# Propositional Logic: Resolution Proofs

CPSC 322 Lecture 19

February 24, 2006
Textbook §4.2

# Proofs

- A proof is a mechanically derivable demonstration that a formula logically follows from a knowledge base.
- Given a proof procedure, $KB \vdash g$ means $g$ can be derived from knowledge base $KB$.
- Recall $KB \models g$ means $g$ is true in all models of $KB$.
- A proof procedure is sound if $KB \vdash g$ implies $KB \models g$.
- A proof procedure is complete if $KB \models g$ implies $KB \vdash g$.

# Bottom-up proof procedure

$KB \vdash g$ if $g \subseteq C$ at the end of this procedure:

$C := \{\}$;
**repeat**
    **select** clause "$h \leftarrow b_1 \wedge \ldots \wedge b_m$" in $KB$ such that
        $b_i \in C$ for all $i$, and $h \notin C$;
    $C := C \cup \{h\}$
**until** no more clauses can be selected.

# Properties of bottom-up proof procedure

- Soundness: If $KB \vdash g$ then $KB \models g$.
- Fixed Point: further applications of our rule of derivation will not change $C$.
- Minimal model: Let $I$ be the interpretation in which every element of the fixed point is true and every other atom is false. $I$ is a model of $KB$.
- Completeness: If $KB \models g$ then $KB \vdash g$.

# Top-down Ground Proof Procedure

Idea: search backward from a query to determine if it is a logical consequence of $KB$.
An answer clause is of the form:

$$yes \leftarrow a_1 \wedge a_2 \wedge \ldots \wedge a_m$$

The SLD Resolution of this answer clause on atom $a_i$ with the clause:

$$a_i \leftarrow b_1 \wedge \ldots \wedge b_p$$

is the answer clause

$$yes \leftarrow a_1 \wedge \cdots \wedge a_{i-1} \wedge b_1 \wedge \cdots \wedge b_p \wedge a_{i+1} \wedge \cdots \wedge a_m.$$

# Derivations

- An answer is an answer clause with $m = 0$. That is, it is the answer clause $yes \leftarrow$ .
- A derivation of query "$?q_1 \wedge \ldots \wedge q_k$" from $KB$ is a sequence of answer clauses $\gamma_0, \gamma_1, \ldots, \gamma_n$ such that
  - $\gamma_0$ is the answer clause $yes \leftarrow q_1 \wedge \ldots \wedge q_k$,
  - $\gamma_i$ is obtained by resolving $\gamma_{i-1}$ with a clause in $KB$, and
  - $\gamma_n$ is an answer.

# Top-down definite clause interpreter

To solve the query $?q_1 \wedge \ldots \wedge q_k$:

> $ac :=$ "$yes \leftarrow q_1 \wedge \ldots \wedge q_k$"
> **repeat**
> > **select** atom $a_i$ from the body of $ac$;
> > **choose** clause $C$ from $KB$ with $a_i$ as head;
> > replace $a_i$ in the body of $ac$ by the body of $C$
>
> **until** $ac$ is an answer.

# Nondeterministic Choice

- Don't-care nondeterminism If one selection doesn't lead to a solution, there is no point trying other alternatives. select
- Don't-know nondeterminism If one choice doesn't lead to a solution, other choices may. choose

# Example: successful derivation

$$
\begin{array}{lll}
a \leftarrow b \wedge c. & a \leftarrow e \wedge f. & b \leftarrow f \wedge k. \\
c \leftarrow e. & d \leftarrow k. & e. \\
f \leftarrow j \wedge e. & f \leftarrow c. & j \leftarrow c.
\end{array}
$$

Query: $?a$

$$
\begin{array}{llll}
\gamma_0: & yes \leftarrow a & \gamma_4: & yes \leftarrow e \\
\gamma_1: & yes \leftarrow e \wedge f & \gamma_5: & yes \leftarrow \\
\gamma_2: & yes \leftarrow f & & \\
\gamma_3: & yes \leftarrow c & &
\end{array}
$$

# Example: failing derivation

$$
\begin{array}{lll}
a \leftarrow b \wedge c. & a \leftarrow e \wedge f. & b \leftarrow f \wedge k. \\
c \leftarrow e. & d \leftarrow k. & e. \\
f \leftarrow j \wedge e. & f \leftarrow c. & j \leftarrow c.
\end{array}
$$

Query: $?a$

$$
\begin{array}{llll}
\gamma_0 : & yes \leftarrow a & \gamma_4 : & yes \leftarrow e \wedge k \wedge c \\
\gamma_1 : & yes \leftarrow b \wedge c & \gamma_5 : & yes \leftarrow k \wedge c \\
\gamma_2 : & yes \leftarrow f \wedge k \wedge c & & \\
\gamma_3 : & yes \leftarrow c \wedge k \wedge c & &
\end{array}
$$