Predicate Logic:
Proofs and Models

April 14, 2004
Computability and Logic

Recall:

- Predicate logic allows us to describe properties of (or relations between) things.

$\forall n.\ (\text{prime}(n) \wedge \text{gt}(n,2) \rightarrow \text{odd}(n))$

$\forall x.\ \forall \varepsilon.\ \exists \delta.\ \forall y.\ (\ |x-y| < \delta \rightarrow\ |f(x)-f(y)| < \delta\ )$
$\forall \varepsilon.\ \forall x.\ \exists \delta.\ \forall y.\ (\ |x-y| < \delta \rightarrow\ |f(x)-f(y)| < \delta\ )$
$\forall \varepsilon.\ \exists \delta.\ \forall x.\ \forall y.\ (\ |x-y| < \delta \rightarrow\ |f(x)-f(y)| < \delta\ )$

Renaming Bound Variables

- Which of the following are logically equivalent?

$\exists\, y.\ x < y$
$\exists\, z.\ x < z$
$\exists\, x.\ x < x$

- What 's the problem?

Substitution

- The notation [t/x] written after any phrase means to take *free* occurrences of x and replace them by t.

  $(x = 0 \vee \exists x.(x > y))\ [3/y] =$
  $(x = 0 \vee \exists x.(x > y))\ [3/x] =$

- We must make sure that free variables in t remain free variables (are not captured).
  - Can always avoid this by renaming bound variables.

  $(x = 0 \vee \exists x.(x > y))\ [(x-1)/y] =$

∀ Elimination

$$\frac{\forall x.\ \phi}{\phi[t/x]}$$ (t is free for x in ϕ)

E.g.,

1.	$\forall x.\ (x=x)$	Assump
2.	$(2+2) = (2+2)$	∀x e

"If ϕ is true for all values, then it must be true for any specific term t."

∀ Introduction

$$\frac{\boxed{\begin{array}{c} x_0 \\ \vdots \\ \phi[x_0/x] \end{array}}}{\forall x.\phi}$$ (x_0 is fresh)

E.g.,

1.	$\boxed{x_0}$	Assump
2.	$x_0=0 \lor \neg(x_0 = 0)$	LEM
3.	$\forall x.\ (x_0=0 \lor \neg(x_0 = 0))$	∀ x i

"If ϕ is true for a term x_0 about which we assume *nothing*, then it must be true for every term."

(∀x e) (∀x i) Example

- Derive $(\forall x)(p(x) \to q(x)),\ (\forall x)\ p(x) \vdash (\forall x)\ q(x)$:

1.	$(\forall x)(p(x) \to q(x))$	Premise
2.	$(\forall x)\ p(x)$	Premise

∃ Elimination

$$\frac{\phi[t/x]}{\exists x.\ \phi}$$ (t is free for x in ϕ)

E.g.,

1.	…	…
2.	$3 > 2$	…
3.	$\exists x.\ (x > 2)$	∃x i

"If ϕ is true for a term t, then we know it is true for some term."

∃ Elimination

$$\frac{\exists x.\phi \qquad \boxed{\begin{array}{c} x_0 \\ \phi[x_0/x] \\ \cdot \\ \cdot \\ \cdot \\ \chi \end{array}}}{\chi}$$

(x_0 is fresh and does not appear in χ)

E.g.,

1.	$\exists x. (x > 2)$	Assump
2.	$\boxed{x_0}$ $x_0 > 2$	Assump.
3.	$x_0+1 > 2+1$	???
4.	$\exists y. (y > 2+1)$	∃x i
5.	$\exists y. (y > 2+1)$	∃x e

"If we know there is an x satisfying ϕ, then we can temporarily let x_0 be that value and see what we can prove."

(∃x i) (∃x e) Example

- Derive $(\forall x)(p(x) \to q(x)),\ (\exists x)\, p(x) \vdash (\exists x)\, q(x)$:

1.	$(\forall x)\,(p(x) \to q(x))$	Premise
2.	$(\exists x)\, p(x)$	Premise
3.		

Equality

- Introduction
- Elimination

$$\frac{}{t = t}$$

$$\frac{t_1 = t_2 \quad \phi[t_1/x]}{\phi[t_2/x]}$$

To show: equality is symmetric and transitive

$\neg\forall x.\phi \vdash \exists x.\neg\phi$

Hint: RAA twice

Interpretations

- Is the proposition $\forall x.\ p(x,x)$ true?

Interpretations

- Each predicate logic formula can be viewed as making a statement (which could be true or false) about a (mathematical) structure. An **interpretation** for a set of formulas consists of:
  - A **domain**: that contains all individuals of interest.
  - A set of **constants** from the domain, one for each constant **symbol** in the logical language.
  - A set of **functions** mapping n-tuples of domain elements into domain elements, one for each function **symbol** in the logical language.
  - A set of **predicates** mapping n-tuples of domain elements into {T, F}, one for each predicate **symbol** in the logical language.

Example: Peano Axioms

- $\forall x.\ \neg(\mathbf{S}(x) = \mathbf{0})$
- $\forall x.\ \forall y.\ (\mathbf{S}(x) = \mathbf{S}(y) \rightarrow x=y)$
- $(\ \phi[0/x] \wedge \forall n.(\phi[n/x]\rightarrow\phi[\mathbf{S}(n)/x])\ ) \rightarrow \forall n.\ \phi[n/x]$

  for *every* proposition ϕ

  where is n is free for x in ϕ.

An Interpretation

Are there other interpretations?

- The **domain** is the set {0, 1, 2, 3, …} of natural numbers
- There is one **constant** symbol: **0**.
  The associated domain value is the number 0.
- There is one **function** symbol: **S**.
  The associated function is successor (i.e., +1).
- There is one **predicate** symbol: =.
  The associated mathematical relation is equality.

Theories and Models

- A **Theory** is a set of formulas called **theorems** derived from a basis set of **axioms**.
- Recall: we write $\Gamma \vdash \psi$ to mean that ψ can be derived from the set of sentences (axioms) in Γ.
- Γ could be **infinite**.
  - In any specific proof, only a finite subset of the axioms could actually be used.
  - Usually the set of axioms is *recursive*

- A *model* is an interpretation that makes the axioms *true*.

The book uses the word model to mean *any* interpretation. This is non-standard.

Elementary Number Theory

  - $(\forall x)\ \neg(S(x) = 0)$
  - $(\forall x)\ (\forall y)\ ((S(x) = S(y)) \to (x = y))$
  - $(\forall x)\ (x + 0) = x$
  - $(\forall x)\ (\forall y)\ (x + S(y)) = S(x+y)$
  - $(\forall x)\ (x * 0) = 0$
  - $(\forall x)\ (\forall y)\ (\forall z)\ (x*S(y)) = ((x*y) + x)$
- The function symbols are {S} (1-ary) and {+, *} (2-ary).
- The predicate symbol is =.
- Note that there is no induction.

Non-Uniqueness of Models

- Note that while some theories, such as number theory, may have a standard **intended** model, there is no guarantee that the model is unique.

- Many theories will intentionally have a **range** of models rather than a single model.

Linear Order Theory

- $(\forall x)\ \neg(x < x)$
- $(\forall x)\ (\forall y)\ (x < y) \vee (x = y) \vee (y < x)$
- $(\forall x)\ (\forall y)\ (\forall z)\ (((x < y) \wedge (y < z)) \to (x < z))$
- There are many possible linear orders, with different numbers of elements

Dense Linear Order Theory

- $(\forall x)\ \neg(x < x)$
- $(\exists x)(\exists y)\ (x < y)$
- $(\forall x)\ (\forall y)\ (x < y) \vee (x = y) \vee (y < x)$
- $(\forall x)\ (\forall y)\ (\forall z)\ (((x < y) \wedge (y < z)) \rightarrow (x < z))$
- $(\forall x)\ (\forall z)\ (x < z) \rightarrow (\exists y)\ (x < y) \wedge (y < z))$

Semi-Group Theory

- $(\forall x)(\forall y)(\forall z)\ (x + (y + z)) = ((x + y) + z)$

Monoid Theory

- $(\forall x)(\forall y)(\forall z)\ (x + (y + z)) = ((x + y) + z)$
- $(\forall x)\ ((x + 0) = x)$

Group Theory

- $(\forall x)(\forall y)(\forall z)\ (x + (y + z)) = ((x + y) + z)$
- $(\forall x)\ ((x + 0) = x)$
- $(\forall x)(\exists y)\ ((x + y) = 0)$

Ex: Proving a Theorem in Group Theory Γ

- Show $\Gamma \vdash (\forall x)(\forall y)(\forall z)\ (((y + x) = (z + x)) \rightarrow (y = z))$

1.	x_0	
2.	y_0	
3.	z_0	
4.	$(y_0 + x_0) = (z_0 + x_0)$	Assumption
5.	$(\forall x)(\exists y)\ ((x + y) = 0)$	Group Axiom
6.	$(\exists y)\ ((x_0 + y) = 0)$	∀x e 5
7.	$w_0 \quad (x_0 + w_0) = 0$	Assumption
8.	$(y_0 + x_0) + w_0 = (z_0 + x_0) + w_0$	Equality Rule
9.	$y_0 + (x_0 + w_0) = z_0 + (x_0 + w_0)$	Group Axiom & Eq. (twice)
10.	$y_0 + 0 = z_0 + 0$	Equality Rule
11.	$y_0 = z_0$	Group Axiom (twice)
12.	$y_0 = z_0$	∃y e 7-11
13.	$(y_0 + x_0) = (z_0 + x_0) \rightarrow y_0 = z_0$	→i 4-12
14.	$(\forall z)\ (y_0 + x_0) = (z + x_0) \rightarrow y_0 = z$	∀z i 3-13
15.	$(\forall y)\ (\forall z)\ (y + x_0) = (z + x_0) \rightarrow y = z$	∀y i 2-14
16.	$(\forall x)(\forall y)(\forall z)\ (((y + x) = (z + x)) \rightarrow (y = z))$	∀x i 1-15

Abelian Group Theory

- $(\forall x)(\forall y)(\forall z)\ (x + (y + z)) = ((x + y) + z)$
- $(\forall x)\ ((x + 0) = x)$
- $(\forall x)(\exists y)\ ((x + y) = 0)$
- $(\forall x)(\forall y)\ ((x + y) = (y + x))$

Ring Theory

- $(\forall x)(\forall y)(\forall z)\ (x + (y + z)) = ((x + y) + z)$
- $(\forall x)\ ((x + 0) = x)$
- $(\forall x)(\exists y)\ ((x + y) = 0)$
- $(\forall x)(\forall y)\ ((x + y) = (y + x))$
- $(\forall x)(\forall y)(\forall z)\ ((x * (y * z)) = ((x * y) * z))$
- $(\forall x)(\forall y)(\forall z)\ ((x * (y + z)) = ((x * y) + (x * z)))$

Field Theory

- $(\forall x)(\forall y)(\forall z)\ (x + (y + z)) = ((x + y) + z)$
- $(\forall x)\ ((x + 0) = x)$
- $(\forall x)(\exists y)\ ((x + y) = 0)$
- $(\forall x)(\forall y)\ ((x + y) = (y + x))$
- $(\forall x)(\forall y)(\forall z)\ ((x * (y * z)) = ((x * y) * z))$
- $(\forall x)(\forall y)(\forall z)\ ((x * (y + z)) = ((x * y) + (x * z)))$
- $(\forall x)(\forall y)\ ((x * y) = (y * x))$
- $(\forall x)\ ((x * 1) = x)$
- $(\forall x)\ (\neg(x = 0) \rightarrow ((\exists y)\ ((x * y) = 1)))$
- $\neg(1 = 0)$