Introduction to Logic

Summary

Michael Genesereth
Computer Science Department
Stanford University
Propositional Logic (logical operators)

\[(p \Rightarrow q) \iff (\neg p \lor q)\]

Relational Logic (variables and quantifiers)

\[(\forall x. p(x)) \lor (\exists x. q(x,x))\]

Functional Logic (compound terms)

\[\forall x. (p(x) \iff p(f(x)) \land p(g(x)))\]
A sentence is *valid* if and only if *every* interpretation satisfies it.

A sentence is *contingent* if and only if *some* interpretation satisfies it and *some* interpretation falsifies it.

A sentence is *unsatisfiable* if and only if *no* interpretation satisfies it.
### Properties of Sentences

<table>
<thead>
<tr>
<th>Valid</th>
<th>A sentence is <em>satisfiable</em> if and only if it is either valid or contingent.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contingent</td>
<td>A sentence is <em>falsifiable</em> if and only if it is contingent or unsatisfiable.</td>
</tr>
<tr>
<td>Unsatisfiable</td>
<td></td>
</tr>
</tbody>
</table>
A sentence \( \phi \) is *logically equivalent* to a sentence \( \psi \) if and only if every truth assignment that satisfies \( \phi \) satisfies \( \psi \) and every truth assignment that satisfies \( \psi \) satisfies \( \phi \).

A sentence \( \phi \) is *consistent with* a sentence \( \psi \) if and only if there is a truth assignment that satisfies both \( \phi \) and \( \psi \).

A set of premises \( \Delta \) *logically entails* a conclusion \( \varphi \) (written as \( \Delta \vdash \varphi \)) if and only if every interpretation that satisfies the premises also satisfies the conclusion.
Propositional Metatheorems:
   Equivalence Theorem
   Deduction Theorem
   Unsatisfiability Theorem
   Consistency Theorem
   Substitution Theorem

These theorems also hold in Relational Logic but only for closed sentences (no free variables).
Determining Logical Entailment

\( \{p(a) \lor p(b), \forall x.(p(x) \Rightarrow q(x))\} \vdash \exists x.q(x)? \)

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th>p(a) \lor p(b)</th>
<th>\forall x.(p(x) \Rightarrow q(x))</th>
<th>\exists x.q(x)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>
\[ \{ \forall x.(p(x) \Rightarrow q(x)), \forall x.(q(x) \Rightarrow r(x)) \} \models \forall x.(p(x) \Rightarrow r(x)) \]

1. \( \forall x.(p(x) \Rightarrow q(x)) \)                Premise
2. \( \forall x.(q(x) \Rightarrow r(x)) \)                Premise
3. \( p([c]) \Rightarrow q([c]) \)                UE: 1
4. \( q([c]) \Rightarrow r([c]) \)                UE: 2
5. \( p([c]) \)                Assumption
6. \( q([c]) \)                IE: 5, 3
7. \( r([c]) \)                IE: 6, 4
8. \( p([c]) \Rightarrow r([c]) \)                II: 5, 7
9. \( \forall x.(p(x) \Rightarrow r(x)) \)                UI: 8
A proof system is *sound* if and only if every provable conclusion is logically entailed.

If $\Delta \vdash \phi$, then $\Delta \models \phi$.

A proof system is *complete* if and only if every logical conclusion is provable.

If $\Delta \models \phi$, then $\Delta \vdash \phi$. 
Fitch is sound and complete for Propositional Logic.

\[ \Delta \vdash \phi \text{ if and only if } \Delta \models \phi. \]

Fitch is sound and complete for Relational Logic.

\[ \Delta \vdash \phi \text{ if and only if } \Delta \models \phi. \]

Fitch (w/ induction) is sound for Functional Logic, not complete.

If \( \Delta \vdash \phi \), then \( \Delta \models \phi \).
Automated Theorem Proving
   *Building computers that can prove theorems*
CS 257 - Automate Reasoning

Logic Programming
   *Using Logic to represent knowledge, program computers*
CS 151 - Logic Programming
CS 227B - General Game Playing

Advanced Logics
   *Metadata, Paraconsistent Reasoning, Probabilities*
Phil 161
Final Exam