Lecture 1 CS6800 Artificial Intelligence:

- Introduction and potential Applications
- Problem Solving
- The 12 coin problem
- Production Systems
- The 8-puzzle
- Problem Representation
- Basic Procedure
- Control
- Representation schemes
- Traveling Salesman Problem
- Backwards & Bi-directional Prod. Systems

Introduction

What is *Artificial Intelligence*?

Why do we study it?

What is the scope of this class?

Applications

- Natural Language Processing
- Expert Systems
- Robotics
- Automatic Programming
- Scheduling Problems
- Perception

Problem Solving

Consider the problem of finding a counterfeit coin among 11 other good coins. You must determine the identity of the bad coin and whether it is lighter or heavier than its 11 brethren in no more than three weighings.

Notes from Berkely CS188

What is AI?

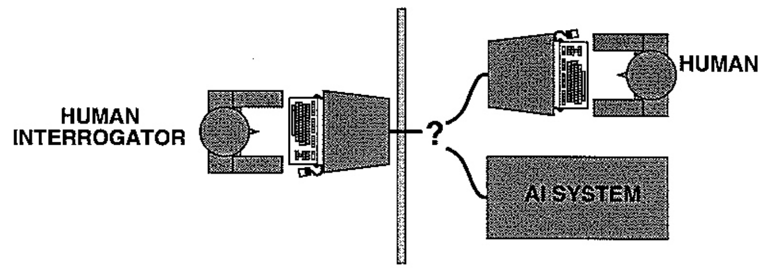
Systems that think like humans	Systems that think rationally
Systems that act like humans	Systems that act rationally

Acting humanly: The Turing test

Turing (1950) "Computing machinery and intelligence":

◇ "Can machines think?" $\longrightarrow$ "Can machines behave intelligently?"

◇ Operational test for intelligent behavior: the Imitation Game



◇ Predicted that by 2000, a machine might have a 30% chance of fooling a lay person for 5 minutes

◇ Anticipated all major arguments against AI in following 50 years

◇ Suggested major components of AI: knowledge, reasoning, language understanding, learning

Problem: Turing test is not reproducible, constructive, or amenable to mathematical analysis

Thinking humanly: Cognitive Science

1960s "cognitive revolution": information-processing psychology replaced prevailing orthodoxy of behaviorism

Requires scientific theories of internal activities of the brain

- What level of abstraction? "Knowledge" or "circuits"?
- How to validate? Requires
  1) Predicting and testing behavior of human subjects (top-down)
  or 2) Direct identification from neurological data (bottom-up)

Both approaches (roughly, Cognitive Science and Cognitive Neuroscience) are now distinct from AI

Both share with AI the following characteristic:
the available theories do not explain (or engender) anything resembling human-level general intelligence

Hence, all three fields share one principal direction!

Thinking rationally: Laws of Thought

Normative (or prescriptive) rather than descriptive

Aristotle: what are correct arguments/thought processes?

Several Greek schools developed various forms of logic:
notation and rules of derivation for thoughts;
may or may not have proceeded to the idea of mechanization

Direct line through mathematics and philosophy to modern AI

Problems:
1) Not all intelligent behavior is mediated by logical deliberation
2) What is the purpose of thinking? What thoughts should I have?

Acting rationally

Rational behavior: doing the right thing

The right thing: that which is expected to maximize goal achievement, given the available information

Doesn't necessarily involve thinking—e.g., blinking reflex—but thinking should be in the service of rational action

Aristotle (Nicomachean Ethics):

> *Every art and every inquiry, and similarly every action and pursuit, is thought to aim at some good*

Rational agents

An agent is an entity that perceives and acts

This course is about designing rational agents

Abstractly, an agent is a function from percept histories to actions:

$$f : \mathcal{P}^* \rightarrow \mathcal{A}$$

For any given class of environments and tasks, we seek the agent (or class of agents) with the best performance

Caveat: *computational limitations make perfect rationality unachievable*
$\rightarrow$ design best program for given machine resources

AI prehistory

Philosophy	logic, methods of reasoning mind as physical system foundations of learning, language, rationality
Mathematics	formal representation and proof algorithms, computation, (un)decidability, (in)tractability probability
Psychology	adaptation phenomena of perception and motor control experimental techniques (psychophysics, etc.)
Economics	formal theory of rational decisions
Linguistics	knowledge representation grammar
Neuroscience	plastic physical substrate for mental activity
Control theory	homeostatic systems, stability simple optimal agent designs

Potted history of AI

1943	McCulloch & Pitts: Boolean circuit model of brain
1950	Turing's "Computing Machinery and Intelligence"
1952–69	Look, Ma, no hands!
1950s	Early AI programs, including Samuel's checkers program, Newell & Simon's Logic Theorist, Gelernter's Geometry Engine
1956	Dartmouth meeting: "Artificial Intelligence" adopted
1965	Robinson's complete algorithm for logical reasoning
1966–74	AI discovers computational complexity Neural network research almost disappears
1969–79	Early development of knowledge-based systems
1980–88	Expert systems industry booms
1988–93	Expert systems industry busts: "AI Winter"
1985–95	Neural networks return to popularity
1988–	Resurgence of probability; general increase in technical depth "Nouvelle AI": ALife, GAs, soft computing
1995–	Agents agents everywhere . . .

State of the art

Which of the following can be done at present?

- ♢ Play a decent game of table tennis
- ♢ Drive along a curving mountain road
- ♢ Drive in the center of Cairo
- ♢ Buy a week's worth of groceries at Berkeley Bowl
- ♢ Buy a week's worth of groceries on the web
- ♢ Play a decent game of bridge
- ♢ Discover and prove a new mathematical theorem
- ♢ Write an intentionally funny story
- ♢ Give competent legal advice in a specialized area of law
- ♢ Translate spoken English into spoken Swedish in real time
- ♢ Perform a complex surgical operation

Unintentionally funny stories

One day Joe Bear was hungry. He asked his friend Irving Bird where some honey was. Irving told him there was a beehive in the oak tree. Joe threatened to hit Irving if he didn't tell him where some honey was.

Henry Squirrel was thirsty. He walked over to the river bank where his good friend Bill Bird was sitting. Henry slipped and fell in the river. Gravity drowned.

Once upon a time there was a dishonest fox and a vain crow. One day the crow was sitting in his tree, holding a piece of cheese in his mouth. He noticed that he was holding the piece of cheese. He became hungry, and swallowed the cheese. The fox walked over to the crow. The end.

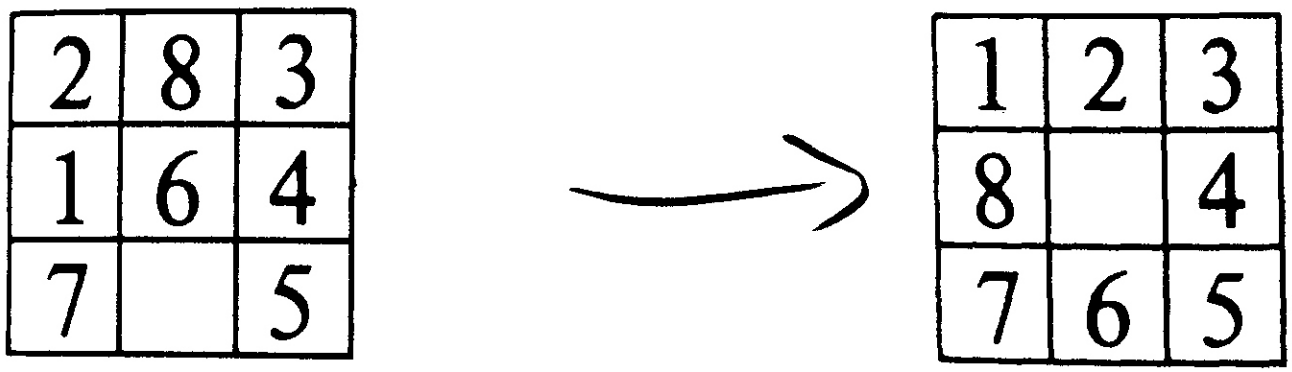
Unintentionally funny stories

Joe Bear was hungry. He asked Irving Bird where some honey was. Irving refused to tell him, so Joe offered to bring him a worm if he'd tell him where some honey was. Irving agreed. But Joe didn't know where any worms were, so he asked Irving, who refused to say. So Joe offered to bring him a worm if he'd tell him where a worm was. Irving agreed. But Joe didn't know where any worms were, so he asked Irving, who refused to say. So Joe offered to bring him a worm if he'd tell him where a worm was ...

Production Systems

Production systems serve as a convenient means of describing the structure and operation of AI systems. A production system is composed of a *global database,* which is manipulated by means of certain well-defined *operations,* all under the control of some global *control strategy*.

An example: The 8-puzzle



Problem Representation

To solve the *representation problem* in AI we must specify the global database, the rules, and control strategy.

Selecting a good problem representation is one of the most difficult parts in applying AI problem solving techniques in a given situation.

In the eight-puzzle, each tile configuration is a *problem state*. The set of all possible configurations is the *space* of problem states or the *problem space*.

The Basic Solution Procedure

Procedure **PRODUCTION**

DATA ← initial database

until *DATA* satisfies the stop condition **do:**

begin

select some rule *R*, in the set of rules applicable to *DATA*

DATA ← result of applying *R* to *DATA*

end

Control

Selecting rules and keeping track of those sequences of rules already tried and the databases they produced constitute what we call the *control strategy* for production systems.

Control does not have perfect information, therefore, there must be a search process for the correct rule to use in a given situation.

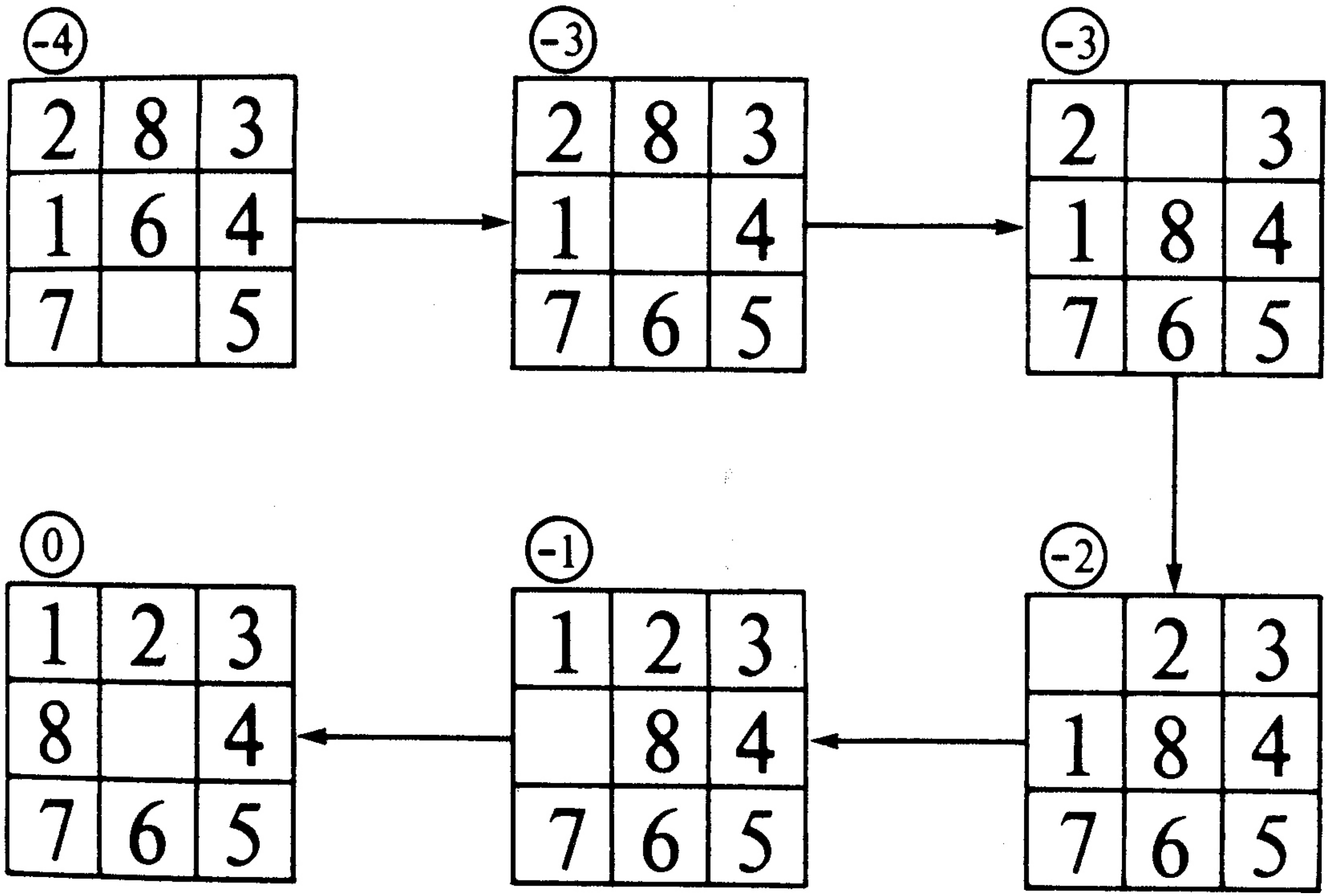
Classes of control strategies:

- Irrevocable
- Tentative
  - backtracking
  - graph-search control

Irrevocable Control

- Is it appropriate?
- Local verses Global information
- Hill-climbing i.e. steepest descent
- How to apply to the 8-puzzle?
- Problems with this method

Simple Application to the 8-puzzle



Backtracking

Basic idea:

1.) Try a rule

2.) If no solution, go back to the state at 1, and try another rule

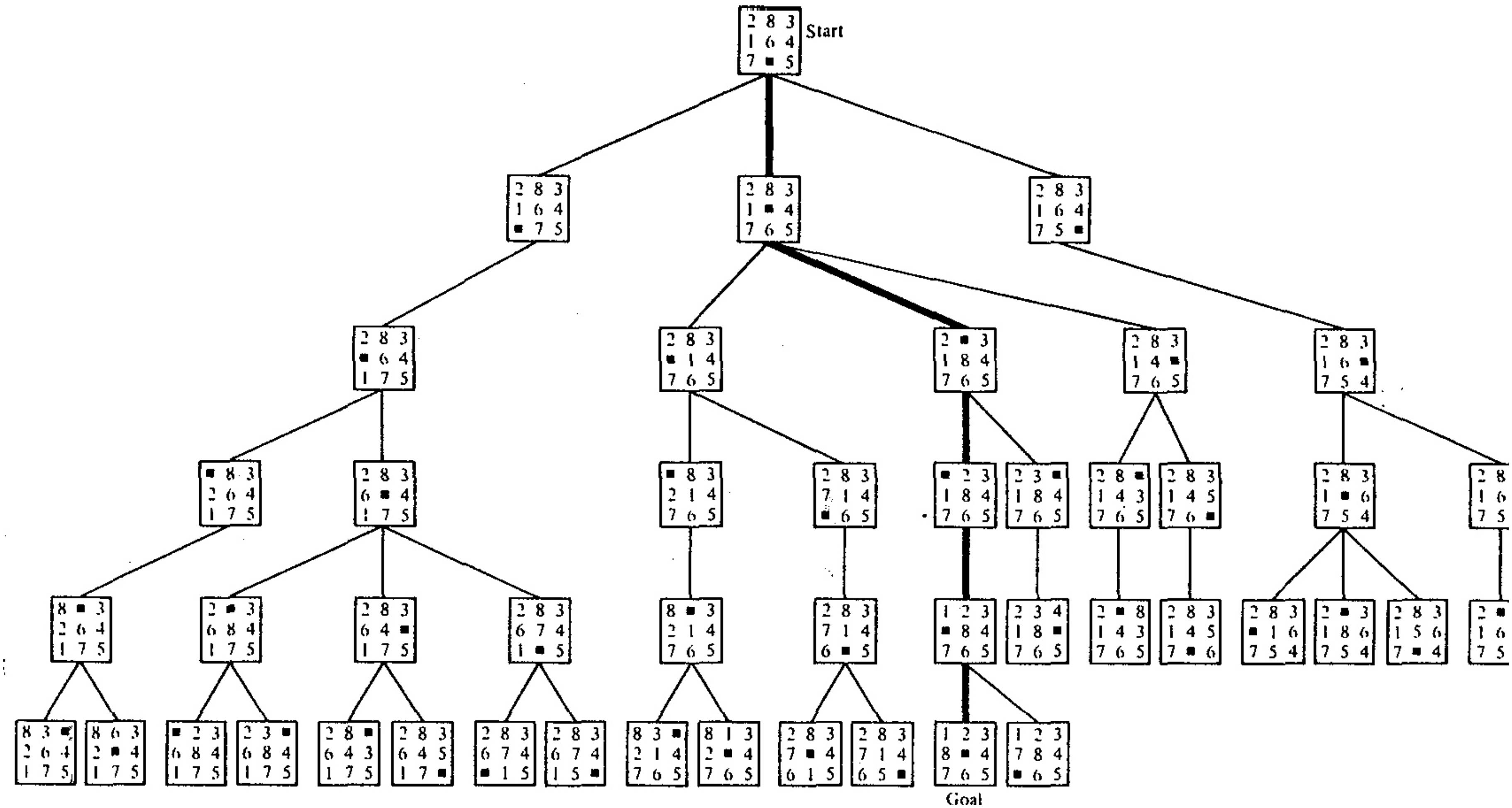
- Backtracking can avoid local minima
- Backtracking allows alternative paths to be explored.
- If rule selection is guided by information then the backtracking can be more efficient.

Graph Search

Graphs and trees are very useful data structures for keeping track of the effects of the application of alternative rules.

In the 8-puzzle example, we can keep track of alternative rules applied by using a structure called a search tree. The ***root*** of the tree is the initial configuration; the various rules that apply to this configuration are the ***directed arcs*** to ***descendent*** nodes. We can grow the tree until a leaf node satisfies the ***termination condition***.

Graph-search Control and the 8-Puzzle



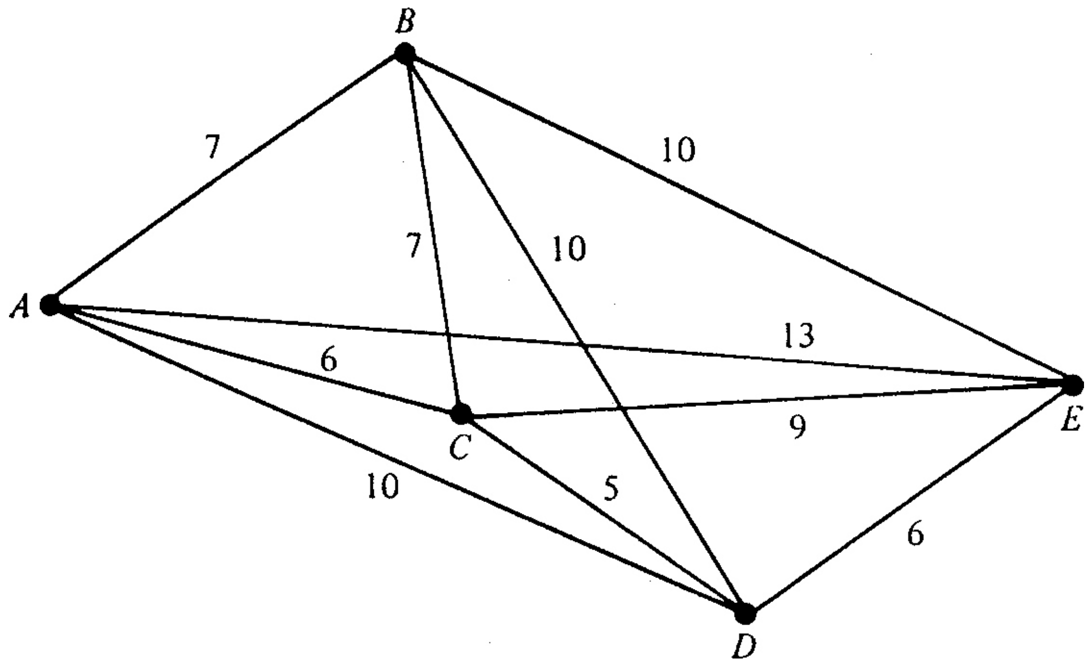
Representation

So far we have talked about representation in terms of the overall problem. Often efficient problem solution requires a good representation for the problem states, and goal conditions. Obviously one prefers representations that have a small number of states. This should simplify the effort in searching the space.

In general, there is no known way to effectively improve a given representation incrementally. Often what is required is human intuition that we haven't yet been able to automate.

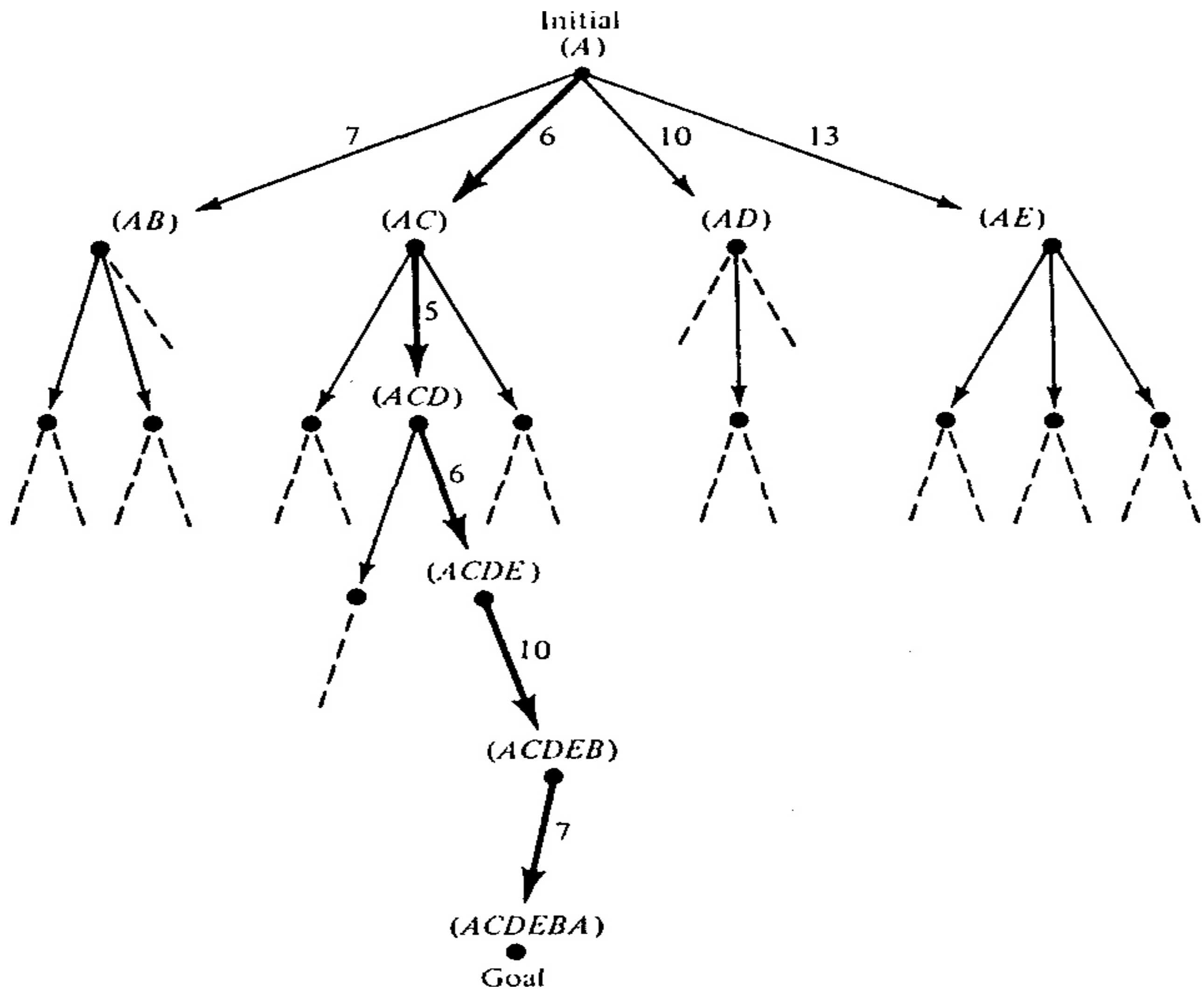
Some more example problem representations

A Traveling salesman problem



Search Tree for the Traveling Salesman Problem

Search Tree for the Traveling Salesman Problem



Other types of Production Systems

Backwards production systems

- Start at goal nodes, and search for initial state(s).

Bi-directional production systems

- Start at both initial states, and goal states, and search towards each other.