Introduction to Proof-theoretic Semantics

Lecture 2: Proof-theoretic Validity

Alexander V. Gheorghiu Tao Gu

University of Southampton & University College London
United Kingdom

Overview

1. Recap Lecture 1
2. Prawitz's Normalization Theory
3. Proof-theoretic Validity
4. Alternative Proof-theoretic Validity

Table of Contents

Theories of Meaning

The denotational idea of meaning is powerful — ‘Tammy is a vixen’ means ‘Tammy’ belongs to the class ‘Vixen’

Theories of Meaning

The denotational idea of meaning is powerful — 'Tammy is a vixen' means 'Tammy' belongs to the class 'Vixen'

But it has some problems in terms of intuitions about meaning.

Theories of Meaning

The denotational idea of meaning is powerful — 'Tammy is a vixen' means 'Tammy' belongs to the class 'Vixen'

But it has some problems in terms of intuitions about meaning.

In particular, having ostensibly been given the 'meaning' of a sentence (e.g., 'Tammy is a vixen') doesn't mean we actually understand it!

We are exploring an alternative called *inferentialism*.

Theories of Meaning

The denotational idea of meaning is powerful — 'Tammy is a vixen' means 'Tammy' belongs to the class 'Vixen'

But it has some problems in terms of intuitions about meaning.

In particular, having ostensibly been given the 'meaning' of a sentence (e.g., 'Tammy is a vixen') doesn't mean we actually understand it!

We are exploring an alternative called *inferentialism*.

Definition (Inferentialism)

Inferentialism is the view that the meaning of a linguistic expression is determined by its role in reasoning — specifically, by the inferences it licenses and is licensed by.

Atomic Systems

To deliver proof-theoretic semantics we will use *atomic systems*.

Atomic Systems

To deliver proof-theoretic semantics we will use *atomic systems*.

Intuitively, an atomic system represents an agent's beliefs about the world as a set of inferential commitments.

Atomic Systems

To deliver proof-theoretic semantics we will use *atomic systems*.

Intuitively, an atomic system represents an agent's beliefs about the world as a set of inferential commitments.

Definition (Atomic System)

An atomic system is a set of atomic rules. An atomic rule is a figure of the following form in which $p_1, ..., p_n, c$ are atomic formulae and $P_1, ..., P_n$ are sets thereof:

$$\frac{}{c} \qquad \frac{p_1 \quad \dots \quad p_n}{c} \qquad \frac{[P_1] \qquad\qquad [P_n]}{p_1 \quad \dots \quad p_n \qquad c}$$

Importantly, atomic systems are pre-logical in the sense that they do not contain any logical syntax.

Collection of Atomic Systems

As things develop later on, it will be useful to consider various notions of atomic system.

Collection of Atomic Systems

As things develop later on, it will be useful to consider various notions of atomic system.

Definition (Basis)

A basis $\mathfrak{B}$ is a set of atomic systems.

Collection of Atomic Systems

As things develop later on, it will be useful to consider various notions of atomic system.

Definition (Basis)

A basis $\mathfrak{B}$ is a set of atomic systems.

In particular, we will consider the following:

- $\mathfrak{B}_1$ — contains atomic systems with 'first-level' rules,

$$\frac{}{c} \qquad \frac{p_1 \quad \dots \quad p_n}{c}$$

Collection of Atomic Systems

As things develop later on, it will be useful to consider various notions of atomic system.

Definition (Basis)

A basis $\mathfrak{B}$ is a set of atomic systems.

In particular, we will consider the following:

- $\mathfrak{B}_1$ — contains atomic systems with 'first-level' rules,

$$\frac{}{c} \qquad \frac{p_1 \quad \dots \quad p_n}{c}$$

- $\mathfrak{B}_2$ — contains atomic systems with 'second-level' rules,

$$\frac{}{c} \qquad \frac{p_1 \quad \dots \quad p_n}{c} \qquad \frac{\begin{matrix}[P_1] & & [P_n] \\ p_1 & \dots & p_n\end{matrix}}{c}$$

Collection of Atomic Systems

As things develop later on, it will be useful to consider various notions of atomic system.

Definition (Basis)

A basis $\mathfrak{B}$ is a set of atomic systems.

In particular, we will consider the following:

- $\mathfrak{B}_1$ — contains atomic systems with 'first-level' rules,

$$\frac{}{c} \qquad \frac{p_1 \quad \dots \quad p_n}{c}$$

- $\mathfrak{B}_2$ — contains atomic systems with 'second-level' rules,

$$\frac{}{c} \qquad \frac{p_1 \quad \dots \quad p_n}{c} \qquad \frac{\begin{matrix}[P_1] & & [P_n] \\ p_1 & \dots & p_n\end{matrix}}{c}$$

Collection of Atomic Systems

As things develop later on, it will be useful to consider various notions of atomic system.

Definition (Basis)

A basis $\mathfrak{B}$ is a set of atomic systems.

In particular, we will consider the following:

- $\mathfrak{B}_1$ — contains atomic systems with 'first-level' rules,

$$\frac{}{C} \qquad \frac{p_1 \quad \dots \quad p_n}{C}$$

- $\mathfrak{B}_2$ — contains atomic systems with 'second-level' rules,

$$\frac{}{C} \qquad \frac{p_1 \quad \dots \quad p_n}{C} \qquad \frac{\begin{matrix}[P_1] & & [P_n] \\ p_1 & \dots & p_n\end{matrix}}{C}$$

This will have a profound effect on our notion of validity. *One that we don't quite understand yet.*

Gentzen's Quote

Remember that in *Investigations into Logical Deduction* (1934), Gentzen writes:

> *The introductions represent, as it were, the 'definitions' of the symbols concerned,*

Gentzen's Quote

Remember that in *Investigations into Logical Deduction* (1934), Gentzen writes:

> *The introductions represent, as it were, the 'definitions' of the symbols concerned,and the eliminations are no more, in the final analysis, than the consequences of these definitions. This fact may be expressed as follows: In eliminating a symbol, we may use the formula with whose terminal symbol we are dealing only 'in the sense afforded it by the introduction of that symbol'.*

Gentzen's Quote

Remember that in *Investigations into Logical Deduction* (1934), Gentzen writes:

> *The introductions represent, as it were, the 'definitions' of the symbols concerned,and the eliminations are no more, in the final analysis, than the consequences of these definitions. This fact may be expressed as follows: In eliminating a symbol, we may use the formula with whose terminal symbol we are dealing only 'in the sense afforded it by the introduction of that symbol'.*

This is an inferentialist position — meaning-as-use, where 'use' means 'inferential role'.

Gentzen's Quote

Remember that in *Investigations into Logical Deduction* (1934), Gentzen writes:

> *The introductions represent, as it were, the 'definitions' of the symbols concerned,and the eliminations are no more, in the final analysis, than the consequences of these definitions. This fact may be expressed as follows: In eliminating a symbol, we may use the formula with whose terminal symbol we are dealing only 'in the sense afforded it by the introduction of that symbol'.*

This is an inferentialist position — meaning-as-use, where 'use' means 'inferential role'.

Task!

Deliver a mathematical account of this theory of meaning for intuitionistic logic.

Table of Contents

Harmony

There have been a range of approaches to reconcile the introductions with the eliminations, both philosophical and mathematical.

Harmony

There have been a range of approaches to reconcile the introductions with the eliminations, both philosophical and mathematical.

A major motivation is to forbid connectives such as Prior's 'tonk' ($\mathtt{t}$):

$$\underbrace{\frac{A}{A\ \mathtt{t}\ B}\ \mathtt{t}_{\mathsf{I}1}\quad \frac{B}{A\ \mathtt{t}\ B}\ \mathtt{t}_{\mathsf{I}2}}_{\text{introductions}} \qquad \underbrace{\frac{A\ \mathtt{t}\ B}{A}\ \mathtt{t}_{\mathsf{E}1}\quad \frac{A\ \mathtt{t}\ B}{B}\ \mathtt{t}_{\mathsf{E}2}}_{\text{eliminations}}$$

Harmony

There have been a range of approaches to reconcile the introductions with the eliminations, both philosophical and mathematical.

A major motivation is to forbid connectives such as Prior's 'tonk' ($\mathtt{t}$):

$$\underbrace{\frac{A}{A\ \mathtt{t}\ B}\ \mathtt{t}_{\mathsf{I1}}\quad \frac{B}{A\ \mathtt{t}\ B}\ \mathtt{t}_{\mathsf{I2}}}_{\text{introductions}}\qquad \underbrace{\frac{A\ \mathtt{t}\ B}{A}\ \mathtt{t}_{\mathsf{E1}}\quad \frac{A\ \mathtt{t}\ B}{B}\ \mathtt{t}_{\mathsf{E2}}}_{\text{eliminations}}$$

A logic with this connective is inconsistent.

The introduction and elimination rules should be in "harmony"[1], or governed by a "principle of harmony"[2].

[1] Dummett, *Frege: Philosophy of Language* (1973)
[2] Tennant, *Natural Logic* (1978)

Harmony

There have been a range of approaches to reconcile the introductions with the eliminations, both philosophical and mathematical.

A major motivation is to forbid connectives such as Prior's 'tonk' ($\mathtt{t}$):

$$\underbrace{\frac{A}{A\ \mathtt{t}\ B}\ \mathtt{t}_{\mathsf{I1}}\quad \frac{B}{A\ \mathtt{t}\ B}\ \mathtt{t}_{\mathsf{I2}}}_{\text{introductions}} \qquad \underbrace{\frac{A\ \mathtt{t}\ B}{A}\ \mathtt{t}_{\mathsf{E1}}\quad \frac{A\ \mathtt{t}\ B}{B}\ \mathtt{t}_{\mathsf{E2}}}_{\text{eliminations}}$$

A logic with this connective is inconsistent.

The introduction and elimination rules should be in "harmony"[1], or governed by a "principle of harmony"[2].

Unfortunately, "this terminology is not uniform and sometimes not even fully clear."[3]

[1] Dummett, *Frege: Philosophy of Language* (1973)
[2] Tennant, *Natural Logic* (1978)
[3] Schroeder-Heister, SEP 'Proof-theoretic Semantics'

Prawitz's Thesis

In 1965, Dag Prawitz successfully defended his thesis *Natural Deduction. A Proof-Theoretic Study*.

Prawitz's Thesis

In 1965, Dag Prawitz successfully defended his thesis *Natural Deduction. A Proof-Theoretic Study*.

This is one of the major works in proof theory, especially natural deduction.

Prawitz's Thesis

In 1965, Dag Prawitz successfully defended his thesis *Natural Deduction. A Proof-Theoretic Study*.

This is one of the major works in proof theory, especially natural deduction.

One important contribution is an interpretation of 'inversion' in terms of natural deduction that gives an explanation of Gentzen's quote:

Prawitz's Thesis

In 1965, Dag Prawitz successfully defended his thesis *Natural Deduction. A Proof-Theoretic Study*.

This is one of the major works in proof theory, especially natural deduction.

One important contribution is an interpretation of 'inversion' in terms of natural deduction that gives an explanation of Gentzen's quote:

> *Let α be an application of an elimination rule that has B as consequence.*

Prawitz's Thesis

In 1965, Dag Prawitz successfully defended his thesis *Natural Deduction. A Proof-Theoretic Study*.

This is one of the major works in proof theory, especially natural deduction.

One important contribution is an interpretation of 'inversion' in terms of natural deduction that gives an explanation of Gentzen's quote:

> *Let α be an application of an elimination rule that has B as consequence. Then, deductions that satisfy the sufficient condition [...] for deriving the major premiss of α, when combined with deductions of the minor premisses of α (if any), already 'contain' a deduction of B;*

Prawitz's Thesis

In 1965, Dag Prawitz successfully defended his thesis *Natural Deduction. A Proof-Theoretic Study*.

This is one of the major works in proof theory, especially natural deduction.

One important contribution is an interpretation of 'inversion' in terms of natural deduction that gives an explanation of Gentzen's quote:

> *Let α be an application of an elimination rule that has B as consequence. Then, deductions that satisfy the sufficient condition [...] for deriving the major premiss of α, when combined with deductions of the minor premisses of α (if any), already 'contain' a deduction of B; the deduction of B is thus obtainable directly from the given deductions without the addition of α.*

Prawitz's Thesis

In 1965, Dag Prawitz successfully defended his thesis *Natural Deduction. A Proof-Theoretic Study*.

This is one of the major works in proof theory, especially natural deduction.

One important contribution is an interpretation of 'inversion' in terms of natural deduction that gives an explanation of Gentzen's quote:

> *Let α be an application of an elimination rule that has B as consequence. Then, deductions that satisfy the sufficient condition [...] for deriving the major premiss of α, when combined with deductions of the minor premisses of α (if any), already 'contain' a deduction of B; the deduction of B is thus obtainable directly from the given deductions without the addition of α.*

That's a mouthful! But it has a useful algebraic treatment.

Prawitz's Reductions

We will give an example.

Prawitz's Reductions

We will give an example. Consider the rules for implication:

$$\frac{\begin{array}{c}[\varphi]\\ \psi\end{array}}{\varphi \to \psi} \qquad \frac{\varphi \quad \varphi \to \psi}{\psi}$$

Prawitz's Reductions

We will give an example. Consider the rules for implication:

$$\dfrac{\begin{array}{c}[\varphi]\\ \psi\end{array}}{\varphi \to \psi} \qquad \dfrac{\varphi \quad \varphi \to \psi}{\psi}$$

Suppose we used them together,

$$\dfrac{\mathcal{D}_1 \quad \begin{array}{c}[A]\\ \mathcal{D}_2\\ B\end{array}}{\dfrac{A \quad A \to B}{B}}$$

Prawitz's Reductions

We will give an example. Consider the rules for implication:

$$\dfrac{\begin{array}{c}[\varphi]\\ \psi\end{array}}{\varphi \to \psi} \qquad \dfrac{\varphi \quad \varphi \to \psi}{\psi}$$

Suppose we used them together,

$$\dfrac{\mathcal{D}_1 \quad \begin{array}{c}[A]\\ \mathcal{D}_2\\ B\end{array}}{\dfrac{A \quad A \to B}{B}}$$

We have derived B using $\mathcal{D}_1$ and $\mathcal{D}_2$ using an elimination rule... but this derivation was 'already contained' in $\mathcal{D}_1$ and $\mathcal{D}_2$!

Normal Forms

We can ‘reduce’ the derivation to get rid of extraneous deductions:

$$\dfrac{\mathcal{D}_1 \quad \overset{[A]}{\overset{\mathcal{D}_2}{B}}}{\dfrac{A \quad A \to B}{B}} \quad \rightsquigarrow \quad \begin{array}{c} \mathcal{D}_1 \\ A \\ \mathcal{D}_2 \\ B \end{array}$$

Normal Forms

We can 'reduce' the derivation to get rid of extraneous deductions:

$$\dfrac{\mathcal{D}_1 \quad \dfrac{\begin{array}{c}[A]\\ \mathcal{D}_2\\ B\end{array}}{A \to B}}{B} \quad \rightsquigarrow \quad \begin{array}{c}\mathcal{D}_1\\ A\\ \mathcal{D}_2\\ B\end{array}$$

Prawitz defined a set of such reductions that define the reduction operator $\rightsquigarrow$.

Normal Forms

We can 'reduce' the derivation to get rid of extraneous deductions:

$$\dfrac{\mathcal{D}_1 \quad \dfrac{\begin{array}{c}[A] \\ \mathcal{D}_2 \\ B\end{array}}{A \to B}}{B} \quad \rightsquigarrow \quad \begin{array}{c}\mathcal{D}_1 \\ A \\ \mathcal{D}_2 \\ B\end{array}$$

Prawitz defined a set of such reductions that define the reduction operator $\rightsquigarrow$.

Definition (Normal Form)

A derivation $\mathcal{D}$ is in normal form if it is irreducible — that is, there is no $\mathcal{D}'$ such that $\mathcal{D} \rightsquigarrow \mathcal{D}'$.

Normalization Theorem

Why does this matter?

Normalization Theorem

Why does this matter?

Theorem (Prawitz 1965)

If $\mathcal{D}$ is an NJ*-derivation, then $\mathcal{D}$ reduces to a normal form $\mathcal{D}'$ with the same end formula and with no maximal formulae — that is, no formulae that are introduced by an introduction rule and then eliminated by an elimination rule.*

Normalization Theorem

Why does this matter?

Theorem (Prawitz 1965)

If $\mathcal{D}$ is an NJ*-derivation, then $\mathcal{D}$ reduces to a normal form $\mathcal{D}'$ with the same end formula and with no maximal formulae — that is, no formulae that are introduced by an introduction rule and then eliminated by an elimination rule.*

While this is interesting in its own right, what matters is a consequence of this result.

Normalization Theorem

Why does this matter?

Theorem (Prawitz 1965)

If $\mathcal{D}$ is an NJ*-derivation, then $\mathcal{D}$ reduces to a normal form $\mathcal{D}'$ with the same end formula and with no maximal formulae — that is, no formulae that are introduced by an introduction rule and then eliminated by an elimination rule.*

While this is interesting in its own right, what matters is a consequence of this result.

Corollary (Prawitz 1965)

If $\mathcal{D}$ is an NJ*-derivation, then $\mathcal{D}$ reduces to a normal form $\mathcal{D}'$ with the last rule used being an introduction.*

Normalization Theorem

Why does this matter?

Theorem (Prawitz 1965)

If $\mathcal{D}$ is an NJ*-derivation, then $\mathcal{D}$ reduces to a normal form $\mathcal{D}'$ with the same end formula and with no maximal formulae — that is, no formulae that are introduced by an introduction rule and then eliminated by an elimination rule.*

While this is interesting in its own right, what matters is a consequence of this result.

Corollary (Prawitz 1965)

If $\mathcal{D}$ is an NJ*-derivation, then $\mathcal{D}$ reduces to a normal form $\mathcal{D}'$ with the last rule used being an introduction.*

This is a crucial result. It gives a kind of priority to the introduction rules as expected given Gentzen's remark.

Aside: Type Theory

Normalization theory is actually a very important topic aside from the philosophical considerations we have used to introduce it.

Aside: Type Theory

Normalization theory is actually a very important topic aside from the philosophical considerations we have used to introduce it.

Type theory underpins much of automated reasoning, programming languages, and formal verification.

Aside: Type Theory

Normalization theory is actually a very important topic aside from the philosophical considerations we have used to introduce it.

Type theory underpins much of automated reasoning, programming languages, and formal verification. It concerns the relationship between two things, *terms* (t) and *types* (A).

Aside: Type Theory

Normalization theory is actually a very important topic aside from the philosophical considerations we have used to introduce it.

Type theory underpins much of automated reasoning, programming languages, and formal verification. It concerns the relationship between two things, *terms* (t) and *types* (A).

Heuristically, types are 'assertions' and terms are 'evidence' of that type.

Aside: Type Theory

Normalization theory is actually a very important topic aside from the philosophical considerations we have used to introduce it.

Type theory underpins much of automated reasoning, programming languages, and formal verification. It concerns the relationship between two things, *terms* (t) and *types* (A).

Heuristically, types are 'assertions' and terms are 'evidence' of that type. This is *very* general and useful.

Aside: Type Theory

Normalization theory is actually a very important topic aside from the philosophical considerations we have used to introduce it.

Type theory underpins much of automated reasoning, programming languages, and formal verification. It concerns the relationship between two things, *terms* (t) and *types* (A).

Heuristically, types are 'assertions' and terms are 'evidence' of that type. This is *very* general and useful.

For example, the number '2' is evidence of 'Even'. That might sound silly, but many AI tools in use understand numbers in this way.

Aside: Type-assignment

How does type theory work?

Aside: Type-assignment

How does type theory work?

Formally, one has *type-assignment* rules that say what terms (read ‘evidence’) witness what types (read ‘assertion’).

Aside: Type-assignment

How does type theory work?

Formally, one has *type-assignment* rules that say what terms (read 'evidence') witness what types (read 'assertion').

For example, λ-abstraction is the type-assignment

$$\frac{x : A, \Gamma \vdash t : B}{\Gamma \vdash \lambda x.t : A \to B}$$

where A, B, and $A \to B$ are types and x, t, and $\lambda x.t$ are terms.

Aside: Type-assignment

How does type theory work?

Formally, one has *type-assignment* rules that say what terms (read 'evidence') witness what types (read 'assertion').

For example, λ-abstraction is the type-assignment

$$\frac{x : A, \Gamma \vdash t : B}{\Gamma \vdash \lambda x.t : A \to B}$$

where A, B, and $A \to B$ are types and x, t, and $\lambda x.t$ are terms. This explicates how term $\lambda x.t$ for $A \to B$ gets constructed from terms x and t for A and B, respectively.

In a series of works, Haskell Curry and William Howard observed that there is a translation between (simply typed) λ-calculus and NJ-derivations.

Aside: Type-assignment

How does type theory work?

Formally, one has *type-assignment* rules that say what terms (read 'evidence') witness what types (read 'assertion').

For example, λ-abstraction is the type-assignment

$$\frac{x : A, \Gamma \vdash t : B}{\Gamma \vdash \lambda x.t : A \to B}$$

where A, B, and $A \to B$ are types and x, t, and $\lambda x.t$ are terms. This explicates how term $\lambda x.t$ for $A \to B$ gets constructed from terms x and t for A and B, respectively.

In a series of works, Haskell Curry and William Howard observed that there is a translation between (simply typed) λ-calculus and NJ-derivations.

This is called the *Curry-Howard Correspondence*.

Aside: β-reduction

In type theory, we care about when two terms are equivalent. The Curry-Howard Correspondence tells us that this is the same as asking when two *proofs* are equivalent.

Aside: β-reduction

In type theory, we care about when two terms are equivalent. The Curry-Howard Correspondence tells us that this is the same as asking when two *proofs* are equivalent.

So this important topic of type theory actually has many of the same concerns as proof-theoretic semantics.

Aside: β-reduction

In type theory, we care about when two terms are equivalent. The Curry-Howard Correspondence tells us that this is the same as asking when two *proofs* are equivalent.

So this important topic of type theory actually has many of the same concerns as proof-theoretic semantics.

In type theory, we use 'reductions' to study equivalence,

$$(\lambda x.t)s \rightsquigarrow_\beta t[s/x]$$

Intuitively, this says, 'replace all (free) occurrences of x in t by the term s'

Aside: β-reduction

In type theory, we care about when two terms are equivalent. The Curry-Howard Correspondence tells us that this is the same as asking when two *proofs* are equivalent.

So this important topic of type theory actually has many of the same concerns as proof-theoretic semantics.

In type theory, we use 'reductions' to study equivalence,

$$(\lambda x.t)s \rightsquigarrow_\beta t[s/x]$$

Intuitively, this says, 'replace all (free) occurrences of x in t by the term s'

Apart from its use in AI, this is the foundation of an entire programming paradigm called *functional programming*. In particular, β-reduction represents execution of a program $(\lambda x.t)s$.

Aside: β-reduction

In type theory, we care about when two terms are equivalent. The Curry-Howard Correspondence tells us that this is the same as asking when two *proofs* are equivalent.

So this important topic of type theory actually has many of the same concerns as proof-theoretic semantics.

In type theory, we use 'reductions' to study equivalence,

$$(\lambda x.t)s \rightsquigarrow_\beta t[s/x]$$

Intuitively, this says, 'replace all (free) occurrences of x in t by the term s'

Apart from its use in AI, this is the foundation of an entire programming paradigm called *functional programming*. In particular, β-reduction represents execution of a program $(\lambda x.t)s$.

Translated through the Curry-Howard Correspondence, β-reduction is actually just Prawitz's reduction for $\rightarrow$ from P-tS!

Table of Contents

Dummett's Fundamental Assumption

Let's return to giving a semantics of proofs using reduction.

Dummett's Fundamental Assumption

Let's return to giving a semantics of proofs using reduction.

Actually, equivalence of terms/proofs is another way of thinking about semantics. The philosopher Willard Quine famously said, 'there's no entity without identity'

Dummett's Fundamental Assumption

Let's return to giving a semantics of proofs using reduction.

Actually, equivalence of terms/proofs is another way of thinking about semantics. The philosopher Willard Quine famously said, 'there's no entity without identity'

In *The Logical Basis of Metaphysics* (1991), Dummett writes,

> *"If we have a valid argument for a complex statement, we can construct a valid argument for it which finishes with an application of one of the introduction rules governing its principal operator."*

Dummett's Fundamental Assumption

Let's return to giving a semantics of proofs using reduction.

Actually, equivalence of terms/proofs is another way of thinking about semantics. The philosopher Willard Quine famously said, 'there's no entity without identity'

In *The Logical Basis of Metaphysics* (1991), Dummett writes,

> *"If we have a valid argument for a complex statement, we can construct a valid argument for it which finishes with an application of one of the introduction rules governing its principal operator."*

Idea: Use the corollary of Prawitz's normalization theorem, which says that an NJ-derivation in normal form ends with an introduction rule, to give a notion of validity based on the introduction rules!

Validity Concepts in Proof-theoretic Semantics

In *Validity Concepts in Proof-theoretic Semantics* (2006), Schroeder-Heister outlines three guiding principles for defining the validity of proofs:

Validity Concepts in Proof-theoretic Semantics

In *Validity Concepts in Proof-theoretic Semantics* (2006), Schroeder-Heister outlines three guiding principles for defining the validity of proofs:

- Determine a class of 'canonical' proofs that are *a priori* valid.

Validity Concepts in Proof-theoretic Semantics

In *Validity Concepts in Proof-theoretic Semantics* (2006), Schroeder-Heister outlines three guiding principles for defining the validity of proofs:

- Determine a class of 'canonical' proofs that are *a priori* valid.
- Completed† non-canonical proofs are valid if they contain‡ a canonical proof.

Validity Concepts in Proof-theoretic Semantics

In *Validity Concepts in Proof-theoretic Semantics* (2006), Schroeder-Heister outlines three guiding principles for defining the validity of proofs:

- Determine a class of 'canonical' proofs that are *a priori* valid.
- Completed[†] non-canonical proofs are valid if they contain[‡] a canonical proof.
- Uncompleted proofs are valid if any way of closing them produces a valid proof.

Validity Concepts in Proof-theoretic Semantics

In *Validity Concepts in Proof-theoretic Semantics* (2006), Schroeder-Heister outlines three guiding principles for defining the validity of proofs:

- Determine a class of 'canonical' proofs that are *a priori* valid.
- Completed[†] non-canonical proofs are valid if they contain[‡] a canonical proof.
- Uncompleted proofs are valid if any way of closing them produces a valid proof.

Validity Concepts in Proof-theoretic Semantics

In *Validity Concepts in Proof-theoretic Semantics* (2006), Schroeder-Heister outlines three guiding principles for defining the validity of proofs:

- Determine a class of 'canonical' proofs that are *a priori* valid.
- Completed[†] non-canonical proofs are valid if they contain[‡] a canonical proof.
- Uncompleted proofs are valid if any way of closing them produces a valid proof.

This requires some unpacking....

Design Choices

Unpacking Schroeder-Heister's guiding principles:

Design Choices

Unpacking Schroeder-Heister's guiding principles:

A 'canonical' proof is one whose structure we already accept.

Design Choices

Unpacking Schroeder-Heister's guiding principles:

A 'canonical' proof is one whose structure we already accept. For example, if an agent has inferential commitments $\mathscr{S}$, they ought to take $\mathscr{S}$-proofs as canonically valid.

Design Choices

Unpacking Schroeder-Heister's guiding principles:

A 'canonical' proof is one whose structure we already accept. For example, if an agent has inferential commitments $\mathscr{S}$, they ought to take $\mathscr{S}$-proofs as canonically valid.

A 'complete' proof is one that has no unwarranted steps — for example, no open hypotheses in natural deduction, no open branches in tableaux, no top sequents in sequent calculi that are not initial sequents, and so on.

Design Choices

Unpacking Schroeder-Heister's guiding principles:

A 'canonical' proof is one whose structure we already accept. For example, if an agent has inferential commitments $\mathscr{S}$, they ought to take $\mathscr{S}$-proofs as canonically valid.

A 'complete' proof is one that has no unwarranted steps — for example, no open hypotheses in natural deduction, no open branches in tableaux, no top sequents in sequent calculi that are not initial sequents, and so on.

By 'contain' we usually mean either structurally (e.g., can be seen by analysis of it) or that we have some reduction maps to probe it with.

Conceptual Blueprint

In *Validity Concepts in Proof-theoretic Semantics* (2006), Schroeder-Heister outlines three guiding principles for defining the validity of proofs:

- Determine a class of 'canonical' proofs that are *a priori* valid.
- Completed† non-canonical proofs are valid if they contain‡ a canonical proof.
- Uncompleted proofs are valid if any way of closing them produces a valid proof.

Blueprint

You can make this precise in various ways and thereby get different notions of validity.

Commitments and Definitions

If we have commitments $\mathscr{S}$, then $\mathscr{S}$-proofs are canonical.

Commitments and Definitions

If we have commitments $\mathscr{S}$, then $\mathscr{S}$-proofs are canonical.

We take the introduction rules as definitions.

Commitments and Definitions

If we have commitments $\mathscr{S}$, then $\mathscr{S}$-proofs are canonical.

We take the introduction rules as definitions.

Therefore, if a proof $\mathcal{A}$ ends with an introduction rule and its immediate sub-proof $\mathcal{A}'$ is $\mathscr{S}$-valid, then $\mathcal{A}$ ought to be $\mathscr{S}$-valid.

Reduction

We take Prawitz's reduction maps as removing extraneous data from a proof.

Reduction

We take Prawitz's reduction maps as removing extraneous data from a proof.

Therefore, if a closed proof $\mathcal{A}$ reduces to a $\mathscr{S}$-valid proof $\mathcal{A}'$, then $\mathcal{A}$ ought to be $\mathscr{S}$-valid.

Closure

We take an open proof to be $\mathscr{S}$-valid if any closure of it results in a $\mathscr{S}$-valid proof.

Closure

We take an open proof to be $\mathscr{S}$-valid if any closure of it results in a $\mathscr{S}$-valid proof. What does that mean precisely?

Closure

We take an open proof to be $\mathscr{S}$-valid if any closure of it results in a $\mathscr{S}$-valid proof. What does that mean precisely?

Closure

We take an open proof to be $\mathscr{S}$-valid if any closure of it results in a $\mathscr{S}$-valid proof. What does that mean precisely?

Let $\mathcal{A}$ have open hypotheses $\varphi_1, ..., \varphi_n$.
Idea: We close $\mathcal{A}$ by supplying $\mathscr{S}$-valid proofs $\mathcal{A}_1$, ..., $\mathcal{A}_n$ for φ_1,... ,φ_n, respectively.

Closure

We take an open proof to be $\mathscr{S}$-valid if any closure of it results in a $\mathscr{S}$-valid proof. What does that mean precisely?

Let $\mathcal{A}$ have open hypotheses $\varphi_1, ..., \varphi_n$.
Idea: We close $\mathcal{A}$ by supplying $\mathscr{S}$-valid proofs $\mathcal{A}_1$, ..., $\mathcal{A}_n$ for φ_1,... ,φ_n, respectively.

But if there are no $\mathscr{S}$-valid proofs of φ_1,...,φ_n, then $\mathcal{A}$ would be vacuously $\mathscr{S}$-valid and that seems wrong.

Closure

We take an open proof to be $\mathscr{S}$-valid if any closure of it results in a $\mathscr{S}$-valid proof. What does that mean precisely?

Let $\mathcal{A}$ have open hypotheses $\varphi_1, ..., \varphi_n$.
Idea: We close $\mathcal{A}$ by supplying $\mathscr{S}$-valid proofs $\mathcal{A}_1$, ..., $\mathcal{A}_n$ for φ_1,... ,φ_n, respectively.

But if there are no $\mathscr{S}$-valid proofs of φ_1,...,φ_n, then $\mathcal{A}$ would be vacuously $\mathscr{S}$-valid and that seems wrong.

Idea: We close $\mathcal{A}$ by considering arbitrary $\mathscr{S}' \supseteq \mathscr{S}$ for which there are $\mathscr{S}'$-valid proofs $\mathcal{A}_1$, ..., $\mathcal{A}_n$ for φ_1,... ,φ_n, respectively, yielding an argument $\mathscr{A}'$.

Closure

We take an open proof to be $\mathscr{S}$-valid if any closure of it results in a $\mathscr{S}$-valid proof. What does that mean precisely?

Let $\mathcal{A}$ have open hypotheses $\varphi_1, ..., \varphi_n$.
Idea: We close $\mathcal{A}$ by supplying $\mathscr{S}$-valid proofs $\mathcal{A}_1$, ..., $\mathcal{A}_n$ for φ_1,... ,φ_n, respectively.

But if there are no $\mathscr{S}$-valid proofs of φ_1,...,φ_n, then $\mathcal{A}$ would be vacuously $\mathscr{S}$-valid and that seems wrong.

Idea: We close $\mathcal{A}$ by considering arbitrary $\mathscr{S}' \supseteq \mathscr{S}$ for which there are $\mathscr{S}'$-valid proofs $\mathcal{A}_1$, ..., $\mathcal{A}_n$ for φ_1,... ,φ_n, respectively, yielding an argument $\mathscr{A}'$. For $\mathcal{A}$ to be $\mathscr{S}$-valid we need $\mathcal{A}'$ to be $\mathscr{S}'$-valid.

Proof-theoretic Validity

Altogether this delivers the following definition (with respect to a basis $\mathfrak{B}$):

Definition ($\mathscr{S}$-valid)

Given an atomic system $\mathscr{S} \in \mathfrak{B}$:

- Every $\mathscr{S}$-derivation $\mathcal{A}$ is valid
- A closed, normal argument $\mathcal{A}$ is $\mathscr{S}$-valid if its immediate sub-derivation is $\mathscr{S}$-valid
- A closed, non-normal argument $\mathcal{A}$ is $\mathscr{S}$-valid if it reduces to an $\mathscr{S}$-valid argument
- An open argument $\mathcal{A}$ is $\mathscr{S}$-valid iff, for any $\mathscr{S}' \supseteq \mathscr{S}$, the result of extending the argument $\mathcal{A}$ with $\mathscr{S}'$-valid arguments for all the open hypotheses results in an $\mathscr{S}'$-valid argument.

An argument $\mathcal{A}$ is valid if it is $\mathscr{S}$-valid for every atomic system $\mathscr{S} \in \mathfrak{B}$.

Axiomatization

So we have a definition of validity. We know what it means to say that a 'proof' $\mathcal{A}$ is valid.

Axiomatization

So we have a definition of validity. We know what it means to say that a 'proof' $\mathcal{A}$ is valid.

Can we give an axiomatic account of it? Does it depend on the basis $\mathfrak{B}$?

Axiomatization

So we have a definition of validity. We know what it means to say that a 'proof' $\mathcal{A}$ is valid.

Can we give an axiomatic account of it? Does it depend on the basis $\mathfrak{B}$?

More precisely: We want a set of natural deduction rules L such that $\mathcal{A}$ is valid iff it is an L-derivation.

Axiomatization

So we have a definition of validity. We know what it means to say that a 'proof' $\mathcal{A}$ is valid.

Can we give an axiomatic account of it? Does it depend on the basis $\mathfrak{B}$?

More precisely: We want a set of natural deduction rules L such that $\mathcal{A}$ is valid iff it is an L-derivation.

It seems like NJ should do the job.

Soundness and Completeness?

Theorem (Soundness)

Every NJ*-derivation* $\mathcal{D}$ *is valid.*

Proof.

This is a direct consequence of Prawitz's normalization theorem. □

Soundness and Completeness?

Theorem (Soundness)

Every NJ*-derivation* $\mathcal{D}$ *is valid.*

Proof.

This is a direct consequence of Prawitz's normalization theorem. □

Prawitz's Conjecture (1973)

To every valid argument $\mathcal{A}$ there 'corresponds' an NJ-derivation.

Soundness and Completeness?

Theorem (Soundness)

Every NJ*-derivation* $\mathcal{D}$ *is valid.*

Proof.

This is a direct consequence of Prawitz's normalization theorem. □

Prawitz's Conjecture (1973)

To every valid argument $\mathcal{A}$ there 'corresponds' an NJ-derivation.

This is more controversial, but the short answer is:

Soundness and Completeness?

Theorem (Soundness)

Every NJ*-derivation* $\mathcal{D}$ *is valid.*

Proof.

This is a direct consequence of Prawitz's normalization theorem. □

Prawitz's Conjecture (1973)

To every valid argument $\mathcal{A}$ there 'corresponds' an NJ-derivation.

This is more controversial, but the short answer is: no.

Soundness and Completeness?

Theorem (Soundness)

Every NJ*-derivation* $\mathcal{D}$ *is valid.*

Proof.

This is a direct consequence of Prawitz's normalization theorem. □

Prawitz's Conjecture (1973)

To every valid argument $\mathcal{A}$ there 'corresponds' an NJ-derivation.

This is more controversial, but the short answer is: no.

Does this mean that the introduction rules of NJ do *not* define the connectives of intuitionistic logic?

Table of Contents

Variations?

We have made several design choices that could have led to this incompleteness.

Variations?

We have made several design choices that could have led to this incompleteness. For example:

- our understanding of ‘canonical’ proof

Variations?

We have made several design choices that could have led to this incompleteness. For example:

- our understanding of ‘canonical’ proof
- our choice of reductions

Variations?

We have made several design choices that could have led to this incompleteness. For example:

- our understanding of ‘canonical’ proof
- our choice of reductions
- prioritizing the introduction rules (over, say, the elimination rules)

Variations?

We have made several design choices that could have led to this incompleteness. For example:

- our understanding of ‘canonical’ proof
- our choice of reductions
- prioritizing the introduction rules (over, say, the elimination rules)

Variations?

We have made several design choices that could have led to this incompleteness. For example:

- our understanding of 'canonical' proof
- our choice of reductions
- prioritizing the introduction rules (over, say, the elimination rules)

These were made in a principled way, but other choices are available.

Importantly, we can do proof-theoretic validity based on the elimination rules.

Variations?

We have made several design choices that could have led to this incompleteness. For example:

- our understanding of 'canonical' proof
- our choice of reductions
- prioritizing the introduction rules (over, say, the elimination rules)

These were made in a principled way, but other choices are available.

Importantly, we can do proof-theoretic validity based on the elimination rules. Does that correspond to NJ better? Yes!

Variations?

We have made several design choices that could have led to this incompleteness. For example:

- our understanding of 'canonical' proof
- our choice of reductions
- prioritizing the introduction rules (over, say, the elimination rules)

These were made in a principled way, but other choices are available.

Importantly, we can do proof-theoretic validity based on the elimination rules. Does that correspond to NJ better? Yes! (sort of)

Stay tuned.