CSE 2315 - Discrete Structures
Lecture 5: Predicate Logic

1 Motivation

• The use of predicates, variables, and quantifiers allows to represent a large number of arguments and expressions in formal logic.

• But: It is generally not possible to build truth tables for well formed formulas which contain variables and quantifiers because the set of objects is potentially infinite.

• Formal logic has thus to be used in order to prove the validity of argument in predicate logic.

2 Proofs using Predicate Logic

• In propositional logic we used proof sequences to show the validity of an argument.
  – Equivalence rules are derived from tautological equivalences and preserve the truth values based solely on the structure.
  – Inference rules are derived from valid arguments which provide correct implications and thus preserve correctness due to their structure.

• Since all rules from propositional logic are truth preserving independent of interpretation, they are also applicable in the presence of predicates and quantifiers and form the basis rules for predicate logic.

\[(\forall x)R(x) \land ((\forall x)R(x) \rightarrow (\forall x)S(x)) \rightarrow (\forall x)S(x)\]

1. \((\forall x)R(x)\) hyp
2. \((\forall x)R(x) \rightarrow (\forall x)S(x)\) hyp
3. \((\forall x)S(x)\) 1, 2 mp

– But: The equivalent formula \((\forall x)R(x) \land ((\forall y)R(y) \rightarrow (\forall z)S(z)) \rightarrow (\forall x)S(x)\) can not be proven because of the different names of the variables.

• Additional rules are required to deal with variables and quantifiers.
– Additional equivalence rules:

<table>
<thead>
<tr>
<th>Rule Name</th>
<th>Expression</th>
<th>Equivalent Expression</th>
</tr>
</thead>
<tbody>
<tr>
<td>Negation</td>
<td>$(\forall x)P(x)$</td>
<td>$(\exists x)\neg P(x)$</td>
</tr>
</tbody>
</table>

Note: Equivalence rules are always invertible and can be applied anytime before and in a proof sequence.

– Additional inference rules:

* To deal with quantifiers and different variables, predicate logic attempts to first strip off the quantifiers and rename variables, apply the rules from propositional logic and then reintroduce quantifiers if necessary.

* When stripping off quantifiers you have to be careful to remain truth preserving.

<table>
<thead>
<tr>
<th>Rule Name</th>
<th>From</th>
<th>Can Derive</th>
<th>Condition for use</th>
</tr>
</thead>
<tbody>
<tr>
<td>Universal instantiation</td>
<td>$(\forall x)P(x)$</td>
<td>$P(y)$</td>
<td>$y$ is a free variable in $P(y)$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$P(a)$</td>
<td></td>
</tr>
<tr>
<td>Existential instantiation</td>
<td>$(\exists x)P(x)$</td>
<td>$P(y)$</td>
<td>$y$ is a new variable</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$P(a)$</td>
<td>$a$ is a new constant</td>
</tr>
</tbody>
</table>

* Universal instantiation follows the fact that if a property holds for all objects $x$ then it has to hold also for the unspecified object $y$ or $a$. The limitation is necessary since the choice of $y$ can not limit the “choices” of subsequent quantifiers.

  · This rule does therefore not allow to use $(\forall x)(\exists y)G(y, x)$ to derive $(\exists y)G(y, y)$.

  · Example: Bill is tall. All jockeys are not tall. Therefore Bill is not a jockey.

$$T(b) \land (\forall x)(T(x) \rightarrow J(x)) \rightarrow \neg J(b)$$

1. $T(b)$ hyp
2. $(\forall x)(T(x) \rightarrow \neg J(x))$ hyp
3. $T(b) \rightarrow \neg J(b)$ 2 ui
4. $\neg J(b)$ 1, 3 mp

* Existential instantiation states that if there exists a particular object which has a given property, then we can call this object $y$. The restrictions are necessary since $y$ is a specific object of which we have no other knowledge than that it exists.

  · It is for example not allowed to use this rule to derive $G(y, z)$ from $(\exists x)(G(x, z) \land P(y))$. 
Example:

1. \((\forall x)(P(x) \rightarrow Q(x))\) hyp
2. \((\exists y)P(y)\) hyp
3. \(P(a)\) 2 ei
4. \(P(a) \rightarrow Q(a)\) 1 ui
5. \(Q(a)\) 3, 4 mp

* Hint: Since existential instantiation can only use variable or constants which have not been used, it should always be applied as early as possible.

* When adding quantifiers you again have to be careful to remain truth preserving.

<table>
<thead>
<tr>
<th>Rule Name</th>
<th>From</th>
<th>Can Derive</th>
<th>Condition for use</th>
</tr>
</thead>
<tbody>
<tr>
<td>Universal generalization</td>
<td>(P(x))</td>
<td>((\forall x)P(x))</td>
<td>(x) was not a free variable in any hypothesis used to derive (P(x)) and it has not become free by the use of existential instantiation</td>
</tr>
<tr>
<td>Existential generalization</td>
<td>(P(x))</td>
<td>((\exists x)P(x))</td>
<td>(x) does not appear in (P(a))</td>
</tr>
</tbody>
</table>

* Universal generalization states that if a property holds for a general, arbitrary object \(x\), then it also holds for all objects. The limitation is necessary since the free occurrence or existential quantifications of a variable makes it not general but rather point to a specific object.

  · This rule does therefore not allow to use a hypothesis \(P(x)\) to derive \((\forall x)P(x)\).
  
  · Example: Everybody is tall. All jockeys are not tall. Therefore everybody is not jockeys.

\[
(\forall x)T(x) \land (\forall x)(T(x) \rightarrow (\forall x)J(x)) \rightarrow \overline{J(b)}
\]

1. \((\forall x)T(x)\) hyp
2. \((\forall x)(T(x) \rightarrow \overline{J(x)})\) hyp
3. \(T(x)\) 1 ui
4. \(T(x) \rightarrow \overline{J(x)}\) 2 ui
5. \(\overline{J(x)}\) 3, 4 mp
6. \((\forall x)\overline{J(x)}\) 5 ug

* Existential generalization states that if a property is true for a given element, then there exists an object for which the property holds. The restriction in the case of a
constant is necessary since the variable \( y \) has to be able to refer to the element \( a \) and can thus not appear in any other form in \( P(a) \).

- It is for example not allowed to use this rule to derive \( (\exists x)G(x, x) \) from \( G(x, a) \).
- Example: Bill is tall. All jockeys are not tall. Therefore there exists someone who is not a jockey.

\[
T(b) \land (\forall x)(T(x) \rightarrow \overline{J(x)}) \rightarrow (\exists x)\overline{J(x)}
\]

1. \( T(b) \) hyp
2. \( (\forall x)(T(x) \rightarrow \overline{J(x)}) \) hyp
3. \( T(b) \rightarrow \overline{J(b)} \) 2 ui
4. \( \overline{J(b)} \) 1, 3 mp
5. \( (\exists x)\overline{J(x)} \) 4 eg

* Hint: To make handling of the conditions easier, try to think of variables as belonging to one of two classes; the ones that refer to an arbitrary object and the ones that refer to a specific object. Universal instantiation results in variables refering to arbitrary, unconstrained objects. Such variables, in urm, can be used in universal and existential generalization. The ones refering to a specific object are mainly free variables in the original argument or result from existential instantiation and can be used only for existential generalization.

- Adding Hypotheses
  - Sometimes the available hypotheses are not sufficient and additional hypotheses are needed to be able to finish a proof sequence.
    - All tautologies or valid arguments can be introduced as additional facts in the proof sequence.
    - The deduction method is one example where an additional hypothesis is introduced.
    - Addition is another equivalence rule which expands the set of facts available.
  - The introduction of temporary hypotheses provides a general means to construct new, valid arguments which are related to the proof sequence and which can be added in order to facilitate proofs.
    - Here a general hypothesis \( P \) is introduced at a point in the proof sequence. Then rules are applied to this part of the proof until a suitable formula \( Q \) is derived. At this point all steps since the introduction of the temporary hypothesis are discarded and only the valid argument \( P \rightarrow Q \) is introduced into the proof sequence.
    - Example:
\[(P(x) \rightarrow (\forall y)Q(x, y)) \rightarrow (\forall y)(P(x) \rightarrow Q(x, y))\]

1. \(P(x) \rightarrow (\forall y)Q(x, y)\) hyp
2. \(P(x)\) temporary hyp
3. \((\forall y)Q(x, y)\) 1, 2 mp
4. \(Q(x, y)\) 3 ui
5. \(P(x) \rightarrow Q(x, y)\) temporary hyp discharged
6. \((\forall y)(P(x) \rightarrow Q(x, y))\) 5 ug

– Note: The proof of arguments in predicate logic requires considerable imagination since the choice of variable names in the instantiation steps, as well as the introduction of temporary hypotheses can drastically influence the proof.

– Hint: In predicate logic always look also at what you want to prove and think about what you would need in order to prove it.

• Exercises - prove or disprove

– If John’s wallet was stolen, then the thief was in the office. Only Jack could have stolen the wallet but Jack was not in the office. Therefore the wallet was not stolen.

\[(\forall x)(S(x) \rightarrow O(x)) \land \overline{S(j)} \rightarrow (\forall x)\overline{S(x)} \land \overline{O(j)} \rightarrow (\forall x)\overline{S(x)}\]

– \((\exists x)R(x) \land (\exists x)(R(x) \land S(x)) \rightarrow (\exists x)\overline{S(x)}\)
– \((\exists x)P(x) \land (\exists x)Q(x) \rightarrow (\exists x)(P(x) \land Q(x))\)
– \((\exists x)P(x) \lor (\exists x)Q(x) \rightarrow (\exists x)(P(x) \lor Q(x))\)
– \((\exists x)(\forall y)Q(x, y) \rightarrow (\forall y)(\exists x)Q(x, y)\)
– \((\forall y)(\exists x)Q(x, y) \rightarrow (\exists x)(\forall y)Q(x, y)\)