Knowledge Representation: Logic II

Manuela Veloso

CMU 15-381, Fall 2001

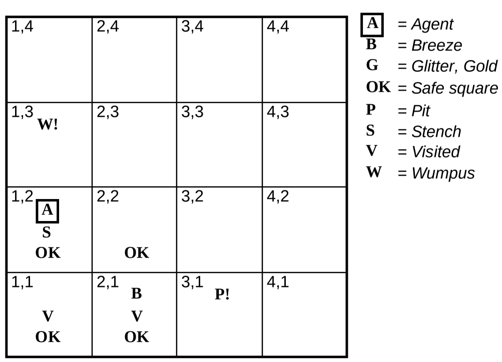
- The need for "knowledge" representation
- Symbolic knowledge
- Propositional logic and inference
- Predicate logic
- The resolution inference procedure

Acting in the World

State and Actions

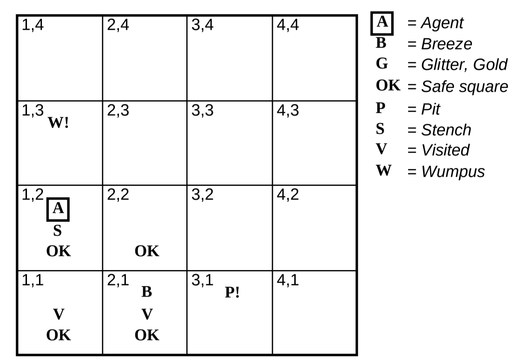
- Complete intelligent agent
  - ▷ Perception - at which *state* am I?
  - ▷ Cognition - which *action* should I take?
  - ▷ Action - how do I *execute* the action?
- State recognition implies state *representation*
- Many examples
- Symbolic knowledge about the world

State - Logic Representation



$$\neg S_{1,1} \wedge \neg S_{2,1} \wedge S_{1,2} \wedge \neg B_{1,1} \wedge \neg B_{1,2} \wedge B_{2,1}$$

Rules



$$\begin{aligned}
R_1 &: \neg S_{1,1} \Rightarrow \neg W_{1,1} \wedge \neg W_{1,2} \wedge \neg W_{2,1} \\
R_2 &: \neg S_{2,1} \Rightarrow \neg W_{1,1} \wedge \neg W_{2,1} \wedge \neg W_{2,2} \wedge \neg W_{3,1} \\
R_3 &: \neg S_{1,2} \Rightarrow \neg W_{1,1} \wedge \neg W_{1,2} \wedge \neg W_{2,2} \wedge \neg W_{1,3} \\
R_4 &: S_{1,2} \Rightarrow W_{1,3} \vee W_{1,2} \vee W_{2,2} \vee W_{1,1}
\end{aligned}$$

Inference Rules for Propositional Logic

- Modus Ponens

$$\frac{\alpha \Rightarrow \beta, \alpha}{\beta}$$

- And-Elimination

$$\frac{\alpha_1 \wedge \ldots \wedge \alpha_n}{\alpha_i}$$

- And-Introduction

$$\frac{\alpha_1, \ldots, \alpha_n}{\alpha_1 \wedge \ldots \wedge \alpha_n}$$

- Or-Introduction

$$\frac{\alpha_i}{\alpha_1 \vee \ldots \vee \alpha_n}$$

Inference Rules for Propositional Logic

- Double-Negation Elimination

$$\frac{\neg\neg\alpha}{\alpha}$$

- Unit Resolution

$$\frac{\alpha \vee \beta, \neg\beta}{\alpha}$$

- Resolution

$$\frac{\alpha \vee \beta, \neg\beta \vee \gamma}{\alpha \vee \gamma}$$

Apply Inference Rules

- Conclude $W_{1,3}$

Logical Inference

- Not brute-force generation of the complete truth tables.
- A **proof theory** provides a set of inference rules.
- KB $\vdash$ S ; S is infered from KB by the proof theory.
- **Soundness**:
  KB $\mid=$ S whenever KB $\vdash$ S
  "what is infered is also true in the real world"
- **Completeness**:
  KB $\vdash$ S whenever KB $\mid=$ S
  "everything that is true in the real world can be infered by the proof theory"

Modus Ponens

For $d = a_i$,

$$\frac{\begin{array}{r} a_1 \wedge \ldots \wedge a_m \Rightarrow b \\ c_1 \wedge \ldots \wedge c_n \Rightarrow d \end{array}}{a_1 \wedge \ldots \wedge a_{i-1} \wedge a_{i+1} \wedge \ldots \wedge a_m \wedge c_1 \wedge \ldots \wedge c_n \Rightarrow b}$$

- Modus ponens is sound
  - Proof: By contradiction.
  - Consider model where premises hold but the conclusion does not. In that case $a_1 \wedge \ldots \wedge a_{i-1} \wedge a_{i+1} \wedge \ldots \wedge a_m \wedge c_1 \wedge \ldots \wedge c_n$ holds but not b.
  - From second premise, then a_i holds.

Modus Ponens

- Then from first premise b also holds as all the a's hold.
- Contradiction.
- Every model in which the premises of modus ponens hold is also a model in which the conclusion holds.

Modus Ponens

For $d = a_i$,

$$\frac{\begin{array}{r} a_1 \wedge \ldots \wedge a_m \Rightarrow b \\ c_1 \wedge \ldots \wedge c_n \Rightarrow d \end{array}}{a_1 \wedge \ldots \wedge a_{i-1} \wedge a_{i+1} \wedge \ldots \wedge a_m \wedge c_1 \wedge \ldots \wedge c_n \Rightarrow b}$$

- Modus ponens is complete only for Horn databases.
  - A Horn clause is one of the form $a_1 \wedge \ldots \wedge a_m \Rightarrow b$, which is equivalent to saying that it has at most one positive atom.
  - Example: incomplete if disjunctions:
  - All classes meet on TTh or MWF. AI meets at 10:30am. Pat has tennis lessons Th and F at 10:30am. Can Pat take AI?

Modus Ponens

1. TTh(AI,10:30) $\vee$ MWF(AI,10:30)
2. TTh(AI,10:30) $\wedge$ busy(Pat,Th,10:30) $\Rightarrow$ conflict(Pat,AI)
3. MWF(AI,10:30) $\wedge$ busy(Pat,F,10:30) $\Rightarrow$ conflict(Pat,AI)
4. busy(Pat,Th,10:30)
5. busy(Pat,F, 10:30)

- Modus ponens removes busy from the premises, but cannot handle the disjunction.

Predicate Logic – First-order Logic

- The symbols a, b, c, ... are constants.
- The symbols ?a, ?b, ?c, ... are variables.
- Constants are terms.
- Variables are terms.
- (function application) If $\alpha_1 \ldots \alpha_n$ are terms and f is an n-ary function, then $(f\ \alpha_1 \ldots \alpha_n)$ is a term.
- (predicate application) If $\alpha_1 \ldots \alpha_n$ are terms and p is an n-ary predicate, then $(p\ \alpha_1 \ldots \alpha_n)$ is a formula.
- (boolean operators) If $\alpha_1 \ldots \alpha_n$ are formulae then (and $\alpha_1 \ldots \alpha_n$), (or $\alpha_1 \ldots \alpha_n$), (not α_1), and ($\leftarrow \alpha_1 \alpha_2$) are also formulae.

Predicate Logic – First-order Logic

- (quantification) If α is a formula and ?x is a variable, then (forall ?x α) and (exists ?x α) are formulae. α is called the *scope* of ?x, and all occurrences of ?x in α are *bound*.
- (wffs) Occurrences of a variable which are not bound are *unbound*. A *well-formed formula (wff)* is a formula which contains no unbound variables.

Resolution Inference Rule

For $d_j = a_i$ (unify with some substitution),

... in implicative form:

$$\begin{array}{rcl} a_1 \wedge \ldots \wedge \boxed{a_i} \wedge \ldots \wedge a_m & \Rightarrow & b_1 \vee \ldots \vee b_k \\ c_1 \wedge \ldots \wedge c_n & \Rightarrow & d_1 \vee \ldots \vee \boxed{d_j} \vee \ldots \vee d_l \\ \hline \multicolumn{3}{r}{a_1 \wedge \ldots \wedge a_{i-1} \wedge a_{i+1} \wedge \ldots \wedge a_m \wedge c_1 \wedge \ldots \wedge c_n \Rightarrow} \\ \multicolumn{3}{r}{b_1 \vee \ldots \vee b_k \vee d_1 \vee \ldots \vee d_{j-1} \vee d_{j+1} \vee \ldots d_l} \end{array}$$

Resolution Inference Rule

For $d_j = a_i$ (unify with some substitution),

... in disjunctive form:

$$\begin{array}{rcl} \neg a_1 \vee \ldots \vee \boxed{\neg a_i} \vee \ldots \vee \neg a_m & \vee & b_1 \vee \ldots \vee b_k \\ \neg c_1 \vee \ldots \vee \neg c_n & \vee & d_1 \vee \ldots \vee \boxed{d_j} \vee \ldots \vee d_l \\ \hline \multicolumn{3}{l}{\neg a_1 \vee \ldots \vee \neg a_{i-1} \vee \neg a_{i+1} \vee \ldots \vee \neg a_m \vee \neg c_1 \vee \ldots \vee \neg c_n} \\ \multicolumn{3}{r}{b_1 \vee \ldots \vee b_k \vee d_1 \vee \ldots \vee d_{j-1} \vee d_{j+1} \vee \ldots} \end{array}$$

Applying the Resolution Rule

- Refutation: To prove α, assume α is false and prove that there is a contradiction in the KB , i.e. prove that *false* can be derived from KB.

$$(KB \wedge \neg\alpha \vdash \mathit{false}) \Leftrightarrow (KB \vdash \alpha)$$

- Apply the resolution rule (*search*!) until *false* is derived.

Example

- Resolve 3. and 5. resulting
  7. MWF(AI,10:30) $\Rightarrow$ conflict(Pat,AI)
- Resolve 2. and 4. resulting
  8. TTh(AI,10:30) $\Rightarrow$ conflict(Pat,AI)
- Resolve 1. and 7. resulting
  9. True $\Rightarrow$ conflict(Pat,AI) $\vee$ TTh(AI,10:30)
- Resolve 8. and 9. resulting
  10. True $\Rightarrow$ conflict(Pat,AI)

Example

1. True $\Rightarrow$ TTh(AI,10:30) $\vee$ MWF(AI,10:30)
2. TTh(AI,10:30) $\wedge$ busy(Pat,Th,10:30) $\Rightarrow$ conflict(Pat,AI)
3. MWF(AI,10:30) $\wedge$ busy(Pat,F,10:30) $\Rightarrow$ conflict(Pat,AI)
4. True $\Rightarrow$ busy(Pat,Th,10:30)
5. True $\Rightarrow$ busy(Pat,F, 10:30)
6. True $\Rightarrow$ $\neg$ conflict(Pat,AI)

Convertion to Normal Form

To perform $\vdash$, i.e., inference using the resolution inference rule, we need to show that any FOL sentence can be put into implicative normal form, or conjunctive normal form.

1. Eliminate implications.
2. Reduce scope of negation.
3. Standardize variables - each quantifier binds a unique variable.
4. Skolemize - Eliminate existential quantifiers – Introduce a Skolem function for each existentially quantified variable.

Convertion to Normal Form

5. Convert to prenex form - Move quantifiers to the left – prefix.
6. Drop the prefix.
7. Distribute `and`, $\wedge$ and `or`, $\vee$, to get a conjunction of disjunctions.
8. Split conjunction - create a separate clause to each conjunct.
9. Standardize variables apart again.

Resolution Example

- Resolve 2. and 4.:
  mortal(x) matches mortal(z)

  results 5. $\neg$man$(x) \vee \neg$immortal(x)
- Resolve 5. and 3.:
  man(x) matches man(Socrates)
  results 6. $\neg$immortal$(Socrates)$
- Resolve 1. and 6: derive *false*.

Resolution Example

Prove $\neg$immortal(Socrates) from the KB:

$\forall x \quad$ man$(x) \Rightarrow$ mortal(x)

man(Socrates)

$\forall x \quad \neg$mortal$(x) \vee \neg$immortal(x)

Convertion to normal form: KB + negated fact:

1. immortal(Socrates)
2. $\neg$man$(x) \vee$ mortal(x)
3. man(Socrates)
4. $\neg$mortal$(z) \vee \neg$immortal(z)

FOL – All you have to know

- Knowledge base (KB): well formed formulas (clauses).
- We want to find if a fact is *entailed* by the KB, i.e. if it holds in any model given that the facts in the KB hold.
- *Derive* facts from KB using *rules of inference*.
- *Sound* inference rules derive facts that are entailed by the KB.
- *Modus ponens* is an inference rule.
- Modus ponens is sound for FOL.
- If a fact is entailed by the KB, then a *complete* inference rule(s) will also derive it.

FOL – All you have to know

- Modus ponens is not complete for FOL.
- Modus ponens is complete for KBs of *Horn clauses*. Horn clauses are clauses with only disjunctions and only one positive term.
- *Resolution* is an inference rule. Resolution is a generalization of modus ponens.
- Resolution is sound and complete for FOL.
- There is a resolution procedure.
- Gödel's incompleteness theorem: If FOL is extended with constructs to handle mathematical induction, then there is no complete inference procedure.