Formal Proofs for Undergraduates

Flávio L. C. de Moura[1]

Departamento de Ciência da Computação
Universidade de Brasília

Seminário Permanente "Lógica no Avião"

April 17, 2017

[1] joint work with Mauricio Ayala-Rincón

Why should Computer Scientists/Mathematicians study (Computational) Logic?

- Therac-25
- Pentium FDIV
- Ariane 5

Formal Verification in CS

- There is an explosion of interest in logic with the necessity to prove programs correct
- Nowadays correctness is required not only for critical systems
- The use of proof assistants for formal verification is becoming a standard technology in computer science

Formal Verification in Mathematics

- Computers are more and more indispensable for checking large proofs
  - Kepler's conjecture (1611)
    - The Flyspeck project 2003-2014
  - Four color theorem (1852)
    - Formalized in Coq by Georges Gonthier (2005)
  - The Feit-Thompson Theorem (Odd-Order Theorem) (1963)
    - Formalized in Coq by Georges Gonthier (2012)

Computational Logic

- Our goal:
  - Provide evidence of applications of logic to interesting problems in both Computer Science and Mathematics
  - Logic as the cornerstone of several applications in Computer Science
- Our approach:
  - Teach logic with focus on deduction:
    - Natural Deduction
    - Sequent calculus
    - Computer-assisted proofs

Course Structure

- Induction principles (weak/incomplete, strong/complete, structural)
- Propositional Logic
  - Natural Deduction
    - Intuitionistic and Classical Logics
    - Correctness and Completeness of Classical Logic
- Predicate Logic
  - Natural deduction (ND)
  - Sequent calculus (SC)
    - Intuitionistic and Classical Logics
    - Equivalence between ND and SC
    - Correctness and Completeness of Classical Predicate Logic
- Formalization project in PVS
  - Correctness of algorithms
  - GCD
  - Sorting algorithms
  - Rewriting Theory

Natural Deduction (Intuitionistic Logic)

introduction rules	elimination rules
$\dfrac{\varphi \quad \psi}{\varphi \wedge \psi}\ (\wedge_i)$	$\dfrac{\varphi \wedge \psi}{\varphi}\ (\wedge_e)$
$\dfrac{\varphi}{\varphi \vee \psi}\ (\vee_i)$	$\dfrac{\varphi \vee \psi \qquad \begin{matrix}[\varphi]^u \\ \vdots \\ \chi\end{matrix} \qquad \begin{matrix}[\psi]^v \\ \vdots \\ \chi\end{matrix}}{\chi}\ (\vee_e)\ u, v$
$\dfrac{\begin{matrix}[\varphi]^u \\ \vdots \\ \psi\end{matrix}}{\varphi \rightarrow \psi}\ (\rightarrow_i) u$	$\dfrac{\varphi \quad \varphi \rightarrow \psi}{\psi}\ (\rightarrow_e)$
$\dfrac{\begin{matrix}[\varphi]^u \\ \vdots \\ \bot\end{matrix}}{\neg\varphi}\ (\neg_i)\ u$	$\dfrac{\varphi \quad \neg\varphi}{\bot}\ (\neg_e)$
	$\dfrac{\bot}{\varphi}\ (\bot_e)$

Natural Deduction (Intuitionistic Logic)

introduction rules	elimination rules
$$\frac{\varphi[x/x_0]}{\forall_x \varphi}\ (\forall_i)$$ where x_0 cannot occur free in any open assumption.	$$\frac{\forall_x \varphi}{\varphi[x/t]}\ (\forall_e)$$
$$\frac{\varphi[x/t]}{\exists_x \varphi}\ (\exists_i)$$	$$\frac{\exists_x \varphi \qquad \begin{matrix}[\varphi[x/x_0]]^u \\ \vdots \\ \chi\end{matrix}}{\chi}\ (\exists_e)\ u$$ where x_0 cannot occur free in any open assumption on the right and in χ.

- Classical logic can be obtained, from intuitionistic logic, by adding one of the following rules:

$$\dfrac{\begin{array}{c}[\neg\phi]^u \\ \vdots \\ \bot\end{array}}{\phi}\ (\text{PBC})\text{u} \qquad \dfrac{}{\phi \vee \neg\phi}\ (\text{LEM})$$

$$\dfrac{\neg\neg\phi}{\phi}\ (\neg\neg_e) \qquad \dfrac{}{((\phi \to \psi) \to \phi) \to \phi}\ (\text{PL})$$

Natural Deduction

Example ($\vdash \varphi \vee \neg\varphi$)

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[\varphi]^v}{\varphi \vee \neg\varphi}\,(\vee_i) \qquad [\neg(\varphi \vee \neg\varphi)]^u}{\bot}\,(\neg_e)}{\neg\varphi}\,(\neg_i)\ ,v}{\varphi \vee \neg\varphi}\,(\vee_i) \qquad [\neg(\varphi \vee \neg\varphi)]^u}{\bot}\,(\neg_e)}{\varphi \vee \neg\varphi}\,(\mathrm{PBC})\mathrm{u}$$

Contextualized example

Example

Prove that there exists irrational numbers x and y such x^y is rational.

Proof.

We consider 2 cases:

1. If $\sqrt{2}^{\sqrt{2}}$ is rational then take $x = y = \sqrt{2}$ and we are done.
2. If $\sqrt{2}^{\sqrt{2}}$ is not rational, i.e., if $\sqrt{2}^{\sqrt{2}}$ is irrational then take $x = \sqrt{2}^{\sqrt{2}}$ and $y = \sqrt{2}$, and we are done since $(\sqrt{2}^{\sqrt{2}})^{\sqrt{2}} = (\sqrt{2})^{\sqrt{2}\cdot\sqrt{2}} = (\sqrt{2})^2 = 2$.

□

Contextualized example in Natural Deduction

$$\nabla : \dfrac{\dfrac{\dfrac{\neg R(\sqrt{2}) \quad \neg R(\sqrt{2})}{\neg R(\sqrt{2}) \wedge \neg R(\sqrt{2})}\,(\wedge_i) \qquad [R(\sqrt{2}^{\sqrt{2}})]^{a_2}}{\neg R(\sqrt{2}) \wedge \neg R(\sqrt{2}) \wedge R(\sqrt{2}^{\sqrt{2}})}\,(\wedge_i)}{\exists x \exists y(\neg R(x) \wedge \neg R(y) \wedge R(x^y))}\,(\exists_i)^2$$

$$\dfrac{(\text{LEM}) \ \neg \mathrm{R}(\sqrt{2}^{\sqrt{2}}) \vee \mathrm{R}(\sqrt{2}^{\sqrt{2}}) \qquad \dfrac{\dfrac{[\neg R(\sqrt{2}^{\sqrt{2}})]^{a_1} \quad \dfrac{\neg R(\sqrt{2}) \quad R((\sqrt{2}^{\sqrt{2}})^{\sqrt{2}})}{\neg R(\sqrt{2}) \wedge R((\sqrt{2}^{\sqrt{2}})^{\sqrt{2}})}\,(\wedge_i)}{\neg R(\sqrt{2}^{\sqrt{2}}) \wedge \neg R(\sqrt{2}) \wedge R((\sqrt{2}^{\sqrt{2}})^{\sqrt{2}})}\,(\wedge_i)}{\exists x \exists y(\neg R(x) \wedge \neg R(y) \wedge R(x^y))}\,(\exists_i)^2 \qquad \nabla}{\exists x \exists y(\neg R(x) \wedge \neg R(y) \wedge R(x^y))}\,(\vee_e)\ a_1, a_2$$

Sequent Calculus

Left Rules	Right Rules
Axioms:	
$\Gamma, \varphi \Rightarrow \varphi, \Delta \quad (\mathrm{Ax})$	$\bot, \Gamma \Rightarrow \Delta \quad (\mathrm{L}_\bot)$
Structural Rules:	
$\dfrac{\Gamma \Rightarrow \Delta}{\varphi, \Gamma \Rightarrow \Delta} \quad (\mathrm{L}Weakening)$	$\dfrac{\Gamma \Rightarrow \Delta}{\Gamma \Rightarrow \Delta, \varphi} \quad (\mathrm{R}Weakening)$
$\dfrac{\varphi, \varphi, \Gamma \Rightarrow \Delta}{\varphi, \Gamma \Rightarrow \Delta} \quad (\mathrm{L}Contraction)$	$\dfrac{\Gamma \Rightarrow \Delta, \varphi, \varphi}{\Gamma \Rightarrow \Delta, \varphi} \quad (\mathrm{R}Contraction)$
Logical Rules:	
$\dfrac{\varphi_{i \in \{1,2\}}, \Gamma \Rightarrow \Delta}{\varphi_1 \wedge \varphi_2, \Gamma \Rightarrow \Delta} \quad (\mathrm{L}_\wedge)$	$\dfrac{\Gamma \Rightarrow \Delta, \varphi \quad \Gamma \Rightarrow \Delta, \psi}{\Gamma \Rightarrow \Delta, \varphi \wedge \psi} \quad (\mathrm{R}_\wedge)$
$\dfrac{\varphi, \Gamma \Rightarrow \Delta \quad \psi, \Gamma \Rightarrow \Delta}{\varphi \vee \psi, \Gamma \Rightarrow \Delta} \quad (\mathrm{L}_\vee)$	$\dfrac{\Gamma \Rightarrow \Delta, \varphi_{i \in \{1,2\}}}{\Gamma \Rightarrow \Delta, \varphi_1 \vee \varphi_2} \quad (\mathrm{R}_\vee)$
$\dfrac{\Gamma \Rightarrow \Delta, \varphi \quad \psi, \Gamma \Rightarrow \Delta}{\varphi \rightarrow \psi, \Gamma \Rightarrow \Delta} \quad (\mathrm{L}_\rightarrow)$	$\dfrac{\varphi, \Gamma \Rightarrow \Delta, \psi}{\Gamma \Rightarrow \Delta, \varphi \rightarrow \psi} \quad (\mathrm{R}_\rightarrow)$
$\dfrac{\varphi[x/t], \Gamma \Rightarrow \Delta}{\forall_x \varphi, \Gamma \Rightarrow \Delta} \quad (\mathrm{L}_\forall)$	$\dfrac{\Gamma \Rightarrow \Delta, \varphi[x/y]}{\Gamma \Rightarrow \Delta, \forall_x \varphi} \quad (\mathrm{R}_\forall), \quad y \notin FV(\Gamma, \Delta)$
$\dfrac{\varphi[x/y], \Gamma \Rightarrow \Delta}{\exists_x \varphi, \Gamma \Rightarrow \Delta} \quad (\mathrm{L}_\exists), \quad y \notin FV(\Gamma, \Delta)$	$\dfrac{\Gamma \Rightarrow \Delta, \varphi[x/t]}{\Gamma \Rightarrow \Delta, \exists_x \varphi} \quad (\mathrm{R}_\exists)$

$$\frac{\Gamma \Rightarrow \Delta, \varphi \quad \varphi, \Gamma' \Rightarrow \Delta'}{\Gamma\Gamma' \Rightarrow \Delta\Delta'} \quad (\mathrm{Cut})$$

Example ($\vdash \varphi \vee (\varphi \to \bot)$)

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\varphi \Longrightarrow \varphi, \bot}\ (\mathrm{Ax})}{\Longrightarrow \varphi, (\varphi \to \bot)}\ (\mathrm{R}_{\to})}{\Longrightarrow \varphi \vee (\varphi \to \bot), (\varphi \to \bot)}\ (\mathrm{R}_{\vee})}{\Longrightarrow \varphi \vee (\varphi \to \bot), \varphi \vee (\varphi \to \bot)}\ (\mathrm{R}_{\vee})}{\Longrightarrow \varphi \vee (\varphi \to \bot)}\ (\mathrm{RC})$$

Contextualized example in Sequent Calculus

$$\nabla_1 : \quad \dfrac{\dfrac{\dfrac{}{\neg R(\sqrt{2}^{\sqrt{2}}) \Rightarrow \neg R(\sqrt{2}^{\sqrt{2}})}\,(\text{Ax}) \quad \dfrac{\dfrac{\dfrac{\Rightarrow \neg R(\sqrt{2})}{\neg R(\sqrt{2}^{\sqrt{2}}) \Rightarrow \neg R(\sqrt{2})}\,(\text{LW}) \quad \dfrac{\Rightarrow R((\sqrt{2}^{\sqrt{2}})^{\sqrt{2}})}{\neg R(\sqrt{2}^{\sqrt{2}}) \Rightarrow R((\sqrt{2}^{\sqrt{2}})^{\sqrt{2}})}\,(\text{LW})}{\neg R(\sqrt{2}^{\sqrt{2}}) \Rightarrow \neg R(\sqrt{2}) \wedge R((\sqrt{2}^{\sqrt{2}})^{\sqrt{2}})}\,(\text{R}_\wedge)}{}}{\neg R(\sqrt{2}^{\sqrt{2}}) \Rightarrow \neg R(\sqrt{2}^{\sqrt{2}}) \wedge \neg R(\sqrt{2}) \wedge R((\sqrt{2}^{\sqrt{2}})^{\sqrt{2}})}\,(\text{R}_\wedge)}{\neg R(\sqrt{2}^{\sqrt{2}}) \Rightarrow \exists x \exists y(\neg R(x) \wedge \neg R(y) \wedge R(x^y))}\,(\text{R}_\exists)^2$$

$$\nabla_2 : \quad \dfrac{\dfrac{\dfrac{\dfrac{\Rightarrow \neg R(\sqrt{2})}{R(\sqrt{2}^{\sqrt{2}}) \Rightarrow \neg R(\sqrt{2})}\,(\text{LW}) \quad \dfrac{\Rightarrow \neg R(\sqrt{2})}{R(\sqrt{2}^{\sqrt{2}}) \Rightarrow \neg R(\sqrt{2})}\,(\text{LW})}{R(\sqrt{2}^{\sqrt{2}}) \Rightarrow \neg R(\sqrt{2}) \wedge \neg R(\sqrt{2})}\,(\text{R}_\wedge) \quad \dfrac{}{R(\sqrt{2}^{\sqrt{2}}) \Rightarrow R(\sqrt{2}^{\sqrt{2}})}\,(\text{Ax})}{R(\sqrt{2}^{\sqrt{2}}) \Rightarrow \neg R(\sqrt{2}) \wedge \neg R(\sqrt{2}) \wedge R(\sqrt{2}^{\sqrt{2}})}\,(\text{R}_\wedge)}{R(\sqrt{2}^{\sqrt{2}}) \Rightarrow \exists x \exists y(\neg R(x) \wedge \neg R(y) \wedge R(x^y))}\,(\text{R}_\exists)^2$$

$$\dfrac{(\text{LEM}) \quad \Rightarrow \neg \mathrm{R}(\sqrt{2}^{\sqrt{2}}) \vee \mathrm{R}(\sqrt{2}^{\sqrt{2}}) \qquad \dfrac{\nabla_1 \qquad \nabla_2}{\neg R(\sqrt{2}^{\sqrt{2}}) \vee R(\sqrt{2}^{\sqrt{2}}) \Rightarrow \exists x \exists y(\neg R(x) \wedge \neg R(y) \wedge R(x^y))}\,(\text{L}_\vee)}{\Rightarrow \exists x \exists y(\neg R(x) \wedge \neg R(y) \wedge R(x^y))}\,(\text{Cut})$$

Equivalence between ND and SC

Theorem (Natural vs deduction *à la* Gentzen for the classical logic)

One has that for the classical Gentzen and natural calculus

$$\vdash_G \Gamma \Rightarrow \varphi \textit{ if and only if } \Gamma \vdash_N \varphi$$

Computer-Assisted Proofs

- The goal is to use (first-order) logic to solve interesting problems in both Computer Science and Mathematics, but
    - Not by doing logic programming, but
    - Proving properties of algorithms or mathematical theories

Example

Available examples include:

1. Formalization of GCD function
2. Correctness of sorting algorithms: insertion sort, merge sort, bubble sort, heap sort, etc
3. Formalization of rewriting theory: Confluence and Newman's Lemma

Prototype Verfication System - PVS

- Proof assistant developed by SRI International Computer Science Laboratory
  - Based on a higher-order logic
  - Type system based on Church's simple theory of types augmented with subtypes and dependent types
  - Good automation tools (good option as a first proof assistant)
  - Based on sequent calculus:

Proof command	Rules
(flatten)	$(R_\vee)$, $(L_\wedge)$, $(R_\rightarrow)$
(split)	$(L_\vee)$, $(R_\wedge)$, $(L_\rightarrow)$
(inst)	$(R_\forall)$, $(L_\exists)$
(skolem)	$(L_\forall)$, $(R_\exists)$
(case), (lemma)	(Cut)
(copy)	(RC), (LC)
(hide)	(RW), (LW)

Example (Summing up the natural numbers from 0 to n)

$$\sum_{i=0}^{n} i = \frac{n(n+1)}{2}, \forall n \in \mathbb{N}$$

- **(IB)** $\sum_{i=0}^{0} i = 0 = \frac{0(0+1)}{2}$
- **(IS)** $\sum_{i=0}^{n} i = n + \sum_{i=0}^{n-1} i \overset{IH}{=} n + \frac{(n-1)n}{2} = \frac{2n+(n-1)n}{2} = \frac{n(n+1)}{2}$

In PVS ...

Example (Summing up the first n odd numbers)

$$\sum_{i=1}^{n}(2i-1) = n^2, \forall n > 0$$

- **(IB)** $\sum_{i=1}^{1}(2i-1) = 1^2$
- **(IS)** $\sum_{i=1}^{n}(2i-1) = (2n-1) + \sum_{i=1}^{n-1}(2i-1) \stackrel{IH}{=} (2n-1) + (n-1)^2 = n^2$

In PVS ...

Example (Correctness of sorting algorithms - Insertion sort)

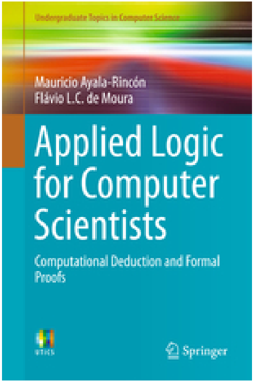
```
insert (x, l): RECURSIVE list[T] =
        IF null?(l) THEN cons(x,null)
        ELSIF x <= car(l) THEN cons(x,l)
        ELSE cons(car(l), insert(x,cdr(l)))
ENDIF MEASURE length(l)

insertion_sort(l): RECURSIVE list[T] =
IF null?(l) THEN null ELSE
insert(car(l), insertion_sort(cdr(l)))
ENDIF MEASURE length(l)

insertion_sort_works : LEMMA
FORALL (l: list[T]): is_sorted?(insertion_sort(l)) AND
                     permutations(l, insertion_sort(l))
```

In PVS ...

Conclusions

- Computational Logic is intensively used in formal methods
- Computational Logic with focus on deduction is a good way to explore student's knowledge to prove the correctness of his/her programs
- The relevance and importance of formalized proofs are no longer restricted to critical systems
- Proofs of interesting (both simple and complex) mathematical and/or computational properties can be built on a relatively small set of basic deductive rules
- The choice of the proof assistant is not important
  - Coq
  - Isabelle/HOL
  - PVS
  - and many others!

Thank you!



Companion website: `logic4CS.cic.unb.br`