese Room argument is to refute strong AI—the claim that running the right sort of program necessarily results in a mind. It does this by exhibiting an apparently intelligent system running the right sort of program that is, according to Searle, *demonstrably* not a mind. Searle appeals to intuition, not proof, for this part: just look at the room; what's there to be a mind? But one could make the same argument about the brain:

just look at this collection of cells (or of atoms), blindly operating according to the laws of biochemistry (or of physics)—what's there to be a mind? Why can a hunk of brain be a mind while a hunk of liver cannot?

Furthermore, when Searle admits that materials other than neurons could in principle be a mind, he weakens his argument even further, for two reasons: first, one has only Searle's intuitions (or one's own) to say that the Chinese room is not a mind, and second, even if we decide the room is not a mind, that tells us nothing about whether a program running on some other physical medium (including a computer) might be a mind.

Searle allows the logical possibility that the brain is actually implementing an AI program of the traditional sort—but the same program running on the wrong kind of machine would not be a mind. Searle has denied that he believes that "machines cannot have minds," rather, he asserts that some machines *do* have minds—humans are biological machines with minds. We are left without much guidance as to what types of machines do or do not qualify.

26.3 The Ethics and Risks of Developing Artificial Intelligence

So far, we have concentrated on whether we *can* develop AI, but we must also consider whether we *should*. If the effects of AI technology are more likely to be negative than positive, then it would be the moral responsibility of workers in the field to redirect their research. Many new technologies have had unintended negative side-effects: the internal combustion engine brought air pollution and the paving-over of paradise; nuclear fission brought Chernobyl, Three Mile Island, and the threat of global destruction. All scientists and engineers face ethical considerations of how they should act on the job, what projects should or should not be done, and how they should be handled. There is even a handbook on the *Ethics of Computing* (Berleur and Brunnstein, 2001). AI, however, seems to pose some fresh problems beyond that of, say, building bridges that don't fall down:

- People might lose their jobs to automation.
- People might have too much (or too little) leisure time.
- People might lose their sense of being unique.
- People might lose some of their privacy rights.
- The use of AI systems might result in a loss of accountability.
- The success of AI might mean the end of the human race.

We will look at each issue in turn.

People might lose their jobs to automation. The modern industrial economy has become dependent on computers in general, and select AI programs in particular. For example, much of the economy, especially in the United States, depends on the availability of consumer credit. Credit card applications, charge approvals, and fraud detection are now done by AI programs. One could say that thousands of workers have been displaced by these AI programs, but in fact if you took away the AI programs these jobs would not exist, because human labor would add an unacceptable cost to the transactions. So far, automation via AI

technology has created more jobs than it has eliminated, and has created more interesting, higher-paying jobs. Now that the canonical AI program is an "intelligent agent" designed to assist a human, loss of jobs is less of a concern than it was when AI focused on "expert systems" designed to replace humans.

People might have too much (or too little) leisure time. Alvin Toffler wrote in *Future Shock* (1970), "The work week has been cut by 50 percent since the turn of the century. It is not out of the way to predict that it will be slashed in half again by 2000." Arthur C. Clarke (1968b) wrote that people in 2001 might be "faced with a future of utter boredom, where the main problem in life is deciding which of several hundred TV channels to select." The only one of these predictions that has come close to panning out is the number of TV channels (Springsteen, 1992). Instead, people working in knowledge-intensive industries have found themselves part of an integrated computerized system that operates 24 hours a day; to keep up, they have been forced to work *longer* hours. In an industrial economy, rewards are roughly proportional to the time invested; working 10% more would tend to mean a 10% increase in income. In an information economy marked by high-bandwidth communication and easy replication of intellectual property (what Frank and Cook (1996) call the "Winner-Take-All Society"), there is a large reward for being slightly better than the competition; working 10% more could mean a 100% increase in income. So there is increasing pressure on everyone to work harder. AI increases the pace of technological innovation and thus contributes to this overall trend, but AI also holds the promise of allowing us to take some time off and let our automated agents handle things for a while.

People might lose their sense of being unique. In *Computer Power and Human Reason,* Weizenbaum (1976), the author of the ELIZA program, points out some of the potential threats that AI poses to society. One of Weizenbaum's principal arguments is that AI research makes possible the idea that humans are automata—an idea that results in a Ross of autonomy or even of humanity. We note that the idea has been around much longer than AI, going back at least to *L'Homme Machine* (La Mettrie, 1748). We also note that humanity has survived other setbacks to our sense of uniqueness: *De Revolutionibus Orbium Coelestium* (Copernicus, 1543) moved the Earth away from the center of the solar system and *Descent of Man* (Darwin, 1871) put *Homo sapiens* at the same level as other species. AI, if widely successful, may be at least as threatening to the moral assumptions of 21st-century society as Darwin's theory of evolution was to those of the 19th century.

People might lose some of their privacy rights. Weizenbaum also pointed out that speech recognition technology could lead to widespread wiretapping, and hence to a loss of civil liberties. He didn't foresee a world with terrorist threats that would change the balance of how much surveillance people are willing to accept, but he did correctly recognize that AI has the potential to mass-produce surveillance. His prediction may have come true: the U.S. government's classified Echelon system "consists of a network of listening posts, antenna fields, and radar stations; the system is backed by computers that use language translation, speech recognition, and keyword searching to automatically sift through telephone, email, fax, and telex traffic."[6] Some accept that computerization leads to a loss of privacy—Sun

[6] See "Eavesdropping on Europe," Wired news, 9130 11998, and cited EU reports.

Microsystems CEO Scott McNealy has said "You have zero privacy anyway. Get over it." Others disagree: Judge Louis Brandeis wrote in 1890, "Privacy is the most comprehensive of all rights ... the right to one's personality."

The use of AI systems might result in a loss of accountability. In the litigious atmosphere that prevails in the United States, legal liability becomes an important issue. When a physician relies on the judgment of a medical expert system for a diagnosis, who is at fault if the diagnosis is wrong? Fortunately, due in part to the growing influence of decision-theoretic methods in medicine, it is now accepted that negligence cannot be shown if the physician performs medical procedures that have high *expected* utility, even if the *actual* result is catastrophic for the patient. The question should therefore be "Who is at fault if the diagnosis is unreasonable?" So far, courts have held that medical expert systems play the same role as medical textbooks and reference books; physicians are responsible for understanding the reasoning behind any decision and for using their own judgment in deciding whether to accept the system's recommendations. In designing medical expert systems as agents, therefore, the actions should be thought of not as directly affecting the patient but as influencing the physician's behavior. If expert systems become reliably more accurate than human diagnosticians, doctors might become legally liable if they *don't* use the recommendations of an expert system. Gawande (2002) explores this premise.

Similar issues are beginning to arise regarding the use of intelligent agents on the Internet. Some progress has been made in incorporating constraints into intelligent agents so that they cannot, for example, damage the files of other users (Weld and Etzioni, 1994). The problem is magnified when money changes hands. If monetary transactions are made "on one's behalf" by an intelligent agent, is one liable for the debts incurred? Would it be possible for an intelligent agent to have assets itself and to perform electronic trades on its own behalf? So far, these questions do not seem to be well understood. To our knowledge, no program has been granted legal status as an individual for the purposes of financial transactions; at present, it seems unreasonable to do so. Programs are also not considered to be "drivers" for the purposes of enforcing traffic regulations on real highways. In California law, at least, there do not seem to be any legal sanctions to prevent an automated vehicle from exceeding the speed limits, although the designer of the vehicle's control mechanism would be liable in the case of an accident. As with human reproductive technology, the law has yet to catch up with the new developments.

The success of AI might mean the end of the human race. Almost any technology has the potential to cause harm in the wrong hands, but with AI and robotics, we have the new problem that the wrong hands might belong to the technology itself. Countless science fiction stories have warned about robots or robot–human cyborgs running amok. Early examples include Mary Shelley's *Frankenstein, or the Modern Prometheus* (1818)[7] and Karel Capek's play R.U.R (1921), in which robots conquer the world. In movies, we have *The Terminator* (1984), which combines the cliches of robots-conquer-the-world with time travel, and *The Matrix* (1999), which combines robots-conquer-the-world with brain-in-a-vat.

[7] As a young man, Charles Babbage was influenced by reading *Frankenstein.*

For the most past, it seems that robots are the protagonists of so many conquer-the-warld stories because they represent the unknown, just like the witches and ghosts of tales from earlier eras. Do they pose a more credible threat than witches and ghosts? If robots are properly designed as agents that adopt their owner's goals, then they probably do not: robots that derive from incremental advances over current designs will serve, not conquer. Humans use their intelligence in aggressive ways because humans have some innately aggressive tendencies, due to natural selection. But the machines we build need not be innately aggressive, unless we decide to build them that way. On the other hand, it is possible that computers will achieve a sort of conquest by serving and becoming irtdispensable, just as automobiles have in a sense conquered the industrialized world. One scenario deserves further consideration. I. J. Good wrote (1965),

> Let an ultraintelligent machine be defined as a machine that can far surpass all the intellectual activities of any man however clever. Since the design of machines is one of these intellectual activities, an ultraintelligent machine could design even better machines; there would then unquestionably be an "intelligence explosion," and the intelligence of man would be left far behind. Thus the first ultraintelligent machine is the *last* invention that man need ever make, provided that the machine is docile enough to tell us how to keep it under control.

TECHNOLOGICAL SINGULARITY

The "intelligence explosion" has also been called the technological singularity by mathematics professor and science fiction author Vernor Vinge, who writes (1993), "Within thirty years, we will have the technological means to create superhuman intelligence. Shortly after, the human era will be ended." Good and Vinge (and many others) correctly note that the curve of technological progress is growing exponentially at present (consider Moore's Law). However, it is quite a step to extrapolate that the curve will continue on to a singularity of near-infinite growth. So far, every other technology has followed an S-shaped curve, where the exponential growth eventually tapers off.

Vinge is concerned and scared about the coming singularity, but other computer scientists and futurists relish it. Hans Moravec's *Robot: Mere Machine to Transcendent Mind* predicts that robots will match human intelligence in 50 years and then exceed it. He writes,

> Rather quickly, they could displace us from existence. I'm not as alarmed as many by the latter possibility, since I consider these future machines our progeny, "mind children" built in our image and likeness, ourselves in more potent form. Like biological children of previous generations, they will embody humanity's best hope for a long-term future. It behooves us to give them every advantage, and to bow out when we can no longer contribute. (Moravec, 2000)

Ray Kurzweil, in *The Age* d *Spiritual Machines* (2000), predicts that by the year 2099 there will be "a strong trend toward a merger of human thinking with the world of machine intelligence that the human species initially created. There is no longer any clear distinction between humans and computers." There is even a new word--transhumanism—for the active social movement that looks forward to this future. Suffice it to say that such issues present a challenge for most moral theorists, who take the preservation of human life and the human species to be a good thing.

TRANSHUMANISM

Finally, let us consider the robot's point of view. If robots become conscious, then to treat them as mere "machines" (e.g., to take them apart) might be immoral. Robots also must themselves act morally—we would need to program them with a theory of what is right and wrong. Science fiction writers have addressed the issue of robot rights and responsibilities, starting with Isaac Asimov (1942). The well-known movie A.I. (Spielberg, 2001) was based on a story by Brian Aldiss about an intelligent robot who was programmed to believe that he was human and fails to understand his eventual abandonment by his owner–mother. The story (and the movie) convince one of the need for a civil rights movement for robots.

26.4 SUMMARY

This chapter has addressed the following issues:

- Philosophers use the term **weak AI** for the hypothesis that machines could possibly behave intelligently, and **strong AI** for the hypothesis that such machines would count as having actual minds (as opposed to simulated minds).
- Alan Turing rejected the question "Can machines think?" and replaced it with a behavioral test. He anticipated many objections to to the possibility of thinking machines. Few AI researchers pay attention to the Turing test, preferring to concentrate on their systems' performance on practical tasks, rather than the ability to imitate humans.
- There is general agreement in modern times that mental states are brain states.
- Arguments for and against strong AI are inconclusive. Few mainstream AI researchers believe that anything significant hinges on the outcome of the debate.
- Consciousness remains a mystery.
- We identified six potential threats to society posed by AI and related technology. We concluded that some of the threats are either unlikely or differ little from threats posed by other, "unintelligent" technologies. One threat in particular is worthy of further consideration: that ultraintelligent machines might lead to a future that is very different from today—we may not like it, and at that point we may not have a choice. Such considerations lead inevitably to the conclusion that we must weigh carefully, and soon, the possible consequences of AI research for the future of the human race.

BIBLIOGRAPHICAL AND HISTORICAL NOTES

The nature of the mind has been a standard topic of philosophical theorizing from ancient times to the present. In the *Phaedo,* Plato specifically considered and rejected the idea that the mind could be an "attunement" or pattern of organization of the parts of the body, a viewpoint that approximates the functionalist viewpoint in modern philosophy of mind. He decided instead that the mind had to be an immortal, immaterial soul, separable from the body and different in substance—the viewpoint of dualism. Aristotle distinguished a variety

of souls (Greek $\psi\upsilon\chi\eta$) in living things, some of which, at least, he described in a functionalist manner. (See Nussbaum (1978) for more on Aristotle's functionalism.)

Descartes is notorious for his dualistic view of the human mind, but ironically his historical influence was toward mechanism and materialism. He explicitly conceived of animals as automata, and he anticipated the Turing test, writing "it is not conceivable [that a machine] should produce different arrangements of words so as to give an appropriately meaningful answer to whatever is said in its presence, as even the dullest of men can do" (Descartes, 1637). Descartes's spirited defense of the animals-as-automata viewpoint actually had the effect of making it easier to conceive of humans as automata as well, even though he himself did not take this step. The book *L'Homme Machine* or *Man a Machine* (La Mettrie, 1748) did explicitly argue that humans are automata.

Modern analytic philosophy has typically accepted materialism (often in the form of the brain-state **identity theory** (Place, 1956; Armstrong, 1968), which asserts that mental states are identical with brain states), but has been much more divided on functionalism, the machine analogy for the human mind, and the question of whether machines can literally think. A number of early philosophical responses to Turing's (1950) "Computing Machinery and Intelligence," for example, Scriven (1953), attempted to deny that it was even *meaningful* to say that machines could think, on the ground that such an assertion violated the meaning of the word. Scriven, at least, had retracted this view by 1963; see his addendum to a reprint of his article (Anderson, 1964). The computer scientist Edsger Dijkstra said that "The question of whether a computer can think is no more interesting than the question of whether a submarine can swim." Ford and Hayes (1995) argue that the Turing Test is not helpful for AI.

Functionalism is the philosophy of mind most nalurally suggested by AI, and critiques of functionalism often take the form of critiques of AI (as in the case of Searle). Following the classification used by Block (1980), we can distinguish varieties of functionalism. **Functional specification theory** (Lewis, 1966, 1980) is a variant of brain-state identity theory that selects the brain states that are to be identified with mental states on the basis of their functional role. **Functional state identity theory** (Putnam, 1960, 1967) is more closely based on a machine analogy. It identifies mental states not with *physical* brain states but with abstract computational states of the brain conceived expressly as a computing device. These abstract states are supposed to be independent of the specific physical composition of the brain, leading some to charge that functional state identity theory is a form of dualism!

Both the brain-state identity theory and the various forms of functionalism have come under attack from authors who claim that they do not account for the *qualia* or "what it's like" aspect of mental states (Nagel, 1974). Searle has focused instead on the alleged inability of functionalism to account for intentionality (Searle, 1980, 1984, 1992). Churchland and Churchland (1982) rebut both these types of criticism.

ELIMINATIVE MATERIALISM

Eliminative materialism (Rorty, 1965; Churchlancl, 1979) differs from all other prominent theories in the philosophy of mind, in that it does not attempt to give an account of why our "folk psychology" or commonsense ideas about the mind are true, but instead rejects them as false and attempts to replace them with a purely scientific theory of the mind. In principle, this scientific theory could be given by classical AI, but in practice, eliminative matcrialists tend to lean on neuroscience and neural network research instead (Churchland,

1986), on the grounds that classical AI, especially "knowledge representation" research of the kind described in Chapter 10, tends to rely on the truth of folk psychology. Although the "intentional stance" viewpoint (Dennett, 1971) could be interpreted as functionalist, it should probably instead be regarded as a form of eliminative materialism, in that taking the "intentional stance" is not supposed to reflect any objective property of the agent toward whom the stance is taken. It should also be noted that it is possible to be an eliminative materialist about some aspects of mentality while analyzing others in some other way. For instance, Dennett (1978) is much more strongly eliminativist about qualia than about intentionality.

Sources for the main critics of weak AI were given in the chapter. Although it became fashionable in the post-neural-network era to deride symbolic approaches, not all philosophers are critical of GOFAI. Some are, in fact, ardent advocates and even practitioners. Zenon Pylyshyn (1984) has argued that cognition can best be understood through a computational model, not only in principle but also as a way of conducting research at present, and has specifically rebutted Dreyfus's criticisms of the computational model of human cognition (Pylyshyn, 1974). Gilbert Harman (1983), in analyzing belief revision, makes connections with AI research on truth maintenance systems. Michael Bratman has applied his "belief-desire-intention" model of human psychology (Bratman, 1987) to AI research on planning (Bratman, 1992). At the extreme end of strong AI, Aaron Sloman (1978, p. xiii) has even described as "racialist" Joseph Weizenbaum's view (Weizenbaum, 1976) that hypothetical intelligent machines should not be regarded as persons.

The philosophical literature on minds, brains, and related topics is large and sometimes difficult to read without proper training in the terminology and methods of argument employed. The *Encyclopedia of Philosophy* (Edwards, 1967) is an impressively authoritative and very useful aide in this process. *The Cambridge Dictionary of Philosophy* (Audi, 1999) is a shorter and more accessible work, but main entries (such as "philosophy of mind) still span 10 pages or more. The *MIT Encyclopedia of Cognitive Science* (Wilson and Keil, 1999) covers the philosophy of mind as well as the biology and psychology of mind. General collections of articles on philosophy of mind, including functionalism and other viewpoints related to AI, are *Materialism and the Mind-Body Problem* (Rosenthal, 1971) and *Readings in the Philosophy of Psychology*, volume 1 (Block, 1980). Biro and Shahan (1982) present a collection devoted to the pros and cons of functionalism. Anthologies of articles dealing more specifically with the relation between philosophy and AI include *Minds and Machines* (Anderson, 1964), *Philosophical Perspectives in Artificial Intelligence* (Ringle, 1979), *Mind Design* (Haugeland, 1981), and *The Philosophy of Artificial Intelligence* (Boden, 1990). There are several general introductions to the philosophical "AI question" (Boden, 1977, 1990; Haugeland, 1985; Copeland, 1993). *The Behavioral and Brain Sciences*, abbreviated *BBS*, is a major journal devoted to philosophical and scientific debates about AI and neuroscience. Topics of ethics and responsibility in AI are covered in journals such as *AI and Society*, *Law, Computers and Artificial Intelligence*, and *Artijicial Intelligence and Law*.

EXERCISES

26.1 Go through Turing's list of alleged "disabilities" of machines, identifying which have been achieved, which are achievable in principle by a program, and which are still problematic because they require conscious mental states.

26.2 Does a refutation of the Chinese room argument necessarily prove that appropriately programmed computers have mental states? Does an acceptance of the argument necessarily mean that computers cannot have mental states?

26.3 In the brain prosthesis argument, it is important to be able to restore the subject's brain to normal, such that its external behavior is as it would have been if the operation had not taken place. Can the skeptic reasonably object that this would require updating those neurophysiological properties of the neurons relating to conscious experience, as distinct from those involved in the functional behavior of the neurons?

26.4 Find and analyze an account in the popular media of one or more of the arguments to the effect that AI is impossible.

26.5 Attempt to write definitions of the terms "intelligence," "thinking," and "consciousness." Suggest some possible objections to your definitions.

26.6 Analyze the potential threats from AI technology to society. What threats are most serious, and how might they be combated? How do they compare to the potential benefits?

26.7 How do the potential threats from AI technology compare with those from other computer science technologies, and to bio-, nano-, and nuclear technologies?

26.8 Some critics object that AI is impossible, while others object that it is *too* possible, and that ultraintelligent machines pose a threat. Which of these objections do you think is more likely? Would it be a contradiction for someone to hold both positions?

27 AI: PRESENT AND FUTURE

In which we take stock of where we are and where we are going, this being a good thing to do before continuing.

In Part I, we proposed a unified view of AI as rational agent design. We showed that the design problem depends on the percepts and actions available to the agent, the goals that the agent's behavior should satisfy, and the nature of the environment. A variety of different agent designs are possible, ranging from reflex agents to fully deliberative, knowledge-based agents. Moreover, the components of these designs can have a number of different instantiations — for example, logical, probabilistic, or "neural." The intervening chapters presented the principles by which these components operate.

For all the agent designs and components, there has been tremendous progress both in our scientific understanding and in our technological capabilities. In this chapter, we stand back from the details and ask, *"Will all this progress lead to a general-purpose intelligent agent that can perform well in a wide variety of environments?* Section 27.1 looks at the components of an intelligent agent to assess what's known and what's missing. Section 27.2 does the same for the overall agent architecture. Section 27.3 asks whether "rational agent design" is the right goal in the first place. (The answer is, "Not really, but it's OK for now.") Finally, Section 27.4 examines the consequences of success in our endeavors.

27.1 AGENT COMPONENTS

Chapter 2 presented several agent designs and their components. To focus our discussion here, we will look at the utility-based agent, which we show again in Figure 27.1. This is the most general of our agent designs; we will also consider its extension with learning capabilities, as depicted in Figure 2.15.

Interaction with the environment through sensors and actuators: For much of the history of AI, this has been a glaring weak point. With a few honorable exceptions, AI systems were built in such a way that humans had to supply the inputs and interpret the outputs, while robotic systems focused on low-level tasks in which high-level reasoning and planning were largely absent. This was due in part to the great expense and engineering effort required

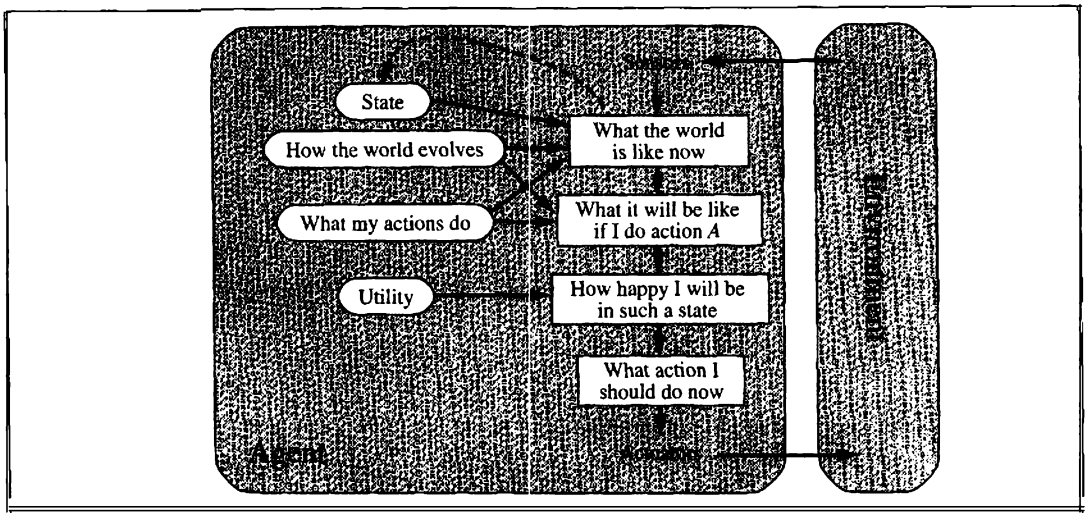


Figure 27.1 A model-based, utility-based agent, as first presented in Figure 2.14.

to get real robots to work at all. The situation has changed rapidly in recent years with the availability of ready-made programmable robots, such as the four-legged robots shown in Figure 25.4(b). These, in turn, have benefited from small, cheap, high-resolution CCD cameras and compact, reliable motor drives. MEMS (micro-electromechanical systems) technology has supplied miniaturized accelerometers and gyroscopes and is now producing actuators that will, for example, power an artificial flying insect. (It may also be possible to combine millions of MEMS actuators to produce very powerful macroscopic actuators.) For physical environments, then, AI systems no longer have a real excuse. Furthermore, an entirely new environment—the Internet—has become available.

Keeping track of the state of the world: This is one of the core capabilities required for an intelligent agent. It requires both perception and updating of internal representations. Chapter 7 described methods for keeping track of worlds described by propositional logic; Chapter 10 extended this to first-order logic; and Chapter 15 described filtering algorithms for tracking uncertain environments. These filtering tools are required when real (and therefore imperfect) perception is involved. Current filtering and perception algorithms can be combined to do a reasonable job of reporting low-level predicates such as "the cup is on the table" but we have some way to go before they can report that "Dr. Russell is having a cup of tea with Dr. Norvig." Another problem is that, although approximate filtering algorithms can handle quite large environments, they are still essentially propositional—like propositional logic, they do not represent objects and relations explicitly. Chapter 14 explained how probability and first-order logic can be combined to solve this problem; we expect that the application of these ideas for tracking complex environments will yield huge benefits. Incidentally, as soon as we start talking about objects in an uncertain environment, we encounter identity uncertainty—we don't know which object is which. This problem has been largely ignored in logic-based AI, where it has generally been assumed that percepts incorporate constant symbols that identify the objects.

IDENTITY UNCERTAINTY

Projecting, evaluating, and selecting future courses of action: The basic knowledge representation requirements here are the same as for keeping track of the world; the primary difficulty is coping with courses of action—such as having a conversation or a cup of tea—that consist eventually of thousands or millions of primitive steps for a real agent. It is only by imposing **hierarchical structure** on behavior that we humans cope at all. Some of the planning algorithms in Chapter 12 use hierarchical representations and first-order representations to handle problems of this scale; on the other hand, the algorithms given in Chapter 17 for decision making under uncertainty are essentially using the same ideas as the state-based search algorithms of Chapter 3. There is clearly a great deal of work to do here, perhaps along the lines of recent developments in **hierarchical reinforcement learning.**

Utility as an expression of preferences: In principle, basing rational decisions on the maximization of expected utility is completely general and avoids many of the problems of purely goal-based approaches, such as conflicting goals and uncertain attainment. As yet, however, there has been very little work on constructing *realistic* utility functions—imagine, for example, the complex web of interacting preferences that must be understood by an agent operating as an office assistant for a human being. It has proven very difficult to decompose preferences over complex states in the same way that Bayes nets decompose beliefs over complex states. One reason may be that preferences over states are really *compiled* from preferences over state histories, which are described by **reward functions** (see Chapter 17). Even if the reward function is simple, the corresponding utility function may be very complex. This suggests that we take seriously the task of knowledge engineering for reward functions as a way of conveying to our agents what it is that we want them to do.

Learning: Chapters 18 to 20 described how learning in an agent can be formulated as inductive learning (supervised, unsupervised, or reinforcement-based) of the functions that constitute the various components of the agent. Very powerful logical and statistical techniques have been developed that can cope with quite large problems, often reaching or exceeding human capabilities in the identification of predictive patterns defined on a given vocabulary. On the other hand, machine learning has made very little progress on the important problem of constructing new representations at levels of abstraction higher than the input vocabulary. For example, how can an autonomous robot generate useful predicates such as *Office* and Cafe if they are not supplied to it by humans? Similar considerations apply to learning behavior—HavingACupOfTea is an important high-level action, but how does it get into an action library that initially contains much simpler actions such as *RaiseArm* and Swallow? Unless we understand such issues, we are faced with the daunting task of constructing large commonsense knowledge bases by hand.

27.2 AGENT ARCHITECTURES

It is natural to ask, "Which of the agent architectures in Chapter 2 should an agent use?" The answer is, "All of them!" We have seen that reflex responses are needed for situations in which time is of the essence, whereas knowledge-based deliberation allows the agent to

HYBRID ARCHITECTURE

plan ahead. A complete agent must be able to do both, using a **hybrid architecture.** One important property of hybrid architectures is that the boundaries between different decision components are not fixed. For example, **compilation** continually converts declarative information at the deliberative level into more efficient representations, eventually reaching the reflex level—see Figure 27.2. (This is the purpose of explanation-based learning, as discussed in Chapter 19.) Agent architectures such as SOAR (Laird *et* al., 1987) and THEO (Mitchell, 1990) have exactly this structure. Every time they solve a problem by explicit deliberation, they save away a generalized version of the solution for use by the reflex component. A less studied problem is the *reversal* of this process: when the environment changes, learned reflexes may no longer be appropriate and the agent must return to the deliberative level to produce new behaviors.

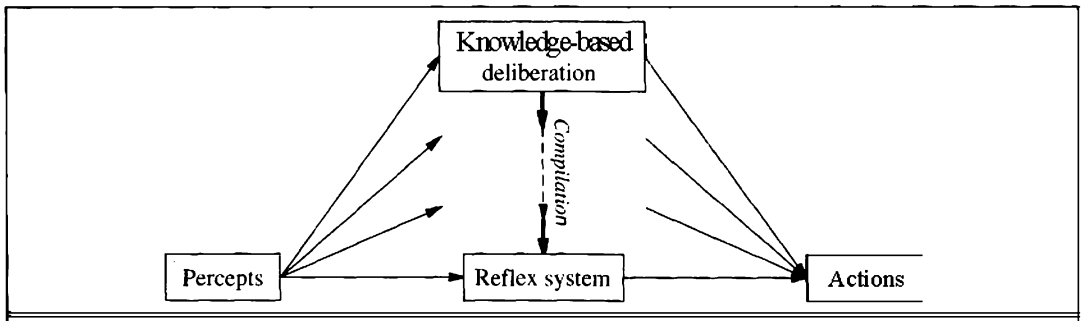


Figure 27.2 Compilation serves to convert deliberative decision making into more efficient, reflexive mechanisms.

Agents also need ways to control their own deliberations. They must be able to cease deliberating when action is demanded, and they must be able to use the time available for deliberation to execute the most profitable computations. For example, a taxi-driving agent that sees an accident ahead must decide in a split second either to brake or to take evasive action. It should also spend that split second thinking about the most important questions, such as whether the lanes to the left and right are clear and whether there is a large truck close behind, rather than worrying about wear and tear on the tires or where to pick up the next passenger. These issues are usually studied under the heading of **real-time** AI. As AI systems move into more complex domains, all problems will become real-time, because the agent will never have long enough to solve the decision problem exactly.

REAL-TIME AI

Clearly, there is a pressing need for methods that work in more general decision-making situations. Two promising techniques have emerged in recent years. The first involves the use of **anytime algorithms** (Dean and Boddy, 1988; Horvitz, 1'987). An anytime algorithm is an algorithm whose output quality improves gradually over time, so that it has a reasonable decision ready whenever it is interrupted. Such algorithms are controlled by a metalevel decision procedure that assesses whether further computation is worthwhile. Iterative deepening search in game playing provides a simple example of an anytime algorithm. More complex systems, composed of many such algorithms working together, can also be constructed (Zilberstein and Russell, 1996). The second technique is **decision-theoretic metareasoning**

ANYTIME ALGORITHMS

DECISION-THEORETIC METAREASONING

(Horvitz, 1989; Russell and Wefald, 1991; Horvitz and Breese, 1996). This method applies the theory of information value (Chapter 16) to the selection of computations. The value of a computation depends on both its cost (in terms of delaying action) and its benefits (in terms of improved decision quality). Metareasoning techniques can be used to design better search algorithms and to guarantee that the algorithms have the anytime property. Metareasoning is expensive, of course, and compilation methods can be applied so that the overhead is small compared to the costs of the computations being controlled.

REFLECTIVE ARCHITECTURE

Metareasoning is but one aspect of a general reflective architecture — that is, an architecture that enables deliberation about the computational entities and actions occurring within the architecture itself. A theoretical foundation for reflective architectures can be built by defining a joint state space composed from the environment state and the computational state of the agent itself. Decision-making and learning algorithms can be designed that operate over this joint state space and thereby serve to implement and improve the agent's computational activities. Eventually, we expect task-specific algorithms such as alpha-beta search and backward chaining to disappear from AI systems, to be replaced by general methods that direct the agent's computations toward the efficient generation of high-quality decisions.

27.3 Are We Going in the Right Direction?

The preceding section listed many advances and many opportunities for further progress. But where is this all leading? Dreyfus (1992) gives the analogy of trying to get to the moon by climbing a tree; one can report steady progress, all the way to the top of the tree. In this section, we consider whether AI's current path is more like a tree climb or a rocket trip.

In Chapter 1, we said that our goal was to build agents that *act rationally*. However, we also said that

> . . . achieving perfect rationality — alwaysdoing the right thing — is not feasible in complicated environments. The computational demands are just too high. For most of the book, however, we will adopt the working hypothesis that perfect rationality is a good starting point for analysis.

Now it is time to consider again what exactly the goal of AI is. We want to build agents, but with what specification in mind? Here are four possibilities:

PERFECT RATIONALITY

Perfect rationality. A perfectly rational agent acts at every instant in such a way as to maximize its expected utility, given the information it has acquired from the environment. We have seen that the calculations necessary to achieve perfect rationality in most environments are too time-consuming, so perfect rationality is not a realistic goal.

CALCULATIVE RATIONALITY

Calculative rationality. This is the notion of rationality that we have used implicitly in designing logical and decision-theoretic agents. A calculatively rational agent *eventually* returns what *would have been* the rational choice at the beginning of its deliberation. This is an interesting property for a system to exhibit, but in most environments, the right answer at the wrong time is of no value. In practice, AI system designers are forced to compromise on decision quality to obtain reasonable overall performance; unfortunately, the theoretical basis

of calculative rationality does not provide a well-founded way to make such compromises.

BOUNDED RATIONALITY

Bounded rationality. Herbert Simon (1957) rejected the notion of perfect (or even approximately perfect) rationality and replaced it with bounded rationality, a descriptive theory of decision making by real agents. He wrote,

> The capacity of the human mind for formulating and solving complex problems is very small compared with the size of the problems whose solution is required for objectively rational behavior in the real world—or even for a reasonable approximation to such objective rationality.

He suggested that bounded rationality works primarily by satisficing — that is, deliberating only long enough to come up with an answer that is "good enough." Simon won the Nobel prize in economics for this work and has written about it in depth (Simon, 1982). It appears to be a useful model of human behaviors in many cases. It is not a formal specification for intelligent agents, however, because the definition of "good enough" is not given by the theory. Furthermore, satisficing seems to be just one of a large range of methods used to cope with bounded resources.

BOUNDED OPTIMALITY

Bounded optimality (BO). A bounded optimal agent behaves as well as possible, given *its computational resources*. That is, the expected utility of the agent program for a bounded optimal agent is at least as high as the expected utility of any other agent program running on the same machine.

Of these four possibilities, bounded optimality seems to offer the best hope for a strong theoretical foundation for AI. It has the advantage of being possible to achieve: there is always at least one best program — something that perfect rationality lacks. Bounded optimal agents are actually useful in the real world, whereas calculatively rational agents usually are not, and satisficing agents might or might not be, depending on their own whims.

The traditional approach in AI has been to start with calculative rationality and then make compromises to meet resource constraints. If the problems imposed by the constraints are minor, one would expect the final design to be similar to a BO agent design. But as the resource constraints become more critical—e.g., as the environment becomes more complex—one would expect the two designs to diverge. In the theory of bounded optimality, these constraints can be handled in a principled fashion.

As yet, little is known about bounded optimality. It is possible to construct bounded optimal programs for very simple machines and for somewhat restricted kinds of environments (Etzioni, 1989; Russell *et al.*, 1993), but as yet we have no idea what BO programs are like for large, general-purpose computers in complex environments. If there is to be a constructive theory of bounded optimality, we have to hope that the design of bounded optimal programs does not depend too strongly on the details of the computer being used. It would make scientific research very difficult if adding a few kilobytes of memory to a gigabyte machine made a significant difference to the design of the BO program. One way to make sure this cannot happen is to be slightly more relaxed about the criteria for bounded optimality. By analogy with the notion of asymptotic complexity (Appendix A), we can define asymptotic bounded optimality (ABO) as follows (Russell and Subramanian, 1995). Suppose a program $\mathbf{P}$ is bounded optimal for a machine M in a class of environments E,

ASYMPTOTIC BOUNDED OPTIMALITY

where the complexity of environments in E is unbounded. Then program **P'** is ABO for M in E if it can outperform **P** by running on a machine kM that is k times faster (or larger) than M. Unless k were enormous, we would be happy with a program that was ABO for a nontrivial environment on a nontrivial architecture. There would be little point in putting enormous effort into finding BO rather than ABO programs, because the size and speed of available machines tends to increase by a constant factor in a fixed amount of time anyway.

We can hazard a guess that BO or ABO programs for powerful computers in complex environments will not necessarily have a simple, elegant structure. We have already seen that general-purpose intelligence requires some reflex capability and some deliberative capability, a variety of forms of knowledge and decision making, learning and compilation mechanisms for all of those forms, methods for controlling reasoning, and a large store of domain-specific knowledge. A bounded optimal agent must adapt to the environment in which it finds itself, so that eventually its internal organization will reflect optimizations that are specific to the particular environment. This is only to be expected, and it is similar to the way in which racing cars restricted by engine capacity have evolved into extremely complex designs. We suspect that a science of artificial intelligence based on bounded optimality will involve a good deal of study of the processes that allow an agent program to converge to bounded optimality and perhaps less concentration on the details of the messy programs that result.

In sum, the concept of bounded optimality is proposed as a formal task for AI research that is both well defined and feasible. Bounded optimality specifies optimal *programs* rather than optimal *actions*. Actions are, after all, generated by programs, and it is over programs that designers have control.

27.4 What if AI Does Succeed?

In David Lodge's *Small World* (1984), a novel about the academic world of literary criticism, the protagonist causes consternation by asking a panel of eminent but contradictory literary theorists the following question: *"What if you* were *right?"* None of the theorists seems to have considered this question before, perhaps because debating unfalsifiable theories is an end in itself. Similar confusion can sometimes be evoked by asking AI researchers, "What if you succeed?" AI is fascinating, and intelligent computers are clearly more useful than unintelligent computers, so why worry?

As Section 26.3 relates, there are ethical issues to consider. Intelligent computers are more powerful, but will that power be used for good or ill? Those who strive to develop AI have a responsibility to see that the impact of their work is a positive one. The scope of the impact will depend on the degree of success of AI. Even modest successes in AI have already changed the ways in which computer science is taught (Stein, 2002) and software development is practiced. AI has made possible new applications such as speech recognition systems, inventory control systems, surveillance systems, robots, and search engines.

We can expect that medium-level successes in AI would affect all kinds of people in their daily lives. So far, computerized communication networks, such as cell phones and the

Internet, have had this kind of pervasive effect on society, but AI has not. We can imagine that truly useful personal assistants for the office or the home would have a large positive impact on people's lives, although they might cause some economic dislocation in the short term. A technological capability at this level might also be applied to the development of autonomous weapons, which many view as an undesirable development.

Finally, it seems likely that a large-scale success in AI—the creation of human-level intelligence and beyond—would change the lives of a majority of humankind. The very nature of our work and play would be altered, as would our view of intelligence, consciousness, and the future destiny of the human race. At this level, AI systems could pose a more direct threat to human autonomy, freedom, and even survival. For these reasons, we cannot divorce AI research from its ethical consequences.

Which way will the future go? Science fiction authors seem to favor dystopian futures over utopian ones, probably because they make for more interesting plots. But so far, AI seems to fit in with other revolutionary technologies (printing, plumbing, air travel, telephony) whose negative repercussions are outweighed by their positive aspects.

In conclusion, we see that AI has made great progress in its short history, but the final sentence of Alan Turing's essay on *Computing Machinery and Intelligence* is still valid today:

> We can see only a short distance ahead, but we can see that much remains to be done.