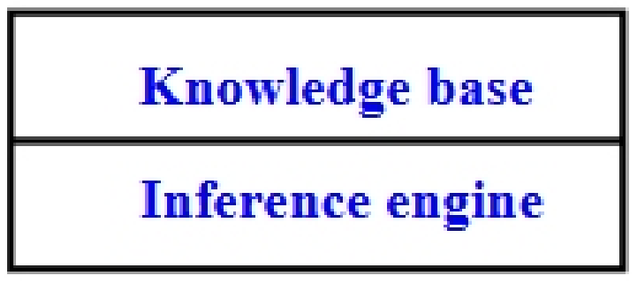
CS 2710 Foundations of AI
Lecture 13

Logic reasoning systems, Situation calculus

Milos Hauskrecht
milos@cs.pitt.edu
5329 Sennott Square

Knowledge-based system



- **Knowledge base:**
  - A set of sentences that describe the world in some formal (representational) language (e.g. first-order logic)
  - Domain specific knowledge
- **Inference engine:**
  - A set of procedures that work upon the representational language and can infer new facts or answer KB queries (e.g. resolution algorithm, forward chaining)
  - Domain independent

Automated reasoning systems

Examples and main differences:

- **Theorem provers**
  - Prove sentences in the first-order logic. Use inference rules, resolution rule and resolution refutation.
- **Deductive retrieval systems**
  - Systems based on rules (KBs in Horn form)
  - Prove theorems or infer new assertions (forward, backward chaining)
- **Production systems** 
  - Systems based on rules with actions in antecedents
  - Forward chaining mode of operation
- **Semantic networks** 
  - Graphical representation of the world, objects are nodes in the graphs, relations are various links

Production systems

Based on rules, but different from KBs in the Horn form

Knowledge base is divided into:

- **A Rule base (includes rules)**
- **A Working memory (includes facts)**

A special type of if – then rule

$$p_1 \wedge p_2 \wedge \ldots p_n \Rightarrow a_1, a_2, \ldots, a_k$$

- **Antecedent:** a conjunction of literals
  - facts, statements in predicate logic
- **Consequent:** a conjunction of actions. An action can:
  - **ADD** the fact to the KB (working memory)
  - **REMOVE** the fact from the KB (consistent with logic ?)
  - **QUERY** the user, etc …

Production systems

Based on rules, but different from KBs in the Horn form

Knowledge base is divided into:

- **A Rule base (includes rules)**
- **A Working memory (includes facts)**

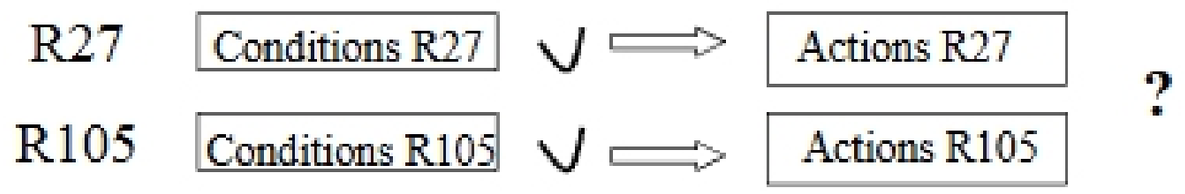
A special type of if – then rule

$$p_1 \wedge p_2 \wedge \ldots p_n \Rightarrow a_1, a_2, \ldots, a_k$$

- **Antecedent:** a conjunction of literals
  - facts, statements in predicate logic
- **Consequent:** a conjunction of actions. An action can:
  - **ADD** the fact to the KB (working memory)
  - **REMOVE** the fact from the KB ← !!! Different from logic
  - **QUERY** the user, etc …

Production systems

- Use **forward chaining to do reasoning**:
  - If the antecedent of the rule is satisfied (rule is said to be "active") then its consequent can be executed (it is "fired")
- **Problem:** Two or more rules are active at the same time. Which one to execute next?



- Strategy for selecting the rule to be fired from among possible candidates is called **conflict resolution**