Soundness and Completeness of Implicational Logic

Asta Halkjær From   Jørgen Villadsen

October 27, 2022

Contents

Abstract

This work is a formalization of soundness and completeness of the Bernays-Tarski axiom system for classical implicational logic. The completeness proof is constructive following the approach by László Kalmár, Elliott Mendelson and others. The result can be extended to full classical propositional logic by uncommenting a few lines for falsehood.

1 Formalization of the Bernays-Tarski Axiom System for Classical Implicational Logic

1.1 Syntax, Semantics and Axiom System

theory *Implicational-Logic* **imports** *Main* **begin**

datatype *form* =

Pro nat (‹·›) |
Imp form form (**infixr** ‹→› *55*)

primrec *semantics* (**infix** ‹$\models$› *50*) **where**

‹$I \models \cdot\ n = I\ n$› |
‹$I \models p \rightarrow q = (I \models p \longrightarrow I \models q)$›

inductive *Ax* (‹$\vdash$ -› *50*) **where**

Simp: ‹$\vdash p \rightarrow q \rightarrow p$› |
Tran: ‹$\vdash (p \rightarrow q) \rightarrow (q \rightarrow r) \rightarrow p \rightarrow r$› |
MP: ‹$\vdash p \rightarrow q \Longrightarrow \vdash p \Longrightarrow \vdash q$› |
PR: ‹$\vdash (p \rightarrow q) \rightarrow p \Longrightarrow \vdash p$›

1.2 Soundness and Derived Formulas

theorem *soundness*: ‹$\vdash p \Longrightarrow I \models p$›
⟨*proof*⟩

lemma *Swap*: ‹$\vdash (p \rightarrow q \rightarrow r) \rightarrow q \rightarrow p \rightarrow r$›
⟨*proof*⟩

lemma *Peirce*: ‹$\vdash ((p \rightarrow q) \rightarrow p) \rightarrow p$›
⟨*proof*⟩

lemma *Hilbert*: ‹$\vdash (p \rightarrow p \rightarrow q) \rightarrow p \rightarrow q$›
⟨*proof*⟩

lemma *Id*: ‹$\vdash p \rightarrow p$›
⟨*proof*⟩

lemma *Tran'*: ‹$\vdash (q \rightarrow r) \rightarrow (p \rightarrow q) \rightarrow p \rightarrow r$›
⟨*proof*⟩

lemma *Frege*: ‹$\vdash (p \rightarrow q \rightarrow r) \rightarrow (p \rightarrow q) \rightarrow p \rightarrow r$›
⟨*proof*⟩

lemma *Imp1*: ‹$\vdash (q \rightarrow s) \rightarrow ((q \rightarrow r) \rightarrow s) \rightarrow s$›
⟨*proof*⟩

lemma *Imp2*: ‹$\vdash ((r \rightarrow s) \rightarrow s) \rightarrow ((q \rightarrow r) \rightarrow s) \rightarrow s$›
⟨*proof*⟩

lemma *Imp3*: ‹$\vdash ((q \rightarrow s) \rightarrow s) \rightarrow (r \rightarrow s) \rightarrow (q \rightarrow r) \rightarrow s$›
⟨*proof*⟩

1.3 Completeness and Main Theorem

fun *pros* **where**
‹$pros\ (p \rightarrow q) = remdups\ (pros\ p\ @\ pros\ q)$› |
‹$pros\ p = (case\ p\ of\ (\cdot\ n) \Rightarrow [n] \mid \text{-} \Rightarrow [])$›

lemma *distinct-pros*: ‹$distinct\ (pros\ p)$›
⟨*proof*⟩

primrec *imply* (**infixr** ‹⇝› *56*) **where**
‹$[] \leadsto q = q$› |
‹$p\ \#\ ps \leadsto q = p \to ps \leadsto q$›

lemma *imply-append*: ‹$ps\ @\ qs \leadsto r = ps \leadsto qs \leadsto r$›
⟨*proof*⟩

abbreviation *Ax-assms* (**infix** ‹⊢› *50*) **where** ‹$ps \vdash q \equiv \vdash ps \leadsto q$›

lemma *imply-Cons*: ‹$ps \vdash q \Longrightarrow p\ \#\ ps \vdash q$›
⟨*proof*⟩

lemma *imply-head*: ‹$p\ \#\ ps \vdash p$›
⟨*proof*⟩

lemma *imply-mem*: ‹$p \in set\ ps \Longrightarrow ps \vdash p$›
⟨*proof*⟩

lemma *imply-MP*: ‹$\vdash ps \leadsto (p \to q) \to ps \leadsto p \to ps \leadsto q$›
⟨*proof*⟩

lemma *MP'*: ‹$ps \vdash p \to q \Longrightarrow ps \vdash p \Longrightarrow ps \vdash q$›
⟨*proof*⟩

lemma *imply-swap-append*: ‹$ps\ @\ qs \vdash r \Longrightarrow qs\ @\ ps \vdash r$›
⟨*proof*⟩

lemma *imply-deduct*: ‹$p\ \#\ ps \vdash q \Longrightarrow ps \vdash p \to q$›
⟨*proof*⟩

lemma *add-imply* [*simp*]: ‹$\vdash p \Longrightarrow ps \vdash p$›
⟨*proof*⟩

lemma *imply-weaken*: ‹$ps \vdash p \Longrightarrow set\ ps \subseteq set\ ps' \Longrightarrow ps' \vdash p$›
⟨*proof*⟩

abbreviation ‹$lift\ t\ s\ p \equiv if\ t\ then\ (p \to s) \to s\ else\ p \to s$›

abbreviation ‹$lifts\ I\ s \equiv map\ (\lambda n.\ lift\ (I\ n)\ s\ (\cdot\ n))$›

lemma *lifts-weaken*: ‹$lifts\ I\ s\ l \vdash p \Longrightarrow set\ l \subseteq set\ l' \Longrightarrow lifts\ I\ s\ l' \vdash p$›
⟨*proof*⟩

lemma *lifts-pros-lift*: ‹$lifts\ I\ s\ (pros\ p) \vdash lift\ (I \models p)\ s\ p$›
⟨*proof*⟩

lemma *lifts-pros*: ‹$I \models p \Longrightarrow$ *lifts* I p (*pros* p) $\vdash p$›
⟨*proof*⟩

theorem *completeness*: ‹$\forall I.\ I \models p \Longrightarrow\ \vdash p$›
⟨*proof*⟩

theorem *main*: ‹$(\vdash p) = (\forall I.\ I \models p)$›
⟨*proof*⟩

1.4 Reference

Wikipedia https://en.wikipedia.org/wiki/Implicational_propositional_calculus
July 2022

end

2 Formalization of Łukasiewicz's Axiom System from 1924 for Classical Propositional Logic

2.1 Syntax, Semantics and Axiom System

theory *Implicational-Logic-Appendix* **imports** *Main* **begin**

datatype *form* =
Pro nat (‹·›) |
Neg form (‹∼›) |
Imp form form (**infixr** ‹→› *55*)

primrec *semantics* (**infix** ‹⊨› *50*) **where**
‹$I \models \cdot\ n = I\ n$› |
‹$I \models \sim p = (\neg\ I \models p)$› |
‹$I \models p \rightarrow q = (I \models p \longrightarrow I \models q)$›

inductive *Ax* (‹⊢ -› *50*) **where**
01: ‹$\vdash (p \rightarrow q) \rightarrow (q \rightarrow r) \rightarrow p \rightarrow r$› |
02: ‹$\vdash (\sim p \rightarrow p) \rightarrow p$› |
03: ‹$\vdash p \rightarrow \sim p \rightarrow q$› |
MP: ‹$\vdash p \rightarrow q \Longrightarrow\ \vdash p \Longrightarrow\ \vdash q$›

2.2 Soundness and Derived Formulas

theorem *soundness*: ‹$\vdash p \Longrightarrow I \models p$›
⟨*proof*⟩

lemma *04*: ‹$\vdash (((q \rightarrow r) \rightarrow p \rightarrow r) \rightarrow s) \rightarrow (p \rightarrow q) \rightarrow s$›
⟨*proof*⟩

lemma *05*: ‹$\vdash (p \rightarrow q \rightarrow r) \rightarrow (s \rightarrow q) \rightarrow p \rightarrow s \rightarrow r$›

⟨*proof*⟩

lemma *06*: ‹$\vdash (p \to q) \to ((p \to r) \to s) \to (q \to r) \to s$›
⟨*proof*⟩

lemma *07*: ‹$\vdash (t \to (p \to r) \to s) \to (p \to q) \to t \to (q \to r) \to s$›
⟨*proof*⟩

lemma *09*: ‹$\vdash ((\sim p \to q) \to r) \to p \to r$›
⟨*proof*⟩

lemma *10*: ‹$\vdash p \to ((\sim p \to p) \to p) \to (q \to p) \to p$›
⟨*proof*⟩

lemma *11*: ‹$\vdash (q \to (\sim p \to p) \to p) \to (\sim p \to p) \to p$›
⟨*proof*⟩

lemma *12*: ‹$\vdash t \to (\sim p \to p) \to p$›
⟨*proof*⟩

lemma *13*: ‹$\vdash (\sim p \to q) \to t \to (q \to p) \to p$›
⟨*proof*⟩

lemma *14*: ‹$\vdash ((t \to (q \to p) \to p) \to r) \to (\sim p \to q) \to r$›
⟨*proof*⟩

lemma *15*: ‹$\vdash (\sim p \to q) \to (q \to p) \to p$›
⟨*proof*⟩

lemma *16*: ‹$\vdash p \to p$›
⟨*proof*⟩

lemma *17*: ‹$\vdash p \to (q \to p) \to p$›
⟨*proof*⟩

lemma *18*: ‹$\vdash q \to p \to q$›
⟨*proof*⟩

lemma *19*: ‹$\vdash ((p \to q) \to r) \to q \to r$›
⟨*proof*⟩

lemma *20*: ‹$\vdash p \to (p \to q) \to q$›
⟨*proof*⟩

lemma *21*: ‹$\vdash (p \to q \to r) \to q \to p \to r$›
⟨*proof*⟩

lemma *22*: ‹$\vdash (q \to r) \to (p \to q) \to p \to r$›
⟨*proof*⟩

lemma *23*: ‹$\vdash ((q \to p \to r) \to s) \to (p \to q \to r) \to s$›
⟨*proof*⟩

lemma *24*: ‹$\vdash ((p \to q) \to p) \to p$›
⟨*proof*⟩

lemma *25*: ‹$\vdash ((p \to r) \to s) \to (p \to q) \to (q \to r) \to s$›
⟨*proof*⟩

lemma *26*: ‹$\vdash ((p \to q) \to r) \to (r \to p) \to p$›
⟨*proof*⟩

lemma *28*: ‹$\vdash (((r \to p) \to p) \to s) \to ((p \to q) \to r) \to s$›
⟨*proof*⟩

lemma *29*: ‹$\vdash ((p \to q) \to r) \to (p \to r) \to r$›
⟨*proof*⟩

lemma *31*: ‹$\vdash (p \to s) \to ((p \to q) \to r) \to (s \to r) \to r$›
⟨*proof*⟩

lemma *32*: ‹$\vdash ((p \to q) \to r) \to (p \to s) \to (s \to r) \to r$›
⟨*proof*⟩

lemma *33*: ‹$\vdash (p \to s) \to (s \to q \to p \to r) \to q \to p \to r$›
⟨*proof*⟩

lemma *34*: ‹$\vdash (s \to q \to p \to r) \to (p \to s) \to q \to p \to r$›
⟨*proof*⟩

lemma *35*: ‹$\vdash (p \to q \to r) \to (p \to q) \to p \to r$›
⟨*proof*⟩

lemma *36*: ‹$\vdash \sim p \to p \to q$›
⟨*proof*⟩

lemmas
Tran = *01* **and**
Clavius = *02* **and**
Expl = *03* **and**
Frege′ = *05* **and**
Clavius′ = *15* **and**
Id = *16* **and**
Simp = *18* **and**
Swap = *21* **and**
Tran′ = *22* **and**
Peirce = *24* **and**
Frege = *35* **and**

$Expl' = 36$

lemma *Neg1*: ‹$\vdash (q \to s) \to (\sim q \to s) \to s$›
⟨*proof*⟩

lemma *Neg2*: ‹$\vdash ((q \to s) \to s) \to \sim q \to s$›
⟨*proof*⟩

lemma *Imp1*: ‹$\vdash (q \to s) \to ((q \to r) \to s) \to s$›
⟨*proof*⟩

lemma *Imp2*: ‹$\vdash ((r \to s) \to s) \to ((q \to r) \to s) \to s$›
⟨*proof*⟩

lemma *Imp3*: ‹$\vdash ((q \to s) \to s) \to (r \to s) \to (q \to r) \to s$›
⟨*proof*⟩

2.3 Completeness and Main Theorem

primrec *pros* **where**
‹$pros\ (\cdot\ n) = [n]$› |
‹$pros\ (\sim p) = pros\ p$› |
‹$pros\ (p \to q) = remdups\ (pros\ p\ @\ pros\ q)$›

lemma *distinct-pros*: ‹$distinct\ (pros\ p)$›
⟨*proof*⟩

primrec *imply* (**infixr** ‹$\rightsquigarrow$› *56*) **where**
‹$[] \rightsquigarrow q = q$› |
‹$p\ \#\ ps \rightsquigarrow q = p \to ps \rightsquigarrow q$›

lemma *imply-append*: ‹$ps\ @\ qs \rightsquigarrow r = ps \rightsquigarrow qs \rightsquigarrow r$›
⟨*proof*⟩

abbreviation *Ax-assms* (**infix** ‹$\vdash$› *50*) **where** ‹$ps \vdash q \equiv \vdash ps \rightsquigarrow q$›

lemma *imply-Cons*: ‹$ps \vdash q \Longrightarrow p\ \#\ ps \vdash q$›
⟨*proof*⟩

lemma *imply-head*: ‹$p\ \#\ ps \vdash p$›
⟨*proof*⟩

lemma *imply-mem*: ‹$p \in set\ ps \Longrightarrow ps \vdash p$›
⟨*proof*⟩

lemma *imply-MP*: ‹$\vdash ps \rightsquigarrow (p \to q) \to ps \rightsquigarrow p \to ps \rightsquigarrow q$›
⟨*proof*⟩

lemma *MP'*: ‹$ps \vdash p \to q \Longrightarrow ps \vdash p \Longrightarrow ps \vdash q$›

$\langle proof \rangle$

lemma *imply-swap-append*: ‹ps @ $qs \vdash r \Longrightarrow qs$ @ $ps \vdash r$›
$\langle proof \rangle$

lemma *imply-deduct*: ‹p # $ps \vdash q \Longrightarrow ps \vdash p \rightarrow q$›
$\langle proof \rangle$

lemma *add-imply* [*simp*]: ‹$\vdash p \Longrightarrow ps \vdash p$›
$\langle proof \rangle$

lemma *imply-weaken*: ‹$ps \vdash p \Longrightarrow set\ ps \subseteq set\ ps' \Longrightarrow ps' \vdash p$›
$\langle proof \rangle$

abbreviation ‹$lift\ t\ s\ p \equiv if\ t\ then\ (p \rightarrow s) \rightarrow s\ else\ p \rightarrow s$›

abbreviation ‹$lifts\ I\ s \equiv map\ (\lambda n.\ lift\ (I\ n)\ s\ (\cdot\ n))$›

lemma *lifts-weaken*: ‹$lifts\ I\ s\ l \vdash p \Longrightarrow set\ l \subseteq set\ l' \Longrightarrow lifts\ I\ s\ l' \vdash p$›
$\langle proof \rangle$

lemma *lifts-pros-lift*: ‹$lifts\ I\ s\ (pros\ p) \vdash lift\ (I \models p)\ s\ p$›
$\langle proof \rangle$

lemma *lifts-pros*: ‹$I \models p \Longrightarrow lifts\ I\ p\ (pros\ p) \vdash p$›
$\langle proof \rangle$

theorem *completeness*: ‹$\forall I.\ I \models p \Longrightarrow \vdash p$›
$\langle proof \rangle$

theorem *main*: ‹$(\vdash p) = (\forall I.\ I \models p)$›
$\langle proof \rangle$

2.4 Reference

Numbered lemmas are from Jan Łukasiewicz: Elements of Mathematical Logic (English Tr. 1963)

end