Logic

III. Classical semantics and meta-theoretic reasoning

柯向上

中央研究院資訊科學研究所

https://josh-hs-ko.github.io

Purpose of deduction systems (such as NJ and NK)

Constructing derivations in a deduction system is like playing a game of symbols, with the rules being strictly followed. But is there any meaning in playing the game?

Yes! We informally introduced the intuitionistic meaning of propositions/formulas and explained how each inference rule in NJ is valid in terms of this meaning. Thus every (correct) derivation gives a valid entailment.

We can make the connection mathematically precise, starting from defining a *semantics* for propositional logic, i.e., translating propositional formulas to (more familiar) mathematical entities.

Preliminary: structured proofs

Leslie Lamport proposes an informal yet principled way of writing proofs, *inspired by natural deduction*.

- Analyse a proof goal into assumptions and a conclusion (simplifying away connectives and quantifiers).
- Give the proof directly if it is simple (or a sketch otherwise).
- If a proof is more complex, separate the proof into intermediate steps, with the last one being 'QED', which stands for the conclusion that we set out to establish.
- Organise intermediate steps as nested, numbered lists, explicitly showing the tree structure of the proof, and making it easy to refer to previous steps.

Leslie Lamport [2012]. How to write a 21st century proof. *Journal of Fixed Point Theory and Applications*, 11:43–63. https://doi.org/10.1007/s11784-012-0071-6

A sample structured proof: even or odd

Theorem. Every natural number is either even or odd.

$\boxed{\text{ASSUME}}$ $x : \mathbb{N}$

$\boxed{\text{GOAL}}$ there exists $y : \mathbb{N}$ such that $x = y + y$ or $x = \mathit{suc}\,(y + y)$

$\boxed{\text{PROOF}}$ Induction on x, switching parity as x increments.

0 there exists $y : \mathbb{N}$ such that $0 = y + y$ or $0 = \mathit{suc}\,(y + y)$

1 $\boxed{\text{ASSUME}}$ $x' : \mathbb{N},\;\; y' : \mathbb{N},\;\; x' = y' + y'$ or $x' = \mathit{suc}\,(y' + y')$

$\boxed{\text{GOAL}}$ there exists $y : \mathbb{N}$ such that $\mathit{suc}\,x' = y + y$ or $\mathit{suc}\,x' = \mathit{suc}\,(y + y)$

2 QED

$\boxed{\text{PROOF}}$ Induction on x, with the goal as the motive: the base case is $\boxed{0}$, and the inductive case is $\boxed{1}$.

A sample structured proof: even or odd (continued)

0 there exists $y : \mathbb{N}$ such that $0 = y + y$ or $0 = suc\ (y + y)$

PROOF Choose $y := 0$, and then the left disjunct is true: $0 = 0 + 0 = y + y$.

1 ASSUME $x' : \mathbb{N}, \ \ y' : \mathbb{N}, \ \ x' = y' + y'$ or $x' = suc\ (y' + y')$

GOAL there exists $y : \mathbb{N}$ such that $suc\ x' = y + y$ or $suc\ x' = suc\ (y + y)$

PROOF Case analysis on whether x' is even or odd.

1.0 ASSUME $x' = y' + y'$

GOAL QED

1.1 ASSUME $x' = suc\ (y' + y')$

GOAL QED

1.2 QED

PROOF $\boxed{1.0}$ and $\boxed{1.1}$ exhaust the two cases in the disjunctive assumption.

A sample structured proof: even or odd (continued)

1.0 ASSUME $x' = y' + y'$

GOAL there exists $y : \mathbb{N}$ such that $suc\ x' = y + y$ or $suc\ x' = suc\ (y + y)$

PROOF (Switch the parity from even to odd.) Choose $y := y'$, and then the right disjunct is true: $suc\ x' = suc\ (y' + y') = suc\ (y + y)$.

1.1 ASSUME $x' = suc\ (y' + y')$

GOAL there exists $y : \mathbb{N}$ such that $suc\ x' = y + y$ or $suc\ x' = suc\ (y + y)$

PROOF Switch the parity from odd to even.

1.1.0 $suc\ x' = suc\ y' + suc\ y'$

1.1.1 QED

PROOF Choose $y := suc\ y'$, and then the left disjunct is proved by $\boxed{1.1.0}$.

A sample structured proof: even or odd (continued)

$\boxed{1.1.0}$ $suc\ x' = suc\ y' + suc\ y'$

$\boxed{\text{PROOF}}$

$$
\begin{aligned}
& suc\ x' \\
= \quad & \{ \text{assumption} \} \\
& suc\ (suc\ (y' + y')) \\
= \quad & \{ \text{property of addition} \} \\
& suc\ (y' + suc\ y') \\
= \quad & \{ \text{definition of addition} \} \\
& suc\ y' + suc\ y'
\end{aligned}
$$

Exercise. Write a structured proof of the property:
$suc\ (x + y) = x + suc\ y$ for all natural numbers x and y.

Questions. How do structured proofs correspond to derivations in natural deduction? What do you think about the design of structured proofs, especially compared with unstructured ones?

Induction principle on lists

The induction principle on lists may be semi-formally written as:

For any $A : \textsc{Type}$ and predicate $P\ xs$ where $xs : \textsc{List}\ A$,
if

- $P\ []$ and
- for all $x : A$ and $xs : \textsc{List}\ A$, $P\ xs$ implies $P\ (x :: xs)$,

then for all $xs : \textsc{List}\ A$, $P\ xs$.

(Note that this sentence involves multi-sorted and higher-order quantification.)

Structured proofs and induction on lists

The induction principle on lists can be used in structured proofs such as

ASSUME $A : \mathrm{Type},\ \ xs : \mathrm{List}\ A$

GOAL $xs +\!\!+ [\,] = xs$

PROOF Induction on xs.

0 $[\,] +\!\!+ [\,] = [\,]$

1 ASSUME $x : A,\ \ xs' : \mathrm{List}\ A,\ \ xs' +\!\!+ [\,] = xs'$

GOAL $(x :: xs') +\!\!+ [\,] = x :: xs'$

2 QED

PROOF Induction on xs, with the goal as the motive: the two cases are proved by $\boxed{0}$ and $\boxed{1}$.

Exercise. Rewrite some of your proofs that use induction on lists into structured proofs.

Classical semantics of propositional logic

Classical semantics adopts the *principle of bivalence*: every proposition denotes one of the two truth-values in the set $\mathbf{2} := \{0, 1\}$ (representing 'false' and 'true').

A function that maps variables to semantic values, such as one from $\mathcal{PV}$ to $\mathbf{2}$, is called an *assignment*.

Definition. The *truth-value interpretation* of propositional formulas is defined by

$$
\begin{array}{lcll}
\llbracket \_ \rrbracket : \textsc{Prop} \to (\mathcal{PV} \to \mathbf{2}) \to \mathbf{2} \\
\llbracket \bot \rrbracket \ \sigma & = & 0 \\
\llbracket V \rrbracket \ \sigma & = & \sigma \, V & \text{for } V : \mathcal{PV} \\
\llbracket \varphi \wedge \psi \rrbracket \ \sigma & = & \mathit{min} \ \{\llbracket \varphi \rrbracket \ \sigma, \llbracket \psi \rrbracket \ \sigma\} \\
\llbracket \varphi \vee \psi \rrbracket \ \sigma & = & \mathit{max} \ \{\llbracket \varphi \rrbracket \ \sigma, \llbracket \psi \rrbracket \ \sigma\} \\
\llbracket \varphi \to \psi \rrbracket \ \sigma & = & \mathit{max} \ \{1 - \llbracket \varphi \rrbracket \ \sigma, \llbracket \psi \rrbracket \ \sigma\}
\end{array}
$$

Exercise. Choose an arbitrary σ and compute $\llbracket (\mathtt{A} \wedge \mathtt{B}) \vee \neg(\mathtt{A} \wedge \mathtt{B}) \rrbracket \ \sigma$.

Semantic consequence

Definition. $\sigma : \mathcal{PV} \to \mathbf{2}$ *satisfies* a propositional formula φ exactly when $[\![\varphi]\!]\,\sigma = 1$; σ satisfies a list Γ of propositional formulas exactly when it satisfies every φ in Γ.

Definition. φ is a *semantic consequence* of Γ exactly when, for any $\sigma : \mathcal{PV} \to \mathbf{2}$, φ is satisfied by σ whenever Γ is satisfied by σ. In this case we write $\Gamma \models \varphi$.

Definition. φ is *valid* exactly when $\models \varphi$. In this case φ is called a *tautology*.

Example: $\models \varphi \vee \neg\varphi$ for any φ

ASSUME $\varphi : \textsc{Prop}, \quad \sigma : \mathcal{PV} \to \mathbf{2}$

GOAL $[\![\varphi \vee \neg\varphi]\!]\ \sigma = 1$

PROOF Case analysis on $[\![\varphi]\!]\ \sigma$.

0 ASSUME $[\![\varphi]\!]\ \sigma = 0$

GOAL QED

PROOF $[\![\varphi \vee \neg\varphi]\!]\ \sigma = \mathit{max}\ \{[\![\varphi]\!]\ \sigma, 1 - [\![\varphi]\!]\ \sigma\} = \mathit{max}\ \{0, 1\} = 1$

1 ASSUME $[\![\varphi]\!]\ \sigma = 1$

GOAL QED

PROOF $[\![\varphi \vee \neg\varphi]\!]\ \sigma = \mathit{max}\ \{[\![\varphi]\!]\ \sigma, 1 - [\![\varphi]\!]\ \sigma\} = \mathit{max}\ \{1, 0\} = 1$

2 QED

PROOF Either $[\![\varphi]\!]\ \sigma = 0$ or $[\![\varphi]\!]\ \sigma = 1$; $\boxed{0}$ and $\boxed{1}$.

Exercise. Prove $\varphi \vee \psi, \neg\psi \models \varphi$ for any φ and ψ.

$\models \varphi \vee \neg\varphi$ — the truth table method

We may just summarise the case analysis on $[\![\varphi]\!]\,\sigma$ and evaluation of the value of the entire propositional formula in a *truth table*.

φ	φ	$\vee$	$\neg$	φ
0	0	1	1	0
1	1	1	0	1

Theorem. Validity in classical propositional logic is *decidable*, i.e., there is a mechanical procedure that, given a propositional formula, decides whether it is valid or not in a finite amount of time.

Exercise. How do you use a truth table to show $\varphi \vee \psi, \neg\psi \models \varphi$?

Classical semantics of first-order logic

Definition. Given a signature $\mathcal{S} = (\mathcal{P}, \mathcal{F})$, an *$\mathcal{S}$-model* is a nonempty set $\mathcal{M}$ with two interpretation functions:

- $\llbracket p \rrbracket : (\mathcal{M} \to)^n\ \mathbf{2}$ for each predicate symbol $p/n : \mathcal{P}$, and
- $\llbracket f \rrbracket : (\mathcal{M} \to)^n\ \mathcal{M}$ for each function symbol $f/n : \mathcal{F}$.

Example. The standard model of the Peano arithmetic signature is $\mathbb{N}$ with $\llbracket \mathtt{Eq} \rrbracket\ x\ y := \begin{cases} 1 & \text{if } x = y \\ 0 & \text{otherwise} \end{cases}$, $\llbracket \mathtt{zero} \rrbracket := 0$, $\llbracket \mathtt{suc} \rrbracket\ x := \mathit{suc}\ x$, $\llbracket \mathtt{add} \rrbracket\ x\ y := x + y$, and $\llbracket \mathtt{mul} \rrbracket\ x\ y := x \times y$.

Question. Why does $\mathcal{M}$ need to be nonempty?

Definition. Let $\mathcal{S} = (\mathcal{P}, \mathcal{F})$ be a signature and $\mathcal{M}$ an $\mathcal{S}$-model. The interpretation of terms is defined by

$$
\begin{array}{lcll}
\multicolumn{4}{l}{\llbracket \_ \rrbracket : \textsc{Term}_{\mathcal{F}} \to (\mathcal{IV} \to \mathcal{M}) \to \mathcal{M}} \\
\llbracket v \rrbracket\ \sigma & = & \sigma\ v & \text{for } v : \mathcal{IV} \\
\llbracket f\ t_1 \ldots t_n \rrbracket\ \sigma & = & \llbracket f \rrbracket\ (\llbracket t_1 \rrbracket\ \sigma)\ \cdots\ (\llbracket t_n \rrbracket\ \sigma) & \text{for } f/n : \mathcal{F}
\end{array}
$$

Classical semantics of first-order logic (continued)

Definition. Let $\mathcal{S} = (\mathcal{P}, \mathcal{F})$ be a signature and $\mathcal{M}$ an $\mathcal{S}$-model. The *truth-value interpretation* of first-order formulas is defined by

$$
\begin{array}{lcll}
[\![\_]\!] : \textsc{Form}_{\mathcal{S}} \to (\mathcal{IV} \to \mathcal{M}) \to \mathbf{2} & & & \\
[\![\bot]\!]\ \sigma & = & 0 & \\
[\![p\ t_1 \ldots t_n]\!]\ \sigma & = & [\![p]\!]\ ([\![t_1]\!]\ \sigma)\ \cdots\ ([\![t_n]\!]\ \sigma) & \text{for } p/n : \mathcal{P} \\
[\![\varphi \wedge \psi]\!]\ \sigma & = & \mathit{min}\ \{[\![\varphi]\!]\ \sigma, [\![\psi]\!]\ \sigma\} & \\
[\![\varphi \vee \psi]\!]\ \sigma & = & \mathit{max}\ \{[\![\varphi]\!]\ \sigma, [\![\psi]\!]\ \sigma\} & \\
[\![\varphi \to \psi]\!]\ \sigma & = & \mathit{max}\ \{1 - [\![\varphi]\!]\ \sigma, [\![\psi]\!]\ \sigma\} & \\
[\![\forall\, v.\ \varphi]\!]\ \sigma & = & \mathit{min}\ \{\, [\![\varphi]\!]\ (\sigma\,[v \mapsto m]) \mid m : \mathcal{M}\,\} & \\
[\![\exists\, v.\ \varphi]\!]\ \sigma & = & \mathit{max}\ \{\, [\![\varphi]\!]\ (\sigma\,[v \mapsto m]) \mid m : \mathcal{M}\,\} &
\end{array}
$$

where $(\sigma\,[v \mapsto m])\ u \ := \begin{cases} m & \text{if } u = v \\ \sigma\ u & \text{otherwise.} \end{cases}$

Exercise. In the standard model of Peano arithmetic, choose an arbitrary σ and compute $[\![\forall\, \mathtt{x}.\ (\mathtt{Eq}\ \mathtt{x}\ \mathtt{zero} \vee \exists\, \mathtt{y}.\ \mathtt{Eq}\ \mathtt{x}\ (\mathtt{suc}\ \mathtt{y}))]\!]\ \sigma$.

Semantic consequence

Let $\mathcal{S}$ be a signature, φ, $\psi : \textsc{Form}_{\mathcal{S}}$, and $\Gamma : \textsc{List Form}_{\mathcal{S}}$.

Definition. An $\mathcal{S}$-model $\mathcal{M}$ and $\sigma : \mathcal{IV} \to \mathcal{M}$ *satisfy* φ exactly when $[\![\varphi]\!]\ \sigma = 1$; they satisfy Γ exactly when they satisfy every formula in Γ.

Definition. φ is a *semantic consequence* of Γ exactly when, for any $\mathcal{S}$-model $\mathcal{M}$ and $\sigma : \mathcal{IV} \to \mathcal{M}$, φ is satisfied by $\mathcal{M}$ and σ whenever Γ is satisfied by $\mathcal{M}$ and σ. In this case we write $\Gamma \models \varphi$.

Definition. φ is *valid* exactly when $\emptyset \models \varphi$. In this case we also call φ a *tautology* and simply write $\models \varphi$.

Exercise. With the Peano arithmetic signature, does $\models \forall\, \mathtt{x}.\ (\mathtt{Eq\ x\ zero} \lor \exists\, \mathtt{y}.\ \mathtt{Eq\ x\ (suc\ y)})$ hold?

Relationship between deduction system and semantics

Theorem. NK is *sound* with respect to the classical semantics: $\Gamma \vdash_{\mathrm{NK}} \varphi$ implies $\Gamma \models \varphi$ for all Γ and φ.

PROOF *Induction on the derivation* of $\Gamma \vdash_{\mathrm{NK}} \varphi$.

Corollary. NJ is sound with respect to the classical semantics.

PROOF Every NJ derivation is an NK derivation.

Theorem. NK is *complete* with respect to the classical semantics: $\Gamma \models \varphi$ implies $\Gamma \vdash_{\mathrm{NK}} \varphi$ for all Γ and φ.

NJ is, however, not complete with respect to the classical semantics, since, for instance, $\mathtt{A} \vee \neg\mathtt{A}$ is classically valid but not derivable in NJ.

Underivability

Theorem (consistency)**.** There is no NJ/NK derivation of $\vdash \bot$.

$\boxed{\text{ASSUME}}$ $\vdash \bot$

$\boxed{\text{GOAL}}$ contradiction

$\boxed{\text{PROOF}}$ By soundness we get $\models \bot$, which is false however.

Question. Why is consistency important?

It is possible to prove this theorem purely syntactically, but it takes more than a straightforward induction.

Inductively defined sets and induction principles

Every inductively defined set —such as $\mathbb{N}$, $\textsc{List}\ A$, various kinds of $\textsc{Tree}\ A$, and $\textsc{Prop}$— is equipped with an *induction principle*.

Informally but generically: Let P be a predicate on an inductively defined set S. If we can show that P is 'propagated' by every construction rule of S, then for any $x : S$, a proof of $P\ x$ can be constructed in the same way as how x is constructed.

Exercise. State and justify the induction principle on $\textsc{Prop}$.

Inductive families

Derivations are also inductively defined, but not as a single set but as an *indexed family of sets*.

Take multi-step β-reduction in λ-calculus as an example:

$$\frac{}{u \twoheadrightarrow_\beta u}\ (0\text{-step}) \qquad \frac{t \to_\beta t' \qquad t' \twoheadrightarrow_\beta u}{t \twoheadrightarrow_\beta u}\ (n{+}1\text{-step})$$

Definition. The family of sets $\mathrm{MBR}\ t\ u$ indexed by $t,\ u : \Lambda$ are inductively defined by the following rules:

- $\dfrac{}{u \twoheadrightarrow_\beta u}\ (0\text{-step}) : \mathrm{MBR}\ u\ u$ for all $u : \Lambda$;
- $\dfrac{d \qquad e}{t \twoheadrightarrow_\beta u}\ (n{+}1\text{-step}) : \mathrm{MBR}\ t\ u$

  for all $t,\ t',\ u : \Lambda,\ d : \mathrm{OBR}\ t\ t'$, and $e : \mathrm{MBR}\ t'\ u$.

We also use $t \twoheadrightarrow_\beta u$ as an alternative name for $\mathrm{MBR}\ t\ u$.

Exercise. Define derivations of one-step β-reduction as a family of sets $\mathrm{OBR}\ t\ u$ indexed by $t,\ u : \Lambda$.

Induction principle on MBR

For any predicate $P\ t\ u\ d$ where t, $u : \Lambda$ and $d : t \twoheadrightarrow_\beta u$,
if

- for all $u : \Lambda$, $P\ u\ u\ \left(\dfrac{}{u \twoheadrightarrow_\beta u}\ (0\text{-step})\right)$ and
- for all t, t', $u : \Lambda$, $d : t \rightarrow_\beta t'$, and $e : t' \twoheadrightarrow_\beta u$,
  $P\ t'\ u\ e$ implies $P\ t\ u\ \left(\dfrac{d \qquad e}{t \twoheadrightarrow_\beta u}\ (n{+}1\text{-step})\right)$,

then for all t, $u : \Lambda$, and $d : t \twoheadrightarrow_\beta u$, $P\ t\ u\ d$.

Exercise. What is the induction principle on OBR?

One-index induction principle on MBR

For any $u : \Lambda$ and predicate $P\ t\ d$ where $t : \Lambda$ and $d : t \twoheadrightarrow_\beta u$, if

- $P\ u\ \left(\dfrac{}{u \twoheadrightarrow_\beta u}\ (0\text{-step})\right)$ and
- for all t, $t' : \Lambda$, $d : t \rightarrow_\beta t'$, and $e : t' \twoheadrightarrow_\beta u$,
  $P\ t'\ e$ implies $P\ t\ \left(\dfrac{d \qquad e}{t \twoheadrightarrow_\beta u}\ (n{+}1\text{-step})\right)$,

then for all $t : \Lambda$ and $d : t \twoheadrightarrow_\beta u$, $P\ t\ d$.

Exercise. Prove this one-index induction principle from the main one on MBR.

Exercise. Give an alternative definition of MBR as a one-index family of sets whose main induction principle is the one above.

Index-only one-index induction principle on MBR

For any $u : \Lambda$ and predicate $P\ t$ where $t : \Lambda$,
if

- $P\ u$ and
- for all t, $t' : \Lambda$, if $t \rightarrow_\beta t'$ and $P\ t'$, then $P\ t$,

then for all $t : \Lambda$, if $t \twoheadrightarrow_\beta u$, then $P\ t$.

Exercise. Prove this index-only version from the full version.

Exercise. Use the above index-only induction principle to prove transitivity of '$\twoheadrightarrow_\beta$': if $t \twoheadrightarrow_\beta u$ and $u \twoheadrightarrow_\beta v$, then $t \twoheadrightarrow_\beta v$.

Exercise. How about using the index-only version of the main induction principle to prove transitivity of '$\twoheadrightarrow_\beta$'?

NJ/NK as an inductive family

Definition. The family of sets $\mathrm{NK}\ \Gamma\ \varphi$ indexed by $\Gamma : \textsc{List Prop}$ and $\varphi : \textsc{Prop}$ are inductively defined by the following rules:

- $\dfrac{}{\Gamma \vdash \varphi}\,(\mathsf{A}) : \mathrm{NK}\ \Gamma\ \varphi$

  for all $\Gamma : \textsc{List Prop}$ and $\varphi : \textsc{Prop}$ such that φ appears in Γ;
- $\dfrac{d}{\Gamma \vdash \varphi \to \psi}\,(\to\mathsf{I}) : \mathrm{NK}\ \Gamma\ (\varphi \to \psi)$

  for all $\Gamma : \textsc{List Prop}$, φ, $\psi : \textsc{Prop}$, and $d : \mathrm{NK}\ (\Gamma, \varphi)\ \psi$;
- and so on.

We also use $\Gamma \vdash_{\mathrm{NK}} \varphi$ or simply $\Gamma \vdash \varphi$ as an alternative name for $\mathrm{NK}\ \Gamma\ \varphi$.

Exercise. Give the rest of the rules.

Induction principle on NJ/NK

For any predicate $P\ \Gamma\ \varphi\ d$
where $\Gamma : \textsc{List Prop}$, $\varphi : \textsc{Prop}$, and $d : \Gamma \vdash \varphi$,
if

- for all $\Gamma : \textsc{List Prop}$ and $\varphi : \textsc{Prop}$,
  if φ appears in Γ, then $P\ \Gamma\ \varphi\ \left(\dfrac{}{\Gamma \vdash \varphi}\ (\mathsf{A})\right)$,
- for all $\Gamma : \textsc{List Prop}$, φ, $\psi : \textsc{Prop}$, and $d : \Gamma, \varphi \vdash \psi$,
  $P\ (\Gamma, \varphi)\ \psi\ d$ implies $P\ \Gamma\ (\varphi \to \psi)\ \left(\dfrac{d}{\Gamma \vdash \varphi \to \psi}\ ({\to}\mathsf{I})\right)$,
- and so on,

then for all $\Gamma : \textsc{List Prop}$, $\varphi : \textsc{Prop}$, and $d : \Gamma \vdash \varphi$, $P\ \Gamma\ \varphi\ d$.

Exercise. What are the rest of the premises?

Index-only induction principle on NJ/NK

For any predicate $P\,\Gamma\,\varphi$ where $\Gamma : \textsc{List Prop}$ and $\varphi : \textsc{Prop}$,
if

- for all $\Gamma : \textsc{List Prop}$ and $\varphi : \textsc{Prop}$,
  if φ appears in Γ, then $P\,\Gamma\,\varphi$,
- for all $\Gamma : \textsc{List Prop}$ and φ, $\psi : \textsc{Prop}$,
  $P\,(\Gamma, \varphi)\,\psi$ implies $P\,\Gamma\,(\varphi \to \psi)$,
- and so on,

then for all $\Gamma : \textsc{List Prop}$, $\varphi : \textsc{Prop}$, if $\Gamma \vdash \varphi$, then $P\,\Gamma\,\varphi$.

Exercise. What are the rest of the premises?

Exercise. Use the induction principle to prove Glivenko's theorem.

Soundness

Theorem. $\Gamma \vdash_{\mathrm{NK}} \varphi$ implies $\Gamma \models \varphi$ for all Γ and φ.

ASSUME $\Gamma : \textsc{List Prop}, \;\; \varphi : \textsc{Prop}, \;\; \Gamma \vdash_{\mathrm{NK}} \varphi$

GOAL $\Gamma \models \varphi$

PROOF Induction on the NK derivation.

0 Soundness of the (A) rule

ASSUME $\Gamma : \textsc{List Prop}, \;\; \varphi : \textsc{Prop}, \;\; \varphi$ appears in Γ,
$\sigma : \mathcal{PV} \to \mathbf{2}, \;\; \sigma$ satisfies Γ

GOAL $[\![\varphi]\!]\ \sigma = 1$

PROOF By the assumptions, σ satisfies Γ, and φ appears in Γ, so σ satisfies φ in particular.

$\vdots$

11 Index-only induction on the assumed derivation of $\Gamma \vdash_{\mathrm{NK}} \varphi$ with the goal as the motive: the cases are proved by $\boxed{0}$–$\boxed{10}$.

Soundness: implication introduction

1 Soundness of the ($\to$I) rule

ASSUME $\Gamma : \textsc{List Prop}$, $\varphi, \psi : \textsc{Prop}$, $\Gamma, \varphi \models \psi$, $\sigma : \mathcal{PV} \to \mathbf{2}$, σ satisfies Γ

GOAL $[\![\varphi \to \psi]\!]\ \sigma = 1$

PROOF Case analysis on the truth value of φ.

1.0 ASSUME $[\![\varphi]\!]\ \sigma = 0$

GOAL QED

1.1 ASSUME $[\![\varphi]\!]\ \sigma = 1$

GOAL QED

1.2 QED

PROOF 1.0 and 1.1 cover all possible values of $[\![\varphi]\!]\ \sigma$.

Soundness: implication introduction (continued)

1.0 ASSUME $[\![\varphi]\!]\ \sigma = 0$

GOAL $[\![\varphi \to \psi]\!]\ \sigma = 1$

PROOF

$$
\begin{array}{ll}
 & [\![\varphi \to \psi]\!]\ \sigma \\
= & \quad \{\, \text{definition of } [\![\_]\!] \,\} \\
 & \mathit{max}\,\{\, 1 - [\![\varphi]\!]\ \sigma,\ [\![\psi]\!]\ \sigma \,\} \\
= & \quad \{\, \text{assumption } [\![\varphi]\!]\ \sigma = 0\text{; arithmetic} \,\} \\
 & \mathit{max}\,\{\, 1,\ [\![\psi]\!]\ \sigma \,\} \\
= & \quad \{\, 1 \geqslant [\![\psi]\!]\ \sigma \,\} \\
 & 1
\end{array}
$$

Soundness: implication introduction (continued)

1.1 ASSUME $[\![\varphi]\!]\ \sigma = 1$

GOAL $[\![\varphi \to \psi]\!]\ \sigma = 1$

PROOF ψ must be true, and therefore so must $\varphi \to \psi$.

1.1.0 σ satisfies Γ, φ. PROOF σ satisfies Γ and φ.

1.1.1 $[\![\psi]\!]\ \sigma = 1$. PROOF $\Gamma, \varphi \models \psi$ and 1.1.0.

1.1.2 QED

PROOF

$$
\begin{array}{cl}
 & [\![\varphi \to \psi]\!]\ \sigma \\
= & \quad \{\text{ definition of } [\![\_]\!] \ \} \\
 & \mathit{max}\ \{\, 1 - [\![\varphi]\!]\ \sigma,\ [\![\psi]\!]\ \sigma \,\} \\
= & \quad \{\text{ assumption and 1.1.1 }\} \\
 & \mathit{max}\ \{\, 0,\ 1 \,\} \\
= & \quad \{\text{ definition of } \mathit{max} \} \\
 & 1
\end{array}
$$

Exercise. What about soundness of other rules?