CAS 701 Fall 2002

05 Partial Functions and Undefined Terms

Instructor: W. M. Farmer

Revised: 19 November 2002

What are Partial Functions?

- Each function f has:
  - A domain of definition D_f where it is defined
  - A domain of application D_f^* where it can be applied

- Examples:
  - $\forall x : \mathbf{R} \, . \, x \neq 0 \Rightarrow x/x = 1$
  - $\forall x : \mathbf{R} \, . \, f(x) \simeq \sqrt{1 - x^2}$

- A function f is **total** if $D_f = D_f^*$

- A function f is **partial** if $D_f \subseteq D_f^*$

The Problem of Undefinedness

- Partial functions are ubiquitous in mathematics and computer science

- Definite description is a powerful technique for defining new objects and concepts

- The use of partial functions and definite description naturally leads to undefined terms

- Traditional logics do not directly admit undefined terms due to the

  Existence Assumption: Terms always have a denotation

Approaches to Undefinedness

1. Nondenoting terms are non-well-formed terms
2. Partial functions are represented as relations
3. Partial functions are considered total functions with unspecified values
4. Partial functions are viewed as total functions with smaller domain sorts
5. The value of an undefined term is an exceptional value
6. The value of an undefined term is a nonexistent value
7. Terms and formulas may be nondenoting
8. Terms may be nondenoting, but formulas are always denoting

The Traditional Approach to Partial Functions and Undefinedness

- **Terms may be undefined**
  - Variables and constants are always defined
  - Definite descriptions may be undefined: $(\mathrm{I}\, x : \mathbf{R} \,.\, x * x = 2)$
  - Functions may be partial and thus their applications may be undefined: $1/0$, $\sqrt{-1}$
  - An application of a function is undefined if any argument is undefined: $0 * (1/0)$

- **Formulas are always true or false**
  - Predicates are always total
  - An application of a predicate is false if any argument is undefined: $1/0 = 1/0$

Partial First-Order Logic (PFOL)

- Admits undefined terms, partial functions, and definite descriptions

- Semantics is based on the traditional approach to partial functions and undefinedness
  - Terms may be undefined
  - Formulas always denote true or false

- PFOL is a “logic of definedness”, not a “logic of existence”
  - Undefined terms are indiscernible

- The new machinery—partial functions and definite descriptions—is purely a convenience and is eliminable

Earlier Logics Similar to PFOL

- R. Schock (1968)
- T. Burge (1971)
- M. Beeson (1985)
- L. Monk (1986)
- S. Feferman (1990)
- LUTINS, W. Farmer, J. Guttman, J. Thayer (1990)
- D. Parnas (1993)

Syntax of PFOL: Symbols

- The **logical symbols** are present in every PFOL language:
  - Usual **logical constants**: $=, \neg, \Rightarrow, \forall$
  - **Definite description operator**: I

- The **nonlogical symbols** characterize a PFOL language

- A **language** of PFOL is a tuple $L = (\mathcal{V}, \mathcal{C}, \mathcal{F}, \mathcal{P})$ where:
  - $\mathcal{V}$ is an infinite set of symbols called **variables**
  - $\mathcal{C}$ is a set of symbols called **individual constants**
  - $\mathcal{F}$ is a set of **function symbols**, each with an assigned arity ≥ 1
  - $\mathcal{P}$ is a set of **predicate symbols**, each with an assigned arity ≥ 1, that includes $=$ as a 2-ary predicate symbol
  - $\mathcal{V}$, $\mathcal{C}$, $\mathcal{F}$, and $\mathcal{P}$ are pairwise disjoint

Syntax of PFOL: Terms and Formulas

T1 $\dfrac{x \in \mathcal{V}}{\textbf{term}_L[x]}$ **T2** $\dfrac{c \in \mathcal{C}}{\textbf{term}_L[c]}$

T3 $\dfrac{x \in \mathcal{V},\ \textbf{form}_L[\varphi]}{\textbf{term}_L[(\mathrm{I}\, x \,.\, \varphi)]}$

T4 $\dfrac{f \in \mathcal{F}\ (n\text{-ary}),\ \textbf{term}_L[t_1], \ldots, \textbf{term}_L[t_n]}{\textbf{term}_L[f(t_1, \ldots, t_n)]}$

F1 $\dfrac{p \in \mathcal{P}\ (n\text{-ary}),\ \textbf{term}_L[t_1], \ldots, \textbf{term}_L[t_n]}{\textbf{form}_L[p(t_1, \ldots, t_n)]}$

F2 $\dfrac{\textbf{form}_L[\varphi]}{\textbf{form}_L[\neg\varphi]}$ **F3** $\dfrac{\textbf{form}_L[\varphi],\ \textbf{form}_L[\psi]}{\textbf{form}_L[(\varphi \Rightarrow \psi)]}$

F4 $\dfrac{x \in \mathcal{V},\ \textbf{form}_L[\varphi]}{\textbf{form}_L[(\forall\, x \,.\, \varphi)]}$

Some Abbreviations

$(s = t)$	for	$= (s, t)$
$(s \neq t)$	for	$\neg(s = t)$
$(\varphi \wedge \psi)$	for	$\neg(\varphi \Rightarrow \neg\psi)$
$(\varphi \vee \psi)$	for	$\neg\varphi \Rightarrow \psi$
$(\varphi \Leftrightarrow \psi)$	for	$(\varphi \Rightarrow \psi) \wedge (\psi \Rightarrow \varphi)$
$(\exists\, x \,.\, \varphi)$	for	$\neg(\forall\, x \,.\, \neg\varphi)$
$(t\downarrow)$	for	$\exists\, x \,.\, x = t$
		where $x \in \mathcal{V}$ and x does not occur in t
$(t\uparrow)$	for	$\neg(t\downarrow)$
$(s \simeq t)$	for	$(s\downarrow \vee\, t\downarrow) \Rightarrow s = t$
$\perp$	for	$\mathrm{I}\, x \,.\, x \neq x$ where $x \in \mathcal{V}$
if(φ, s, t)	for	$\mathrm{I}\, x \,.\, (\varphi \Rightarrow x = s) \wedge (\neg\varphi \Rightarrow x = t)$
		where $x \in \mathcal{V}$ and
		x does not occur in φ, s, or t

Some Simple Examples

$$|x| \simeq \mathsf{if}(x < 0, -x, x)$$

$$f(x, y) \simeq \sqrt{\frac{x+y}{x-y}}$$

$$x/y \simeq (\mathrm{I}\, z \;.\; x = y * z)$$

Semantics of PFOL: Models

- A **model** for a PFOL language L is a pair $M = (D, I)$ where D is a nonempty domain and I is a total function on $\mathcal{C} \cup \mathcal{F} \cup \mathcal{P}$ such that:
  - If $c \in \mathcal{C}$, then $I(c) \in D$.
  - If $f \in \mathcal{F}$ is n-ary, then $I(f)$ is a **partial** function from $D \times \cdots \times D$ (n times) to D.
  - If $p \in \mathcal{P}$ is n-ary, then $I(p)$ is a **total** function from $D \times \cdots \times D$ (n times) to $\{\mathsf{T}, \mathsf{F}\}$ (domain of truth values). $I(=)$ is the identity relation on D.

- A **variable assignment** into M is a total function from $\mathcal{V}$ to D.

Semantics of PFOL: Valuation (1)

- Let $M = (D, I)$ be a model of a language $L = (\mathcal{V}, \mathcal{C}, \mathcal{F}, \mathcal{P})$ of PFOL

- The **valuation function** for M is a binary function V^M such that satisfies the following conditions for all variable assignments A and all terms and formulas of L:

  1. If $t \in \mathcal{V}$, then $V^M_A(t) = A(t)$.
  2. If $t \in \mathcal{C}$, then $V^M_A(t) = I(t)$.
  3. Let $t = \mathrm{I}\, x \,.\, \varphi$. If there is a unique $d \in D$ such that

     $$V^M_{A[x \mapsto d]}(\varphi) = \mathsf{T},$$

     then $V^M_A(t) = d$; otherwise $V^M_A(t)$ is undefined.

Semantics of PFOL: Valuation (2)

4. Let $t = f(t_1, \ldots, t_n)$. If $V_A^M(t_1), \ldots, V_A^M(t_n)$ are defined and $I(f)$ is defined at $\langle V_A^M(t_1), \ldots, V_A^M(t_n) \rangle$, then

   $$V_A^M(t) = I(f)(V_A^M(t_1), \ldots, V_A^M(t_n));$$

   otherwise $V_A^M(t)$ is undefined.

5. Let $\varphi = p(t_1, \ldots, t_n)$. If $V_A^M(t_1), \ldots, V_A^M(t_n)$ are defined, then

   $$V_A^M(\varphi) = I(p)(V_A^M(t_1), \ldots, V_A^M(t_n));$$

   otherwise $V_A^M(\varphi) = \mathsf{F}$.

Semantics of PFOL: Valuation (3)

6. Let $\varphi = \neg\varphi'$. If $V_A^M(\varphi') = \mathsf{F}$, then $V_A^M(\varphi) = \mathsf{T}$; otherwise $V_A^M(\varphi) = \mathsf{F}$.

7. Let $\varphi = \varphi' \Rightarrow \varphi''$. If $V_A^M(\varphi') = \mathsf{T}$ and $V_A^M(\varphi'') = \mathsf{F}$, then $V_A^M(\varphi) = \mathsf{F}$; otherwise $V_A^M(\varphi) = \mathsf{T}$.

8. Let $\varphi = \forall\, x \;.\; \varphi'$. If $V_{A[x \mapsto d]}^M(\varphi') = \mathsf{T}$ for all $d \in D$, then $V_A^M(\varphi) = \mathsf{T}$; otherwise $V_A^M(\varphi) = \mathsf{F}$.

Elimination Theorem

Theorem For every PFOL theory $T = (L, \Gamma)$, there is a FOL theory $T^* = (L^*, \Gamma^*)$ and a translation from each formula φ of L to a formula φ^* of L^* such that

$$T \models_{\text{PFOL}} \varphi \quad \text{iff} \quad T^* \models_{\text{FOL}} \varphi^*.$$

Moreover, $T^* = T$ if L contains no function symbols and Γ contains no occurrences of I, and $\varphi^* = \varphi$ if φ contains no function symbols and no occurrences of I.

Proof ideas:

- n-ary function symbols are replaced by $(n+1)$-ary predicate symbols
- Definite descriptions are eliminated in the same way Russell eliminates them in “On Denoting”

An Axiomatization of PFOL (1)

Axioms schemata:

1. $\varphi \Rightarrow (\psi \Rightarrow \varphi)$

2. $[\varphi \Rightarrow (\psi \Rightarrow \theta)] \Rightarrow [(\varphi \Rightarrow \psi) \Rightarrow (\varphi \Rightarrow \theta)]$

3. $(\neg\varphi \Rightarrow \neg\psi) \Rightarrow (\psi \Rightarrow \varphi)$

4. $(\forall\, x \,.\, \varphi \Rightarrow \psi) \Rightarrow (\varphi \Rightarrow \forall\, x \,.\, \psi)$ where x is not free in φ

5. $[(\forall\, x \,.\, \varphi) \wedge t\!\downarrow] \Rightarrow \varphi[x \mapsto t]$ where t is free for x in φ

6. $\forall\, x \,.\, x = x$

An Axiomatization of PFOL (2)

7. $s \simeq t \Rightarrow (\varphi \Rightarrow \varphi^*)$ where φ^* is the result of replacing one occurrence of s in φ by an occurrence of t, provided that the occurrence of s is not a variable immediately after $\forall$ or I

8. $x\!\downarrow$ where $x \in \mathcal{V}$

9. $a\!\downarrow$ where $a \in \mathcal{C}$

10. $(\mathrm{I}\, x \,.\, \varphi)\!\downarrow \Leftrightarrow [\exists\, x \,.\, \varphi \wedge (\forall\, y \,.\, \varphi[x \mapsto y] \Rightarrow y = x)]$
    where y does not occur in φ

11. $(\mathrm{I}\, x \,.\, \varphi)\!\downarrow \Rightarrow \varphi[x \mapsto (\mathrm{I}\, x \,.\, \varphi)]$
    where $(\mathrm{I}\, x \,.\, \varphi)$ is free for x in φ

An Axiomatization of PFOL (3)

12. $[t_1\uparrow \vee \cdots \vee t_n\uparrow] \Rightarrow f(t_1, \ldots, t_n)\uparrow$ where $f \in \mathcal{F}$ is n-ary

13. $[t_1\uparrow \vee \cdots \vee t_n\uparrow] \Rightarrow \neg p(t_1, \ldots, t_n)$ where $p \in \mathcal{P}$ is n-ary

Rules of inference:

Modus Ponens: From $\varphi \Rightarrow \psi$ and φ infer ψ

Generalization: From φ infer $\forall x \,.\, \varphi$

Partial Logics

- PFOL, Partial First-Order Logic
  - Version of first-order logic
  - W. Farmer, J. Guttman, *Studia Logica*, 2000
- LUTINS
  - Version of Church's simple theory of types
  - Logic of the IMPS Interactive Mathematical Proof System
  - W. Farmer, *J. Symbolic Logic*, 1990
- BESTT, a Basic Extended Simple Type Theory
  - Extended version of Church's simple theory of types
  - W. Farmer, 2001
- STMM, a Set Theory for Mechanized Mathematics
  - Version of NBG set theory
  - W. Farmer, *J. Automated Reasoning*, 2001

Conclusion

- The traditional approach to partial functions and undefinedness can be formalized in FOL and other traditional logics without sacrificing the underlying intuition and semantics

- The new machinery allows one to reason about undefined terms, partial functions, and definite descriptions is a natural and direct way

- IMPS demonstrates that this kind of machinery can be effectively implemented

- The ideas of PFOL should be:
  - Incorporated into practice-oriented logics
  - Taught to all engineers and scientists