University of Malta
BSc(Hons.) IT Year II

Problem Solving Techniques in Artificial Intelligence

Lecture 9.2

Reasoning

-
- Expert Systems
- Backward Chaining
- Forward Chaining

Department of Computer Science and AI
1998/9

Expert Systems
see Bratko chapter 14

- An *expert system* is a program that behaves like an expert in some, usually narrow domain of applicatin, e.g diagnosis, data interpretation.

- The solution of problems in such domains requires expert knowledge to be represented in some explicit form, For this reason, , so expert systems are often referred to as "knowledge-based systems" (KBSs).

- An expert system should be capable of *explaining* its behaviour to the user by identifying the knowledge employed in reaching a given conclusion.

- An expert system should be capable of dealing with *inexactitude*, both in terms of incomplete or unreliable information about the problem, and in terms of imperfect inference rules: some drug may cause some problem, but usually does not.

Expert Systems: Main Components

- Knowledge Base specific to the domain of application.
- Inference Engine which shows how to use knowledge in the domain.
- User Interface
  - Knowledge Acquisition
  - Problem Specification
  - Problem Solving

Separation of knowledge from algorithms that use it.

Knowledge in the form of if-then rules.

If-then Rules

- Such rules are conditional statements, but have various interpretations, e.g.
  - If precondition P then conclusion C
  - If situation S then action A
  - If conditions C1 and C2 hold then condition C does not hold
  - If condition A then conclusion B with certainty F
- Desirable Characteristics
  - Modularity
  - Incrementability
  - Modififiability
  - Transparency

Reasoning Procedures: Forward and Backward Chaining

Once knowledge is represented, we need a reasoning procedure to draw conclusions.

- **Backward Chaining**. Start with a hypothesis, then look in database for confirmation in the form of
  - Matching *facts*
  - Matching *rules* whose conclusion matches the hypothesis. In this case, for each rule, create new hypotheses for the conditions to the rule.
- **Forward Chaining**
  - Start with facts.
  - Find the rules whose conditions are satisfied by the facts.
  - Add facts to the database corresponding to the conclusions of the rules.

Forward versus Backward Chaining

- **Backward Chaining**
  - Is conservative, reducing goals to subgoals only to the extent needed to obtain a solution.
  - Does not modify the database
  - Is used to model reasoning
- **Forward Chaining**
  - Does not terminate until all ramifications have been computed
  - Does change the database
  - Is used as an automatic database routine rather than as a reasoning mechanism.