Artificial Intelligence
A Modern Approach

Shale Craig
`sakcraig@uwaterloo.ca`

April 28, 2013

Contents

Chapter 1

Introduction

We've tried to understand how we think (perceive, under-
stand, predict & manipulate). AI goes further and also
tries to build these intelligent entities.

1.1 What is AI?

AI can be split into four sought after qualities in comput-
ers:

- Thinking Humanly [Subsection 1.1.2]

- Thinking Rationally [Subsection 1.1.3]

- Acting Humanly [Subsection 1.1.1]

- Acting Rationally [Subsection 1.1.4]

1.1.1 Acting Humanly (The Turing Test Approach)

The Turing Test provides an operational definition of intelligence. If a human interrogator cannot tell if the responses are from human or computer. For now, a computer that passes needs the following capabilities:

1. NLP - to communicate in English.

2. Knowledge Representation - to store what it knows and hears.

3. Automated Reasoning - reason on stored information to new conclusions.

4. Machine learning to be adaptive and detect and extrapolate patterns.

Turing avoids physical interaction because it is unrelated to intelligence. A similar Total Turing Test uses a video input as well. Passing the Total Turing Test requires these extra skills:

5. Computer Vision - to perceive objects.

6. Robotics - to manipulate objects.

These six disciplines compose most of AI.

1.1.2 Thinking Humanly (The Cognitive Modeling Approach)

To write a program that thinks like a human, we need to understand human minds. We can understand through

introspection, psychological experiments, and brain imaging. Once we have a theory, we can implement it as a computer program. Cognitive Science brings together Computer Models (AI) and experimental psychology to understand how the human mind operates. It is important to note that an algorithm behaving accurately doesn't imply it uses a good model.

1.1.3 Thinking Rationally (The "Laws of Thought" Approach)

Generally, rational thought is synonymous with logical thought. We can create programs to compute logical correctness, but these programs have bad problems (decidability being one of them). It's hard to turn informal knowledge and state it in logical notation. There is a difference between solving problems "in principle" and in practice.

1.1.4 Acting Rationally (The Rational Agent Approach)

Computer agents are expected to autonomously perceive their environment, persist over time, create and set goals. Rational agents act to achieve the best outcome. In the "laws of thought" approach, emphasis is on inference, but that's only part of being a rational agent. Acting rationally doesn't always involve inference (i.e. reflexively recoiling from a hot stove), and sometimes acting rationally occurs because something just "must be done". Learning is not

just for knowledge, but it also allows us to more effectively behave rationally. This approach has advantages over the other approaches. It is more general than the "laws of thought" approach because it's more than just correct inference. Scientific development helps this approach more than the other developments. Sometimes we can't always react 100% rationally, as that is too computationally expensive. We assume that perfection is a good starting point for our analysis.

1.2 Foundations Of Artificial Intelligence

This is a brief history of ideas that contribute to AI. It glosses over many parts.

1.2.1 Philosophy

Reasoning is like numerical computation, we can use formal rules to form conclusions.

If the mind is governed by physical laws, then it has no more free will than a rock "deciding" to fall toward the center of the earth. Free will is the perception of choices appears to the choosing entity. Nothing can be trusted from senses, because they can be lies. General rules are acquired by exposure to repeated associations between their elements. Logical positivism states that all knowledge is logical theories of connected observation sentences.

How does knowledge lead to action?
Aristotle defines that actions are justified by expected outcome and goals. He defines an iterative algorithm that builds dependencies to achieve a goal.