Recap
Hunt The Wumpus

---

1. **Stench**
2. **Breeze**
3. **Gold**
4. **PIT**
Hunt The Wumpus
Partially Observable

Don’t know everything about the state of the world

Have to represent uncertainty in our knowledge

Have to explore
Propositional Logic

• Programming language for knowledge
• Factored representation of state
  • Propositions
  • Connectives
You perceive a breeze at \([1,1]\) if and only if there is a pit at \([2,1]\) or there is a pit at \([2,2]\)

\[ \begin{align*}
B_{1,1} : & \text{You perceive a breeze at } [1,1] \\
P_{1,2} : & \text{There is a pit at } [1,2] \\
P_{2,1} : & \text{There is a pit at } [2,1] 
\end{align*} \]

\[ B_{1,1} \leftrightarrow (P_{1,2} \lor P_{2,1}) \]
Model (Possible World)

• Assignment of true or false to all the propositional variables

• A model satisfies a sentence if it makes the sentence true

• “A model of the sentence”
## Satisfaction

<table>
<thead>
<tr>
<th>$B_{1,1}$</th>
<th>$P_{1,2}$</th>
<th>$P_{2,1}$</th>
<th>$P_{1,2} \lor P_{2,1}$</th>
<th>$B_{1,1} \leftrightarrow (P_{1,2} \lor P_{2,1})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>true</td>
<td>true</td>
<td>true</td>
<td>true</td>
<td>true</td>
</tr>
<tr>
<td>false</td>
<td>true</td>
<td>true</td>
<td>true</td>
<td>false</td>
</tr>
<tr>
<td>true</td>
<td>false</td>
<td>true</td>
<td>true</td>
<td>true</td>
</tr>
<tr>
<td>false</td>
<td>false</td>
<td>true</td>
<td>true</td>
<td>false</td>
</tr>
<tr>
<td>true</td>
<td>true</td>
<td>false</td>
<td>true</td>
<td>true</td>
</tr>
<tr>
<td>false</td>
<td>true</td>
<td>false</td>
<td>true</td>
<td>false</td>
</tr>
<tr>
<td>true</td>
<td>false</td>
<td>false</td>
<td>false</td>
<td>false</td>
</tr>
<tr>
<td>false</td>
<td>false</td>
<td>false</td>
<td>false</td>
<td>true</td>
</tr>
</tbody>
</table>
Entailment

• If $\alpha$ entails $\beta$:
  • Whenever $\alpha$ is true, so is $\beta$
  • Every model of $\alpha$ is a model of $\beta$
  • $\text{Models}(\alpha) \subseteq \text{Models}(\beta)$
  • $\alpha$ is at least as strong an assertion as $\beta$
  • Rules out no fewer possible worlds
Propositional Inference

• We need to compute what follows from our knowledge (entailment)

• Searching for proofs is an alternative to enumerating models

  • May be faster in practice

  • Still co-NP-complete in general
Inference Rules

And-elimination

Double negation

DeMorgan's Laws

Modus Ponens

Definition of biconditional
Properties of Inference

Rules

• Soundness:
  \[ \text{if } \alpha \vdash \beta \text{ then } \alpha \models \beta \]

• Completeness:
  \[ \text{if } \alpha \models \beta \text{ then } \alpha \vdash \beta \]
Propositional Inference

- Resolution is a sound and complete inference rule (that works on clauses)
- Forward chaining derives conclusions from facts using rules
- Backward chaining reasons backwards from goal to facts using rules (and recursively establishing their premises)
- Horn clauses are natural and permit efficient inference
Hunt The Wumpus

<table>
<thead>
<tr>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stench</td>
<td>Breeze</td>
<td>PIT</td>
<td></td>
</tr>
<tr>
<td>Stench</td>
<td>Stench</td>
<td>Gold</td>
<td>Breeze</td>
</tr>
<tr>
<td>Stench</td>
<td>Breeze</td>
<td>PIT</td>
<td></td>
</tr>
<tr>
<td>START</td>
<td>Breeze</td>
<td>PIT</td>
<td>Breeze</td>
</tr>
</tbody>
</table>

1 2 3 4
• Rooms adjacent to pits will have breezes

• Socrates is a person
  All people are mortal

• Anybody’s grandmother is either their mother’s or their father’s mother
First-Order Logic

• Constants and that denote objects in the world
• Function symbols that denote a mapping from a tuple of objects to another object
• Predicate symbols that denote relations (sets of tuples of objects)
• Connectives
• Variables and Quantifiers
Interpretation

• Mapping from elements a sentence to elements of a possible world

• Specifies exactly which objects, relations, and functions are referred to by the constant, predicate, and function symbols
Interpretation

- Domain of objects $D$

- Mapping from constant symbols to objects: $I(c) \in D$

- Mapping from predicate symbols to relations: $I(p) \subseteq D^n$ for $p$ of arity $n$

- Mapping from function symbols to (total) functions: $I(f) : D^n \to D$ for $f$ of arity $n$
Extended Interpretation

• Adds mapping from variables to objects:

\[ I(v) \in D \]
Semantics of FOL

- Atomic sentences: Use interpretation
- Complex sentences: Use semantics of connectives
- $\forall a$: True in every extended interpretation
- $\exists a$: True in at least one extended interp.
Entailment

• If $\alpha$ entails $\beta$:
  • Whenever $\alpha$ is true, so is $\beta$
  • Every model of $\alpha$ is a model of $\beta$
  • $\text{Models}(\alpha) \subseteq \text{Models}(\beta)$
  • $\alpha$ is at least as strong an assertion as $\beta$
    • Rules out no fewer possible worlds
137,506,194,466 models with $\leq 6$ objects!
Computing Entailment

- Number of models (probably) unbounded
  - And anyway hard to evaluate truth in a model
- Can’t do model checking
- Look for inference rules, do theorem proving
First-Order Inference
• $\forall x \text{ King}(x) \Rightarrow \text{ Evil}(x)$

• $\text{King}(J)$
• $\forall x \text{ King}(x) \Rightarrow \text{ Evil}(x)$

• $\text{King}(J)$

• $\text{Evil}(J)$

“Modus Ponens”
Universal Instantiation

\[\forall x P(x)\]

\[P(a), P(b), P(c), P(f(a)), P(f(b)), P(g(a)), \ldots\]
Propositionalization

∀x King(x) ⇒ Evil(x)

King(R) ⇒ Evil(R)

King(J) ⇒ Evil(J)

King(Father(R)) ⇒ Evil(Father(R))

...
Propositionalization

• Convert FOL sentences to PL sentences and do PL inference on them

• Seems kind of indirect

• Technical problem enumerating all the ground terms (could be infinite; see book)
• $\forall x \text{ King}(x) \Rightarrow \text{ Evil}(x)$

• $\text{King}(J)$
• $\forall x \ King(x) \Rightarrow Evil(x)$

• $King(J)$
• King(J) ⇒ Evil(J)

• King(J)
• King(J) $\Rightarrow$ Evil(J)

• King(J)

• Evil(J)
• \( \forall x \text{ King}(x) \land \text{ Greedy}(x) \Rightarrow \text{ Evil}(x) \)

• \( \text{ King}(J) \)

• \( \text{ Greedy}(J), \text{ Greedy}(R) \)
\begin{itemize}
  \item $\forall x \ King(x) \land Greedy(x) \Rightarrow Evil(x)$
  \item King(J)
  \item Greedy(J), Greedy(R)
  \item Evil(J)
\end{itemize}
• \( \forall x \text{ King}(x) \land \text{ Greedy}(x) \implies \text{ Evil}(x) \)

• King(J)

• Greedy(J), Greedy(R)

• Evil(J)
• $\forall x \ King(x) \land Greedy(x) \Rightarrow Evil(x)$

• King(J)

• Greedy(J), Greedy(R)

• Evil(J)
• $\forall x \ King(x) \land Greedy(x) \Rightarrow Evil(x)$

• $King(J)$

• $Greedy(J), Greedy(R)$

• $Evil(J)$
Substitution

- Replacement (binding) of variables by terms
  - \{ x/J \}
  - \{ x/R, y/Father(j) \}
• \( \forall x \text{ King}(x) \land \text{ Greedy}(x) \Rightarrow \text{ Evil}(x) \)

• \( \text{ King}(J) \) \( \{ x/J \} \)

• \( \text{ Greedy}(J), \text{ Greedy}(R) \)

If there is some substitution that makes the premises true, then the conclusion with the same substitution is also true.
\begin{itemize}
\item $\forall x \text{ King}(x) \land \text{Greedy}(x) \Rightarrow \text{Evil}(x)$
\item $\text{King}(J)$
\item $\forall y \text{ Greedy}(y)$
\item \{ $x/J$, $y/J$ \}
\end{itemize}
Generalized Modus Ponens

\[ \text{if } (p_1 \land p_2 \land \ldots \land p_n) \Rightarrow q, \text{ then } \text{Subst}(\Theta, q) \]

are atomic sentences

\[ \text{is a substitution such that:} \]

\[ \text{Subst}(\Theta, p_i) = \text{Subst}(\Theta, p_i) \]
Lifted Inference Rule

• Inference rule lifted from ground (variable-free) propositional logic to first-order logic

• “The key advantage of lifted inference rules over propositionalization is that they make only those substitutions that are required to allow particular inferences to succeed”
Inference Rules

\[ \frac{\alpha \land \beta}{\alpha} \]

**And-elimination**

\[ \frac{\alpha \land \beta}{\beta} \]

**Modus Ponens**

\[ \frac{\alpha \land \beta}{\alpha \land \beta} \]

\[ \frac{\alpha \land \beta}{\alpha \land \beta} \]

\[ \frac{\alpha \land \beta}{\alpha \land \beta} \]

**Double negation**

\[ \frac{\neg \neg \alpha}{\alpha} \]

**DeMorgan’s Laws**

\[ \neg (\alpha \land \beta) \equiv \neg \alpha \lor \neg \beta \]

\[ \neg (\alpha \lor \beta) \equiv \neg \alpha \land \neg \beta \]

**Definition of biconditional**

\[ \alpha \equiv \beta \]

\[ (\alpha \Rightarrow \beta) \land (\beta \Rightarrow \alpha) \]

\[ (\alpha \land \beta) \land (\beta \land \alpha) \]
Unit Resolution

\[ l_1 \lor \ldots \lor l_k, m, \ldots \lor l_i \lor \ldots \lor l_{i-1} \lor l_{i+1} \lor \ldots \lor l_k \]

\[ l_i, \ldots, l_k \text{ are literals} \]
\[ l_i \text{ and } m \text{ are complementary} \]

Positive literal: \( P \)

Negative literal: \( \neg P \)

Complementary literals
Resolution

$l_1, \ldots, l_k, m_1, \ldots, m_n$ are literals

$l_i$ and $m_j$ are complementary

Technical note: Resulting clause must be factored to contain only one copy of each literal.
Resolution

- Sound
- Complete
- Can it be lifted to FOL?
CNF for FOL

• Every sentence of first-order logic can be converted into an inferentially equivalent sentence in conjunctive normal form (CNF)
CNF for FOL

- Eliminate implications
- Move negation inwards
- Standardize variables apart
- Skolemize existential variables
- Drop universal variables
- Distribute $\lor$ over $\land$
CNF for FOL

• Every sentence of first-order logic can be converted into an inferentially equivalent sentence in conjunctive normal form (CNF)

• Not logically equivalent

• Inferentially equivalent: (un)satisfiable iff original sentence is (un)satisfiable
Resolution

\[ l_1, \ldots, l_k, m_1, \ldots, m_n \text{ are literals} \]
\[ l_i \text{ and } m_j \text{ are complementary} \]

Technical note: Resulting clause must be factored to contain only one copy of each literal.
• $\forall x \text{ King}(x) \land \text{ Greedy}(x) \Rightarrow \text{ Evil}(x)$

• $\text{King}(J)$

• $\forall y \text{ Greedy}(y)$

{ $x/J, y/J$ }
Unification

Unify(p, q) = Θ where Subst(Θ, p) = Subst(Θ, q)
• Knows(John,x) and Knows(John,Jane)
  • \{ x/Jane \}
• Knows(John,x) and Knows(y, Bill)
  • \{x/Bill, y/John \}
• Knows(John,x) and Knows(y, Mother(y))
  • \{ y/John, x/Mother(John) \}
• Knows(John,x) and Knows(x,Elizabeth)
  • Fails
Unification

$\text{Unify}(p, q) = \emptyset$ where $\text{Subst}(\emptyset, p) = \text{Subst}(\emptyset, q)$

- Variables need to be standardized apart
- Occurs check
- Most general unifier: places the fewest restrictions on the variables
  - Is unique
- AIMA Fig 9.1
FOL Resolution

$\text{Subst}(\Theta, \text{ } l_1 \lor \cdots \lor l_i \lor \cdots \lor l_k \lor m_1 \lor \cdots \lor m_j \lor \cdots \lor m_n) \equiv \text{Unify}(l_i, \neg m_j)$
Proof by Resolution

- Convert KB to CNF
- Convert $\neg \alpha$ to CNF and add to KB
- Apply resolution rule to complementary clauses (with unification)
- Until you derive the empty clause
The law says that it is a crime for an American to sell weapons to hostile nations. The country Nono, an enemy of America, has some missiles, and all its missiles were sold to it by Colonel West, who is American.

To prove: West is a criminal
It is a crime for an American to sell weapons to hostile nations.
\[
\forall x, y, z. \text{American}(x) \land \text{Weapon}(y) \land \text{Sells}(x, y, z) \land \text{Hostile}(z) \Rightarrow \text{Criminal}(x)
\]

Nono ... has some missiles.
\[
\exists x. \text{Owns}(\text{Nono}, x) \land \text{Missile}(x)
\]

All Nono’s missiles were sold to it by Colonel West.
\[
\forall x. \text{Missile}(x) \land \text{Owns}(\text{Nono}, x) \Rightarrow \text{Sells}(\text{West}, x, \text{Nono})
\]

Missiles are weapons.
\[
\forall x. \text{Missile}(x) \Rightarrow \text{Weapon}(x)
\]

An enemy of America counts as “hostile.”
\[
\forall x. \text{Enemy}(x, \text{America}) \Rightarrow \text{Hostile}(x)
\]

West is an American.
\[
\text{American}(\text{West})
\]

Nono is an enemy of America.
\[
\text{Enemy}(\text{Nono, America})
\]
∀x, y, z American(x) ∧ Weapon(y) ∧ Sells(x, y, z) ∧ Hostile(z) ⇒ Criminal(x)

∃x Owns(Nono, x) ∧ Missile(x)

∀x Missile(x) ∧ Owns(Nono, x) ⇒ Sells(West, x, Nono)

∀x Missile(x) ⇒ Weapon(x)

∀x Enemy(x, America) ⇒ Hostile(x)

American(West)            Enemy(Nono, America)
Proof by Resolution

• Convert KB to CNF
• Convert \( \neg \alpha \) to CNF and add to KB
• Apply resolution rule to complementary clauses (with unification)
• Until you derive the empty clause
Convert to CNF

• Eliminate implications
• Move negation inwards
• Standardize variables apart
• Skolemize existential variables
• Drop universal variables
• Distribute $\lor$ over $\land$
Eliminate implications:
∀x, y, z (American(x) ∧ Weapon(y) ∧ Sells(x, y, z) ∧ Hostile(z) ⇒ Criminal(x))

Move negation inwards (DeMorgan’s Laws)
∀x, y, z [¬American(x) ∨ ¬Weapon(y) ∨ ¬Sells(x, y, z) ∨ ¬Hostile(z)] ∨ Criminal(x)

Standardize variables apart

Skolemize existential variables

Drop universal quantifiers

¬American(x) ∨ ¬Weapon(y) ∨ ¬Sells(x, y, z) ∨ ¬Hostile(z) ∨ Criminal(x)

Distribute ∨ over ∧
∀x, y, z American(x) ∧ Weapon(y) ∧ Sells(x, y, z) ∧ Hostile(z) → Criminal(x)

¬American(x) ∨ ¬Weapon(y) ∨ ¬Sells(x, y, z) ∨ ¬Hostile(z) ∨ Criminal(x)
∀ x, y, z \: American(x) \land Weapon(y) \land Sells(x, y, z) \land Hostile(z) \Rightarrow Criminal(x)

∀ x \: Missle(x) \Rightarrow Weapon(x)

∀ x \: Missle(x) \land Owns(Nono, x) \Rightarrow Sells(West, x, Nono)

∀ x \: Missle(x) \Rightarrow Missle(x)

∀ x \: Missle(x) \lor Owns(Nono, x) \lor Sells(West, x, Nono)

∀ x \: Missle(x) \lor Weapon(x)

∀ x \: Enemy(x, America) \Rightarrow Hostile(x)

¬ Enemy(x, America) \lor Hostile(x)

American(West) \land Enemy(Nono, America)

American(West) \land Enemy(Nono, America)
\neg American(x) \lor \neg Weapon(y) \lor \neg Sells(x, y, z) \lor \neg Hostile(z) \lor \neg Criminal(x) \\
\neg \text{Missile}(x) \lor \neg \text{Owns}(\text{Nono}, x) \lor \text{Sells}(\text{West}, x, \text{Nono}) \\
\neg \text{Missile}(x) \lor \text{Weapon}(x) \\
\neg \text{Enemy}(x, \text{America}) \lor \text{Hostile}(x) \\
\text{American(\text{West})} \\
\text{Enemy(\text{Nono}, \text{America})}
Proof by Resolution

• Convert KB to CNF

• Convert \( \neg \alpha \) to CNF and add to KB

• Apply resolution rule to complementary clauses (with unification)

• Until you derive the empty clause
“West is a criminal”

\[ \neg \text{Criminal} (\text{West}) \]

Negate and convert to CNF

\[ \neg \neg \text{Criminal} (\text{West}) \]
\neg \text{American}(x) \lor \neg \text{Weapon}(y) \lor \neg \text{Sells}(x, y, z) \lor \neg \text{Hostile}(z) \lor \text{Criminal}(x) \\
\text{Owns}(\text{Nono}, M_1), \text{Missile}(M_1) \\
\neg \text{Missile}(x) \lor \neg \text{Owns}(\text{Nono}, x) \lor \text{Sells}(\text{West}, x, \text{Nono}) \\
\neg \text{Missile}(x) \lor \text{Weapon}(x) \\
\neg \text{Enemy}(x, \text{America}) \lor \text{Hostile}(x) \\
\text{American}(\text{West}) \\
\text{Enemy}(\text{Nono}, \text{America}) \\
\neg \text{Criminal}(\text{West})
Proof by Resolution

- Convert KB to CNF
- Convert \( \neg \alpha \) to CNF and add to KB
- Apply resolution rule to complementary clauses (with unification)
- Until you derive the empty clause
¬American(x) ∨ ¬Weapon(y) ∨ ¬Sells(x, y, z) ∨ ¬Hostile(z) ∨ Criminal(x)

Owns(Nono, M_1), Missile(M_1)

¬Missile(x) ∨ ¬Owns(Nono, x) ∨ Sells(West, x, Nono)

¬Missile(x) ∨ Weapon(x)

¬Enemy(x, America) ∨ Hostile(x)

American(West)

Enemy(Nono, America)

¬Criminal(West)
\(\neg \text{American}(x) \lor \neg \text{Weapon}(y) \lor \neg \text{Sells}(x, y, z) \lor \neg \text{Hostile}(z) \lor \neg \text{Criminal}(x)\)

\(\neg \text{Owns}(\text{Nono}, M_1), \neg \text{Missile}(M_1)\)

\(\neg \text{Missile}(x) \lor \neg \text{Owns}(\text{Nono}, x) \lor \neg \text{Sells}(\text{West}, x, \text{Nono})\)

\(\neg \text{Missile}(x) \lor \text{Weapon}(x)\)

\(\neg \text{Enemy}(x, \text{America}) \lor \text{Hostile}(x)\)

\(\text{American}(\text{West})\)

\(\text{Enemy}(\text{Nono}, \text{America})\)

\(\neg \text{Criminal}(\text{West})\)

\(\{x/\text{West}\}\)
¬American(x) V ¬Weapon(y) V ¬Sells(x, y, z) V ¬Hostile(z) V Criminal(x) V ¬Criminal(West) 

{x/West}
\neg\text{American}(x) \lor \neg\text{Weapon}(y) \lor \neg\text{Sells}(x,y,z) \lor \neg\text{Hostile}(z) \lor \text{Criminal}(x)

\neg\text{Criminal}(\text{West})

\neg\text{American}(\text{West}) \lor \neg\text{Weapon}(y) \lor \neg\text{Sells}(\text{West},y,z) \lor \neg\text{Hostile}(z)
Resolution Proof

Since \[ KB \cup \{ \neg \text{Criminal}(\text{West}) \} \] is unsatisfiable,

\[ KB \models \text{Criminal}(\text{West}) \]
Forward Chaining

• Knowledge base of definite clauses

• Starting from known facts, trigger rules whose premises are satisfied

• Using substitution to match

• Add their conclusions to the KB

• Until the query is answered or no new facts can be generated
Backward Chaining

• Work backward from the goal, chaining through rules to find known facts that support the proof
  • Allow substitutions when matching facts and rules
• DFS => incomplete
• The basis of logic programming (Prolog)
FOL Inference

- Semantics of first-order sentences
- Entailment: same as PL
- Lifted inference rules
- Forward and backward chaining
- Resolution (first-order CNF, unification)
  - Proof by contradiction
Hunt The Wumpus

```
  START
  /   \
  |   |
  /   \
  |   |
  /   \
  |   |
  /   \
  |   |
  /   \
  |   |
```
For Next Time:
Review for Midterm
AIMA Chapters 1–9
(per syllabus)
Exam issues: email me ASAP