Overview

Basic system

Exploitation and planning rules		
sentence	*as a resource*	*as a goal*
atomic sentence	*none*	IP
negation $\neg\varphi$	CR (if φ not atomic & goal is ⊥)	RAA
conjunction $\varphi \wedge \psi$	Ext	Cnj
disjunction $\varphi \vee \psi$	PC	PE
conditional $\varphi \rightarrow \psi$	RC (if goal is ⊥)	CP
universal $\forall x\,\theta x$	UI	UG
existential $\exists x\,\theta x$	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

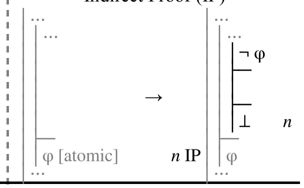
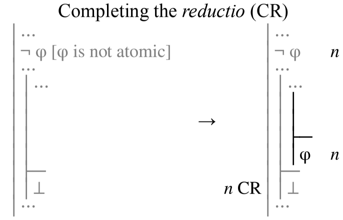
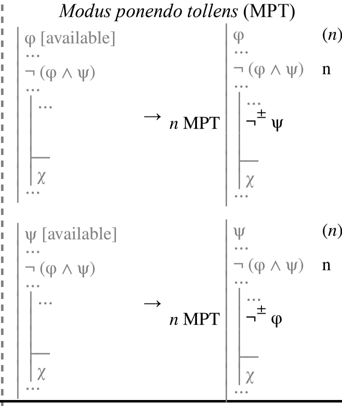
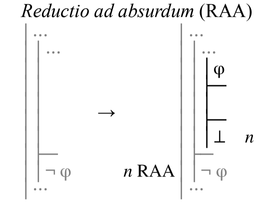
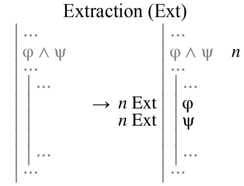
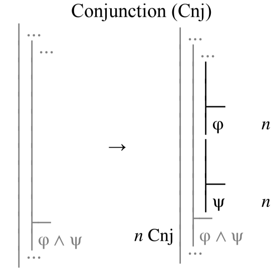
Attachment rules	
resource to be added	*rule*
φ ∧ ψ	Adj
¬ (φ ∧ ψ)	Wk
φ ∨ ψ	Wk
φ → ψ	Wk
τ = υ	CE
$\theta\upsilon_1\ldots\upsilon_n$	Cng
∃x θx	EG

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

Derivation rules

Basic system

Rules for developing gaps

logical form	*as resource*	*as goal*
atomic sentence	*no rule*	Indirect Proof (IP) 
negation ¬ φ	Completing the *reductio* (CR)  *Modus ponendo tollens* (MPT) 	*Reductio ad absurdum* (RAA) 
conjunction φ ∧ ψ	Extraction (Ext) 	Conjunction (Cnj) 

Rules for developing gaps		
logical form	*as resource*	*as goal*
disjunction $\varphi \vee \psi$	Proof by Cases (PC) … $\varphi \vee \psi$ … … χ → *n* PC … $\varphi \vee \psi$ *n* … … φ χ *n* ψ χ *n* χ *Modus Tollendo Ponens* (MTP) $\neg^{\pm}\varphi$ [available] … $\varphi \vee \psi$ … … χ → *n* MTP $\neg^{\pm}\varphi$ (*n*) … $\varphi \vee \psi$ n … … ψ χ $\neg^{\pm}\psi$ [available] … $\varphi \vee \psi$ … … χ → *n* MTP $\neg^{\pm}\psi$ (*n*) … $\varphi \vee \psi$ n … … φ χ	Proof of Exhaustion (PE) … … $\varphi \vee \psi$ → *n* PE … … $\neg^{\pm}\varphi$ ψ *n* $\varphi \vee \psi$ *OR* … … $\varphi \vee \psi$ → *n* PE … … $\neg^{\pm}\psi$ φ *n* $\varphi \vee \psi$
conditional $\varphi \to \psi$	Rejecting a Conditional (RC) … $\varphi \to \psi$ … … $\bot$ → *n* RC … $\varphi \to \psi$ *n* … … φ *n* ψ $\bot$ *n* $\bot$ *Modus Ponendo Ponens* (MPP) φ [available] … $\varphi \to \psi$ … … χ → *n* MPP φ (*n*) … $\varphi \to \psi$ *n* … … ψ χ *Modus Tollendo Tollens* (MTT) $\neg^{\pm}\psi$ [available] … $\varphi \to \psi$ … … χ → *n* MTT $\neg^{\pm}\psi$ (*n*) … $\varphi \to \psi$ *n* … … $\neg^{\pm}\varphi$ χ	Conditional Proof (CP) … … $\varphi \to \psi$ → *n* CP … … φ ψ *n* $\varphi \to \psi$

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

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

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 [available] … ψ [available] … | … | χ … → n Adj … φ (n) … ψ (n) … | … | φ ∧ ψ X | χ …
¬$^{\pm}$ φ or ¬$^{\pm}$ ψ is available	¬ (φ ∧ ψ)	Weakening (Wk) … ¬$^{\pm}$ φ [available] … | … | χ … → n Wk … ¬$^{\pm}$ φ (*n*) … | … | ¬ (φ ∧ ψ) X | χ … … ¬$^{\pm}$ ψ [available] … | … | χ … → n Wk … ¬$^{\pm}$ ψ (*n*) … | … | ¬ (φ ∧ ψ) X | χ …
φ or ψ is available	φ ∨ ψ	… φ [available] … | … | χ … → n Wk … φ (*n*) … | … | φ ∨ ψ X | χ … … ψ [available] … | … | χ … → n Wk … ψ (*n*) … | … | φ ∨ ψ X | χ …
¬$^{\pm}$ φ or ψ is available	φ → ψ	¬$^{\pm}$ φ [available] … | … | χ … → n Wk ¬$^{\pm}$ φ (*n*) … | … | φ → ψ X | χ … ψ [available] … | … | χ … → n Wk ψ (*n*) … | … | φ → ψ X | χ …

Rules for developing gaps		
logical form	*as resource*	*as goal*
universal ∀x θx	Universal Instantiation (UI) … ∀x θx … | … | φ … → n UI … ∀x θx τ:*n* … | … | θτ | φ …	Universal Generalization (UG) … | … | ∀x θx … → … | … | ⓐ | θa *n* | ∀x θx n UG …
existential ∃x θx	Proof by Choice (PCh) … ∃x θx … | … | φ … → … ∃x θx *n* … | … | ⓐ θa | φ *n* | φ n PCh …	Non-constructive Proof (NcP) … | … | ∃x θx … → … | … | ∀x ¬$^{\pm}$ θx | ⊥ *n* | ∃x θx n 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: resources	*goal*	*rule*
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: co-aliases	*resources*	*goal*	*rule*
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*) … │ … ● ⊢ ⊥ …