Overview

Basic system

Exploitation and planning rules		
sentence	*as a resource*	*as a goal*
atomic sentence	*none*	IP
negation ¬ φ	CR (if φ not atomic & goal is ⊥)	RAA
conjunction φ ∧ ψ	Ext	Cnj
disjunction φ ∨ ψ	PC	PE
conditional φ → ψ	RC (if goal is ⊥)	CP
universal ∀x θx	UI	UG
existential ∃x θx	PCh	NcP

Rules for closing gaps			
when to close			*rule*
co-aliases	*resources*	*goal*	
	φ	φ	QED
	φ and ¬ φ	⊥	Nc
	any	⊤	ENV
	⊥	*any*	EFQ
τ—υ	*any*	τ = υ	EC
τ—υ	¬ τ = υ	⊥	DC
τ_1—υ_1, …, τ_n—υ_n	$P\tau_1\ldots\tau_n$	$P\upsilon_1\ldots\upsilon_n$	QED=
τ_1—υ_1, …, τ_n—υ_n	$P\tau_1\ldots\tau_n$ $\neg P\upsilon_1\ldots\upsilon_n$	⊥	Nc=

Detachment rules (optional)		
required resources		*rule*
main	*auxiliary*	
¬ (φ ∧ ψ)	φ or ψ	MPT
φ ∨ ψ	$\neg^{\pm}$ φ or $\neg^{\pm}$ ψ	MTP
φ → ψ	φ	MPP
	$\neg^{\pm}$ ψ	MTT

In addition, if the conditions for applying a rule are met except for differences between co-aliases, then the rule can be applied and is notated by adding "="; QED= and Nc= are examples of this.

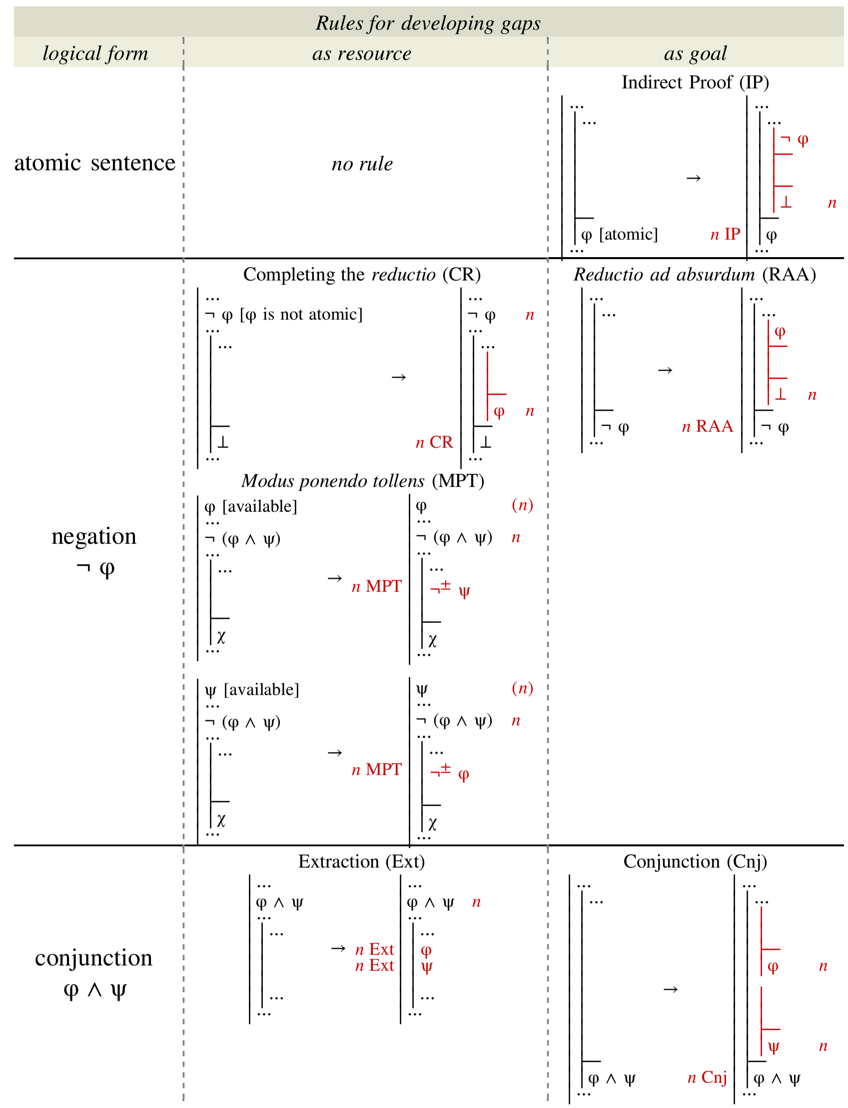
Additional rules

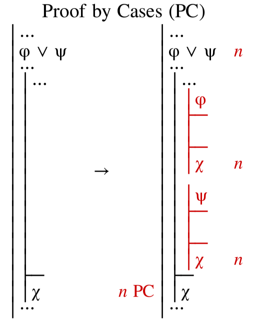
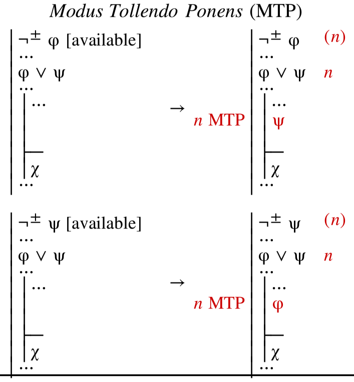
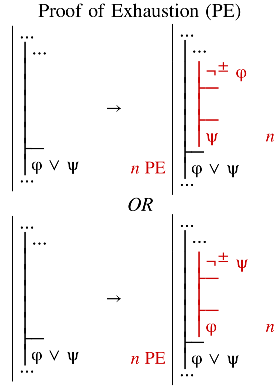
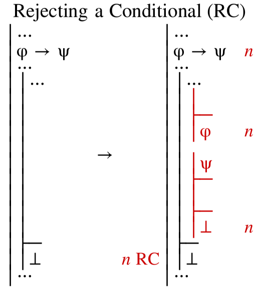
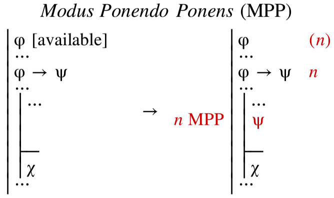
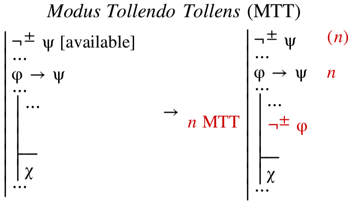
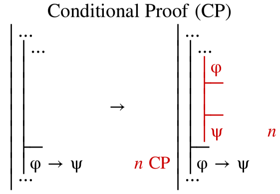
Attachment rules	
added resource	*rule*
φ ∧ ψ	Adj
¬ (φ ∧ ψ)	Wk
φ ∨ ψ	Wk
φ → ψ	Wk
τ = υ	CE
$\theta\upsilon_1\ldots\upsilon_n$	Cng
∃x θx	EG

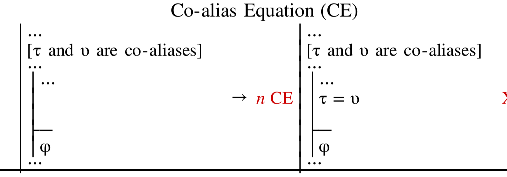
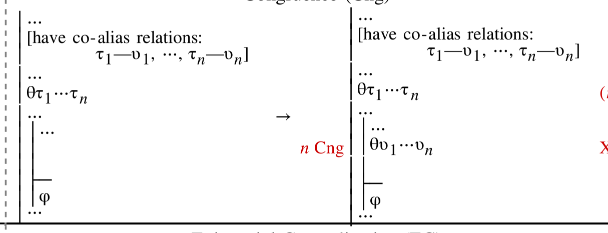
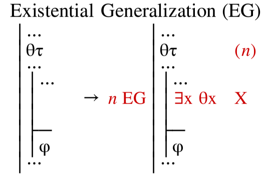
Rule for lemmas	
prerequisite	*rule*
the goal is ⊥	LFR

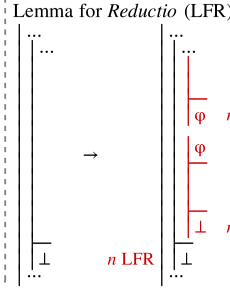
Derivation rules

Basic system

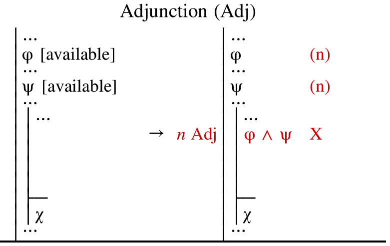
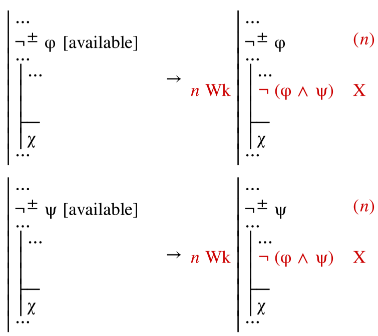
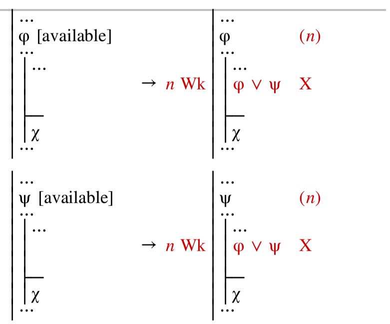
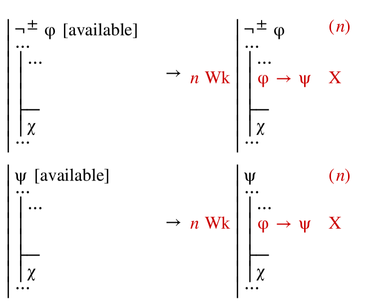


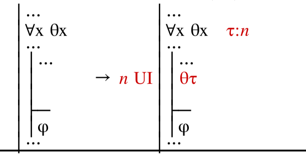
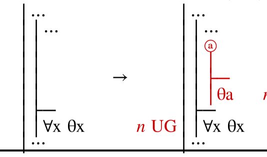
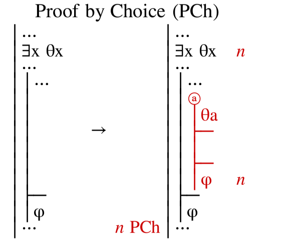
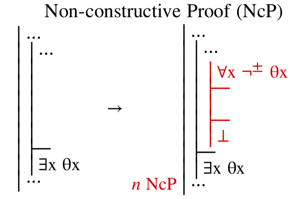
Rules for developing gaps		
logical form	*as resource*	*as goal*
disjunction $\varphi \lor \psi$	Proof by Cases (PC)  *Modus Tollendo Ponens* (MTP) 	Proof of Exhaustion (PE) 
conditional $\varphi \to \psi$	Rejecting a Conditional (RC)  *Modus Ponendo Ponens* (MPP)  *Modus Tollendo Tollens* (MTT) 	Conditional Proof (CP) 

Attachment rules		
what is required	*added resource*	*rule*
τ and υ are co-aliases	$\tau = \upsilon$	Co-alias Equation (CE) 
have co-alias relations τ_1—υ_1, ⋯, τ_n—υ_n and $\theta\tau_1\cdots\tau_n$ is available	$\theta\upsilon_1\cdots\upsilon_n$	Congruence (Cng) 
$\theta\tau$ is available	$\exists x\, \theta x$	Existential Generalization (EG) 

Rule for lemmas	
prerequisite	*rule*
the goal is $\bot$	Lemma for *Reductio* (LFR) 

Additional rules (not guaranteed to be progressive)

Attachment rules		
what is required	*added resource*	*rule*
φ and ψ are both available	φ ∧ ψ	Adjunction (Adj) 
¬± φ or ¬± ψ is available	¬ (φ ∧ ψ)	Weakening (Wk) 
φ or ψ is available	φ ∨ ψ	
¬± φ or ψ is available	φ → ψ	

Rules for developing gaps		
logical form	*as resource*	*as goal*
universal ∀x θx	Universal Instantiation (UI) 	Universal Generalization (UG) 
existential ∃x θx	Proof by Choice (PCh) 	Non-constructive Proof (NcP) 

The parameter a used in UG and PCh should be new to the derivation; that is, it should appear only to the right of the scope line it labels

Rules for closing gaps (truth-functional logic)

when to close		*rule*
resources	*goal*	
φ	φ	*Quod Erat Demonstrandum* (QED) … φ [available] … | … | φ … → *n* QED … φ (*n*) … | … ● | φ …
φ and ¬ φ	⊥	Non-contradiction (Nc) … ¬ φ [available] … φ [available] … | … | ⊥ … → *n* Nc … ¬ φ (*n*) … φ (*n*) … | … ● | ⊥ …
any	⊤	*Ex Nihilo Verum* (ENV) … | … | ⊤ … → *n* ENV … | … ● | ⊤ …
⊥	*any*	*Ex Falso Quodlibet* (EFQ) … ⊥ … | … | φ … → *n* EFQ … ⊥ (*n*) … | … ● | φ …

Rules for closing gaps (equations)

In addition to the following rules for closing gaps, if the conditions for applying any rule are met except for differences between co-aliases, then the rule can be applied and is notated by adding “=” to its label; QED= and Nc= below are examples of this in the case of rules for closing gaps.

when to close			*rule*
co-aliases	*resources*	*goal*	
τ—υ	*any*	τ = υ	Equated Co-aliases (EC) … [τ and υ are co-aliases] … | … | τ = υ … → *n* EC … [τ and υ are co-aliases] … | … ● | τ = υ …
τ—υ	¬ τ = υ	⊥	Distinguished Co-aliases (DC) … [τ and υ are co-aliases] … ¬ τ = υ … | … | ⊥ … → *n* DC … [τ and υ are co-aliases] … ¬ τ = υ (*n*) … | … ● | ⊥ …
τ_1—υ_1, ⋯, τ_n—υ_n	$P\tau_1\cdots\tau_n$	$P\upsilon_1\cdots\upsilon_n$	QED given equations (QED=) … [have co-alias relations: τ_1—υ_1, ⋯, τ_n—υ_n] … $P\tau_1\cdots\tau_n$ … | … | $P\upsilon_1\cdots\upsilon_n$ … → *n* QED= … [have co-alias relations: τ_1—υ_1, ⋯, τ_n—υ_n] … $P\tau_1\cdots\tau_n$ (*n*) … | … ● | $P\upsilon_1\cdots\upsilon_n$ …
τ_1—υ_1, ⋯, τ_n—υ_n	$P\tau_1\cdots\tau_n$ ¬ $P\upsilon_1\cdots\upsilon_n$	⊥	Non-contradiction given equations (Nc=) … [have co-alias relations: τ_1—υ_1, ⋯, τ_n—υ_n] … $P\tau_1\cdots\tau_n$ … ¬ $P\upsilon_1\cdots\upsilon_n$ … | … | ⊥ … → *n* Nc= … [have co-alias relations: τ_1—υ_1, ⋯, τ_n—υ_n] … $P\tau_1\cdots\tau_n$ (*n*) … ¬ $P\upsilon_1\cdots\upsilon_n$ (*n*) … | … ● | ⊥ …