Harmony, Local Soundness, and Local Completeness

April 6, 2026

Plan

1. What to take at CMU if you want more logic, PL, and type theory
2. Other ways to plug into the CMU PL space
3. Today's lecture: local soundness and local completeness in natural deduction

CMU classes and the PL ecosystem

If you liked this material, CMU has a lot more of it

- We only saw a small part of PL
- CMU has classes that go in *deeper* on:
  - logic and proof systems
  - type systems and semantics
  - theorem proving and proof assistants
  - program verification and automated reasoning
- Roughly speaking:
  - **15-317**: more logic systems and proof theory
  - **15-312**: core PL theory and semantics
  - **15-311**: symbolic logic + Lean + SAT/SMT/theorem proving

15-317 / 15-657: Constructive Logic

- Best follow-up if you liked the *logic* side of what we did
- Official topics include:
  - natural deduction and sequent calculus
  - harmony, cut elimination, inversion
  - constructive vs. classical logic
  - theorem proving, logic programming, Prolog
  - linear, modal, and other substructural logics

15-312 / 15-652: Foundations of Programming Languages

- Best follow-up if you liked the *PL semantics / type theory* side
- Big themes:
  - statics, dynamics, and safety
  - binding and substitution
  - functions, products, sums, recursive types
  - polymorphism, representation independence, control, effects, concurrency
  - late-semester topics connect back to proofs-as-programs and dependent typing
- This is one of the central CMU PL classes
- Fall 2026 listing names **Robert Harper** as instructor
- If you want deeper type theory later, this is one of the main gateways

15-311 / 15-653: Logic and Mechanized Reasoning

- Great option if you want more *formal reasoning tools*
- Focuses on three levels at once:
  - theory: propositional and first-order logic
  - implementation: representing logic in Lean / functional code
  - application: SAT solvers, SMT solvers, theorem provers
- Spring 2026 topics include Lean, DPLL/CDCL, unification, SMT, skolemization, and automated theorem proving

Other classes to know about

- **15-414 Bug Catching: Automated Program Verification**
  - specifications, correctness proofs, model checking, Why3, verified implementations
- **15-814 Types and Programming Languages**
  - graduate types course; Fall 2025 listing is taught by **Frank Pfenning**
- **15-816 Advanced Topics in Logic: Automated Reasoning and Satisfiability**
  - research-facing logic / SAT / proof technology course
- **15-411 Compiler Design**
  - hard, but a classic class for PL people because it connects theory to real language implementation
- Also on the official PL elective list: **15-316**, **15-413**, **15-417**, **15-714**

Beyond classes: ways to participate in the CMU PL / logic space

- **Hoskinson Center for Formal Mathematics**
  - Baker Hall 139
  - formal math, Lean, theorem proving, automated reasoning
  - great place to meet people working around proof assistants and formalization
- **Principles of Programming Seminar / PL talks**
  - CMU regularly hosts programming languages seminars and talks

Local soundness and local completeness

Harmony: the big idea

- In natural deduction, a connective is supposed to be determined by its **introduction rule(s)**
- The **elimination rule(s)** should match that meaning exactly
- If eliminations are too strong, we can get information we never really put in
- If eliminations are too weak, we cannot recover all the information the connective was meant to contain

Harmony: the big idea

- In natural deduction, a connective is supposed to be determined by its **introduction rule(s)**
- The **elimination rule(s)** should match that meaning exactly
- If eliminations are too strong, we can get information we never really put in
- If eliminations are too weak, we cannot recover all the information the connective was meant to contain

Harmony = two local checks

- **Local soundness**: eliminations are not too strong
- **Local completeness**: eliminations are not too weak

Local soundness: “introduce, then eliminate” should be redundant

Suppose we *just built* a proof of a connective using its introduction rule.

Then if we immediately eliminate that connective, we should not learn anything new.

Intuition

An introduction followed immediately by an elimination should reduce to a more direct proof.

Local soundness: "introduce, then eliminate" should be redundant

Suppose we *just built* a proof of a connective using its introduction rule.

Then if we immediately eliminate that connective, we should not learn anything new.

Intuition

An introduction followed immediately by an elimination should reduce to a more direct proof.

$$\textit{constructor followed by destructor} = \textit{no real progress}$$

Local completeness: eliminations should let us recover the connective

Now go in the other direction.

Start from an *arbitrary* proof of a proposition with primary connective $\star$.

If the elimination rules are good enough, we should be able to:

1. eliminate the connective to expose all of its information, then
2. use the introduction rule(s) to rebuild the original proposition

Local completeness: eliminations should let us recover the connective

Now go in the other direction.

Start from an *arbitrary* proof of a proposition with primary connective $\star$.

If the elimination rules are good enough, we should be able to:

1. eliminate the connective to expose all of its information, then
2. use the introduction rule(s) to rebuild the original proposition

Intuition

The elimination rules collectively expose exactly the information that the connective contains.

Example 1: conjunction $A \wedge B$

Introduction and elimination rules

$$\frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B}\ (\wedge I) \qquad \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash A}\ (\wedge E_1) \qquad \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash B}\ (\wedge E_2)$$

Example 1: conjunction $A \wedge B$

Introduction and elimination rules

$$\frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B}\ (\wedge I) \qquad \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash A}\ (\wedge E_1) \qquad \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash B}\ (\wedge E_2)$$

Local soundness

$$\frac{\frac{D}{\Gamma \vdash A} \qquad \frac{E}{\Gamma \vdash B}}{\Gamma \vdash A \wedge B}\ \wedge I \quad \text{then} \ \wedge E_1 \quad \rightsquigarrow \quad D$$

Likewise, introducing $A \wedge B$ and then applying $\wedge E_2$ reduces directly to E.

Example 1: conjunction $A \wedge B$

Introduction and elimination rules

$$\frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B}\ (\wedge I) \qquad \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash A}\ (\wedge E_1) \qquad \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash B}\ (\wedge E_2)$$

Local soundness

$$\frac{\frac{D}{\Gamma \vdash A} \qquad \frac{E}{\Gamma \vdash B}}{\Gamma \vdash A \wedge B} \wedge I \qquad \text{then } \wedge E_1 \qquad \rightsquigarrow \qquad D$$

Likewise, introducing $A \wedge B$ and then applying $\wedge E_2$ reduces directly to E.
Local completeness From any proof of $A \wedge B$, extract A and B with $\wedge E_1$ and $\wedge E_2$, then re-introduce $A \wedge B$ with $\wedge I$.

Example 2: implication $A \supset B$

Rules

$$\frac{\Gamma, A \vdash B}{\Gamma \vdash A \supset B}\ (\supset I) \qquad\qquad \frac{\Gamma \vdash A \supset B \quad \Gamma \vdash A}{\Gamma \vdash B}\ (\supset E)$$

Example 2: implication $A \supset B$

Rules

$$\frac{\Gamma, A \vdash B}{\Gamma \vdash A \supset B}\ (\supset I) \qquad\qquad \frac{\Gamma \vdash A \supset B \quad \Gamma \vdash A}{\Gamma \vdash B}\ (\supset E)$$

Local soundness

- If we introduce $A \supset B$ by proving B from hypothesis A
- and then immediately eliminate it using a proof of A
- we should get the same thing as just **substituting** that proof of A into the derivation of B

Example 2: implication $A \supset B$

Rules

$$\frac{\Gamma, A \vdash B}{\Gamma \vdash A \supset B}\ (\supset I) \qquad\qquad \frac{\Gamma \vdash A \supset B \quad \Gamma \vdash A}{\Gamma \vdash B}\ (\supset E)$$

Local soundness

- If we introduce $A \supset B$ by proving B from hypothesis A
- and then immediately eliminate it using a proof of A
- we should get the same thing as just **substituting** that proof of A into the derivation of B

Intuition

A proof of $A \supset B$ is a way of turning proofs of A into proofs of B. Applying it immediately to a proof of A just performs that promised transformation.

Example 3: disjunction $A \vee B$

Rules

$$\frac{\Gamma \vdash A}{\Gamma \vdash A \vee B}\ (\vee I_1) \qquad\qquad \frac{\Gamma \vdash B}{\Gamma \vdash A \vee B}\ (\vee I_2)$$

$$\frac{\Gamma \vdash A \vee B \quad \Gamma, A \vdash C \quad \Gamma, B \vdash C}{\Gamma \vdash C}\ (\vee E)$$

Example 3: disjunction $A \vee B$

Rules

$$\frac{\Gamma \vdash A}{\Gamma \vdash A \vee B} \ (\vee I_1) \qquad \frac{\Gamma \vdash B}{\Gamma \vdash A \vee B} \ (\vee I_2)$$

$$\frac{\Gamma \vdash A \vee B \quad \Gamma, A \vdash C \quad \Gamma, B \vdash C}{\Gamma \vdash C} \ (\vee E)$$

Why the elimination is more complicated

- From $A \vee B$ alone, we do *not* know whether A or B holds
- So elimination must reason by cases

Example 3: disjunction $A \vee B$

Rules

$$\frac{\Gamma \vdash A}{\Gamma \vdash A \vee B}\ (\vee I_1) \qquad \frac{\Gamma \vdash B}{\Gamma \vdash A \vee B}\ (\vee I_2)$$

$$\frac{\Gamma \vdash A \vee B \quad \Gamma, A \vdash C \quad \Gamma, B \vdash C}{\Gamma \vdash C}\ (\vee E)$$

Why the elimination is more complicated

- From $A \vee B$ alone, we do *not* know whether A or B holds
- So elimination must reason by cases

Local soundness If the disjunction was introduced on the left, case analysis reduces to the A-branch. If it was introduced on the right, case analysis reduces to the B-branch.

Why this matters

- Harmony is a sanity check on the definition of a connective
- Local soundness says: eliminations do not smuggle in extra power
- Local completeness says: eliminations are expressive enough to recover what the connective means
- Together they explain why the usual natural deduction rules feel "just right"

Why this matters

- Harmony is a sanity check on the definition of a connective
- Local soundness says: eliminations do not smuggle in extra power
- Local completeness says: eliminations are expressive enough to recover what the connective means
- Together they explain why the usual natural deduction rules feel "just right"

Suspicious rule pattern

If someone proposes an elimination rule that extracts more than what the introductions justify, the connective definition is probably broken.

Why this matters

- Harmony is a sanity check on the definition of a connective
- Local soundness says: eliminations do not smuggle in extra power
- Local completeness says: eliminations are expressive enough to recover what the connective means
- Together they explain why the usual natural deduction rules feel "just right"

Suspicious rule pattern

If someone proposes an elimination rule that extracts more than what the introductions justify, the connective definition is probably broken.

Introduction tells us what counts as a proof; elimination tells us how that proof may be used.

Takeaway

- For conjunction, implication, disjunction, truth, and falsehood, the standard natural deduction rules are harmonious
- Local soundness and completeness are the *local* witnesses of that fact
- Later, these ideas connect to normalization, proof theory, and eventually to programs and type systems
- But even at a high level, the message is simple:

good elimination rules exactly match what the introduction rules put in