Mathematical and Logical Foundations of Computer Science

Lecture 12 - Predicate Logic (Natural Deduction Proofs)

Vincent Rahli

(some slides were adapted from Rajesh Chitnis' slides)

University of Birmingham

Where are we?

- Symbolic logic
- Propositional logic
- **Predicate logic**
- Constructive vs. Classical logic
- Type theory

Today

- Natural Deduction proofs for Predicate Logic
- $\forall/\exists$ rules
- substitution

Further reading:

- Chapter 8 of http://leanprover.github.io/logic_and_proof/

Recap: Beyond Propositional Logic

Famous derivation in logic:

- All men are mortal
- Socrates is a man
- Therefore, Socrates is mortal

Cannot be expressed in propositional logic

We introduced:

- predicates, quantifiers, variables, functions, and constants

We can write this argument as $\forall x.(p(x) \to q(x)), p(s) \vdash q(s)$

- **Domain**: people
- **Predicates**: $p(x) =$ "x is a man"; $q(x) =$ "x is mortal"
- **Quantifier**: The "for all" symbol $\forall$
- **Variable**: x to denote an element of the domain
- **Constant**: s which stands for Socrates

Recap: Syntax

The syntax of predicate logic is defined by the following grammar:

$$
\begin{array}{lcl}
t & ::= & x \mid f(t, \ldots, t) \\
P & ::= & p(t, \ldots, t) \mid \neg P \mid P \wedge P \mid P \vee P \mid P \rightarrow P \mid \forall x.P \mid \exists x.P
\end{array}
$$

where:

- x ranges over variables
- f ranges over function symbols
- $f(t_1, \ldots, t_n)$ is a well-formed term only if f has arity n
- p ranges over predicate symbols
- $p(t_1, \ldots, t_n)$ is a well-formed formula only if p has arity n

The pair of a collection of function symbols, and a collection of predicate symbols, along with their arities, is called a **signature**.

The scope of a quantifier extends as far right as possible. E.g., $P \wedge \forall x.p(x) \vee q(x)$ is read as $P \wedge \forall x.(p(x) \vee q(x))$

Recap: Examples

Consider the following domain and signature:

- Domain: $\mathbb{N}$
- Functions: $0, 1, 2, \ldots$ (arity 0); $+$ (arity 2)
- Predicates: $\texttt{prime}$, $\texttt{even}$, $\texttt{odd}$ (arity 1); $=$, $>$, $\geqslant$ (arity 2)

Express the following sentences in predicate logic

- All prime numbers are either 2 or odd.
  $\forall x.\texttt{prime}(x) \to x = 2 \vee \texttt{odd}(x)$
- Every even number is equal to the sum of two primes.
  $\forall x.\texttt{even}(x) \to \exists y.\exists z.\texttt{prime}(y) \wedge \texttt{prime}(z) \wedge x = y + z$
- There is no number greater than all numbers.
  $\neg \exists x.\forall y.x \geqslant y$
- All numbers have a number greater than them.
  $\forall x.\exists y.y > x$

One more example (from the book – section 7.6.2)

Domain is people, and we have 6 predicates

politician(x) rich(x) crazy(x) trusts(x, y) knows(x, y) related-to(x, y)

Express the following sentences in predicate logic

- Nobody trusts a politician.
  $\neg\exists x.\exists y.\text{politician}(y) \wedge \text{trusts}(x, y)$
- Anyone who trusts a politician is crazy.
  $\forall x.(\exists y.\text{politician}(y) \wedge \text{trusts}(x, y)) \rightarrow \text{crazy}(x)$
- Everyone knows someone who is related to a politician.
  $\forall x.\exists y.\text{knows}(x, y) \wedge \exists z.\text{politician}(z) \wedge \text{related-to}(y, z)$
- Everyone who is rich is either a politician or knows a politician.
  $\forall x.\text{rich}(x) \rightarrow \text{politician}(x) \vee \exists y.\text{knows}(x, y) \wedge \text{politician}(y)$

Inference rules for ∀ and ∃?

Propositional logic: Each connective has at least 2 inference rules

- At least 1 for introduction
- At least 1 for elimination

Introduction and elimination rules for ∀ and ∃?

$$\frac{?}{\forall y.P}\ [\forall I] \qquad \frac{\forall x.P}{?}\ [\forall E]$$

$$\frac{?}{\exists y.P}\ [\exists I] \qquad \frac{\exists x.P}{?}\ [\exists E]$$

Free & Bound Variables

Free variables and **Bound** variables:

Bound variables:

- Consider the formula $\forall x.\mathsf{even}(x) \vee \mathsf{odd}(x)$
  Here the variable x is **bound** by the quantifier $\forall$
- $\forall x.\mathsf{even}(x) \vee \mathsf{odd}(x)$ is considered the same as $\forall y.\mathsf{even}(y) \vee \mathsf{odd}(y)$
  Renaming a **bound** variable **doesn't** change the meaning!

Free variables:

- Consider the formula $\forall y.x \leqslant y$
- y is a **bound** variable and x is a **free** variable
- variables are **free** if they are not bound
- $\forall y.x \leqslant y$ is the **same** as $\forall z.x \leqslant z$
- $\forall y.x \leqslant y$ is **not the same** as $\forall y.w \leqslant y$
- Renaming a **free** variable **changes** the meaning!

Free & Bound Variables

The **scope** of a quantified formula of the form $\forall x.P$ or $\exists x.P$ is P. The quantifier are said to **bind** x.

Bound variables: a variable x occurs bound in a formula, if it occurs in the scope of a quantifier quantifying x

Free variables: a variable x occurs free in a formula, if it does not occur in the scope of a quantifier quantifying x

The set of variables occurring free/bound in a terms and formulas is recursively computed as follows:

$$
\begin{array}{lcl}
\mathtt{fv}(x) & = & \{x\} \\
\mathtt{fv}(f(t_1, \ldots, t_n)) & = & \mathtt{fv}(t_1) \cup \ldots \cup \mathtt{fv}(t_n) \\
\mathtt{fv}(p(t_1, \ldots, t_n)) & = & \mathtt{fv}(t_1) \cup \ldots \cup \mathtt{fv}(t_n) \\
\hline
\mathtt{fv}(\neg P) & = & \mathtt{fv}(P) \\
\mathtt{fv}(P_1 \wedge P_2) & = & \mathtt{fv}(P_1) \cup \mathtt{fv}(P_2) \\
\mathtt{fv}(P_1 \vee P_2) & = & \mathtt{fv}(P_1) \cup \mathtt{fv}(P_2) \\
\mathtt{fv}(P_1 \rightarrow P_2) & = & \mathtt{fv}(P_1) \cup \mathtt{fv}(P_2) \\
\hline
\mathtt{fv}(\forall x.P) & = & \mathtt{fv}(P) \backslash \{x\} \\
\mathtt{fv}(\exists x.P) & = & \mathtt{fv}(P) \backslash \{x\}
\end{array}
$$

$$
\begin{array}{lcl}
\mathtt{bv}(p(t_1, \ldots, t_n)) & = & \emptyset \\
\hline
\mathtt{bv}(\neg P) & = & \mathtt{bv}(P) \\
\mathtt{bv}(P_1 \wedge P_2) & = & \mathtt{bv}(P_1) \cup \mathtt{bv}(P_2) \\
\mathtt{bv}(P_1 \vee P_2) & = & \mathtt{bv}(P_1) \cup \mathtt{bv}(P_2) \\
\mathtt{bv}(P_1 \rightarrow P_2) & = & \mathtt{bv}(P_1) \cup \mathtt{bv}(P_2) \\
\hline
\mathtt{bv}(\forall x.P) & = & \mathtt{bv}(P) \cup \{x\} \\
\mathtt{bv}(\exists x.P) & = & \mathtt{bv}(P) \cup \{x\}
\end{array}
$$

Free & Bound Variables

What are the free variables of the following formulas

- $P_1 = (\texttt{odd}(x) \wedge \exists y.y < x \wedge \texttt{odd}(y))$
  $\texttt{fv}(P_1) = \{x\}$
- $P_2 = (\texttt{odd}(x) \wedge x > y \wedge \exists y.y < x \wedge \texttt{odd}(y))$
  $\texttt{fv}(P_2) = \{x, y\}$
- $P_3 = (\forall x.\texttt{odd}(x) \wedge x > y \wedge \exists y.y < x \wedge \texttt{odd}(y))$
  $\texttt{fv}(P_3) = \{y\}$

Note: In $(\texttt{odd}(x) \wedge x > y \wedge \exists y.y < x \wedge \texttt{odd}(y))$ the green occurrence of y is **not** the same variable as the red occurrence of y.

The formula $(\texttt{odd}(x) \wedge x > y \wedge \exists y.y < x \wedge \texttt{odd}(y))$ is considered the same as $(\texttt{odd}(x) \wedge x > y \wedge \exists z.z < x \wedge \texttt{odd}(z))$

Inference rules for $\forall$ and $\exists$?

Propositional logic: Each connective has at least 2 inference rules

- At least 1 for introduction
- At least 1 for elimination

Introduction and elimination rules for $\forall$ and $\exists$?

$$\frac{?}{\forall y.P}\ [\forall I] \qquad \frac{\forall x.P}{?}\ [\forall E]$$

$$\frac{?}{\exists y.P}\ [\exists I] \qquad \frac{\exists x.P}{?}\ [\exists E]$$

WARNING ⚠

Trickier than inference rules from propositional logic!

We need to be careful with free and bound variables!

Inference Rule for “for all elimination” – 1st attempt

$$\frac{\forall x.P}{?}\ [\forall E]$$

What can we conclude from the fact that P is true for all x?

Predicate P is true for all elements x of the domain

- For any element of the domain t, we can deduce that P is true where x is replaced by t is true
- This “replacing” operation is a **substitution** operation as seen in lecture 2.
- However, we now have to be careful with free/bound variables.

Substitution

Substitution is defined recursively on terms and formulas:

$P[x\backslash t]$ substitute all the free occurrences of x in P with t.

1st attempt (WRONG)

$$
\begin{array}{lcl}
x[x\backslash t] & = & t \\
x[y\backslash t] & = & x \\
(f(t_1,\ldots,t_n))[x\backslash t] & = & f(t_1[x\backslash t],\ldots,t_n[x\backslash t]) \\
(p(t_1,\ldots,t_n))[x\backslash t] & = & p(t_1[x\backslash t],\ldots,t_n[x\backslash t]) \\
\hline
(\neg P)[x\backslash t] & = & \neg P[x\backslash t] \\
(P_1 \wedge P_2)[x\backslash t] & = & P_1[x\backslash t] \wedge P_2[x\backslash t] \\
(P_1 \vee P_2)[x\backslash t] & = & P_1[x\backslash t] \vee P_2[x\backslash t] \\
(P_1 \to P_2)[x\backslash t] & = & P_1[x\backslash t] \to P_2[x\backslash t] \\
\hline
(\forall x.P)[x\backslash t] & = & \forall x.P \\
(\exists x.P)[x\backslash t] & = & \exists x.P \\
(\forall y.P)[x\backslash t] & = & \forall y.P[x\backslash t] \\
(\exists y.P)[x\backslash t] & = & \exists y.P[x\backslash t]
\end{array}
$$

Why is this wrong? $(\forall y.y > x)[x\backslash y]$ would return $\forall y.y > y$, where the free y is now bound! The free y got **captured**! The red occurrences of y stand for different variables than the green ones.

Substitution

Substitution is defined recursively on terms and formulas:

$P[x\backslash t]$ substitute all the free occurrences of x in P with t.

2nd attempt (CORRECT)

$$
\begin{array}{lcl}
x[x\backslash t] & = & t \\
x[y\backslash t] & = & x \\
(f(t_1,\ldots,t_n))[x\backslash t] & = & f(t_1[x\backslash t],\ldots,t_n[x\backslash t]) \\
(p(t_1,\ldots,t_n))[x\backslash t] & = & p(t_1[x\backslash t],\ldots,t_n[x\backslash t]) \\
\hline
(\neg P)[x\backslash t] & = & \neg P[x\backslash t] \\
(P_1 \wedge P_2)[x\backslash t] & = & P_1[x\backslash t] \wedge P_2[x\backslash t] \\
(P_1 \vee P_2)[x\backslash t] & = & P_1[x\backslash t] \vee P_2[x\backslash t] \\
(P_1 \rightarrow P_2)[x\backslash t] & = & P_1[x\backslash t] \rightarrow P_2[x\backslash t] \\
\hline
(\forall x.P)[x\backslash t] & = & \forall x.P \\
(\exists x.P)[x\backslash t] & = & \exists x.P \\
(\forall y.P)[x\backslash t] & = & \forall y.P[x\backslash t], \text{ if } y \notin \mathtt{fv}(t) \\
(\exists y.P)[x\backslash t] & = & \exists y.P[x\backslash t], \text{ if } y \notin \mathtt{fv}(t)
\end{array}
$$

The additional **conditions** ensure that **free variables do not get captured**.

These conditions can always be met by silently renaming bound variables before substituting.

Inference Rule for "for all elimination" – 2nd attempt

The correct rule is:

$$\frac{\forall x.P}{P[x\backslash t]}\ [\forall E]$$

Condition: $\texttt{fv}(t)$ must not clash with any bound variables of P

Example: consider the formula $\forall x.\exists y.y > x$

- True over domain of natural numbers
- P is $\exists y.y > x$
- Let t be y
- This condition guarantees that we can do the substitution
- Substituting x with y without renaming bound variables would give the wrong answer (see previous slide)
- Therefore, we first rename bound variables that clash with $\texttt{fv}(t)$, i.e., with y: $\exists z.z > x$
- Then, we substitute: $\exists z.z > y$

Inference Rule for "for all introduction"

$$\frac{?}{\forall x.P} \quad [\forall I]$$

When can we conclude P is true for all x?

If we have proved P for a "**general/representative/typical**" variable

$$\frac{P[x\backslash y]}{\forall x.P} \quad [\forall I]$$

Condition: y must not be free in any not-yet-discharged hypothesis or in $\forall x.P$

What could go wrong without this condition?

Otherwise, given the assumption $y > 2$, we could derive $\forall x.x > 2$, which is clearly wrong.

Inference Rule for "exists introduction"

$$\frac{?}{\exists x.P}\ [\exists I]$$

When can we conclude P is true for some x?

If we have proved predicate P for an element of the domain

$$\frac{P[x\backslash t]}{\exists x.P}\ [\exists I]$$

Condition: $\texttt{fv}(t)$ must not clash with $\texttt{bv}(P)$

Example: Consider the predicate $P = (\forall y.y = x)$

- Without the substitution conditions $P[x\backslash y]$ would be true
- We could then deduce $\exists x.\forall y.y = x$, i.e., numbers are all equal to each other — obviously incorrect!
- The substitution conditions prevents such captures
- $[\exists I]$'s condition guarantees that the substitution conditions hold

Inference Rule for "exists elimination"

$$\frac{\exists x. P}{?} \; [\exists E]$$

What can we conclude from the fact that P is true for some x?

We know that it holds about some element of the domain,
but we do not know which

$$\dfrac{\exists x. P \qquad \begin{array}{c} \dfrac{}{P[x \backslash y]}\,1 \\ \vdots \\ Q \end{array}}{Q} \; 1 \; [\exists E]$$

Condition: y must not be free in Q or in not-yet-discharged hypotheses or in $\exists x. P$

This rule is similar to OR-elimination!

All four inference rules in one slide

$$\frac{P[x\backslash y]}{\forall x.P}\ [\forall I]$$

Condition: y must not be free in any not-yet-discharged hypothesis or in $\forall x.P$

$$\frac{\forall x.P}{P[x\backslash t]}\ [\forall E]$$

Condition: $\texttt{fv}(t)$ must not clash with $\texttt{bv}(P)$

$$\frac{P[x\backslash t]}{\exists x.P}\ [\exists I]$$

Condition: $\texttt{fv}(t)$ must not clash with $\texttt{bv}(P)$

$$\dfrac{\exists x.P \qquad \begin{array}{c}\overline{P[x\backslash y]}^{\,1}\\ \vdots \\ Q\end{array}}{Q}\ 1\ [\exists E]$$

Condition: y must not be free in Q or in not-yet-discharged hypotheses or in $\exists x.P$

A simple proof

Prove that $(\forall z.p(z)) \to \forall x.p(x) \lor q(x)$

We use backward reasoning

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\forall z.p(z)}\ 1}{p(y)}\ [\forall E]}{p(y) \lor q(y)}\ [\lor I_L]}{\forall x.p(x) \lor q(x)}\ [\forall I]}{(\forall z.p(z)) \to \forall x.p(x) \lor q(x)}\ 1\ [\to I]
$$

Conditions:

- y does not occur free in not-yet-discharged hypotheses or in $\forall x.p(x) \lor q(x)$
- y does not clash with bound variables in $p(z)$

A simple proof

More generally, we can prove:

$$\dfrac{\dfrac{\dfrac{\dfrac{\overline{\forall z.P}^{\,1}}{P[x\backslash y]}\,[\forall E]}{P[x\backslash y] \vee Q[x\backslash y]}\,[\vee I_L]}{\forall x.P \vee Q}\,[\forall I]}{(\forall z.P) \to \forall x.P \vee Q}\,1\ [\to I]$$

We assume that y does not occur in P or Q

Conclusion

What did we cover today?

- Natural Deduction proofs for Predicate Logic
- $\forall/\exists$ rules
- substitution

Next time?

- Natural Deduction proofs for Predicate Logic – continued