Constructive Logic

Frank Pfenning
Carnegie Mellon University

$\boxed{\text{Draft of January 9, 2003}}$

Material for the course *Constructive Logic* at Carnegie Mellon University, Fall 2000. Material for this course is available at
`http://www.cs.cmu.edu/~fp/courses/logic/`.

Please send comments to `fp@cs.cmu.edu`

This material is in rough draft form and is likely to contain errors. Furthermore, citations are in no way adequate or complete. Please do not cite or distribute this document.

This work was supported in part by the University Education Council at Carnegie Mellon University and by NSF Grant CCR-9619684.

Contents

Chapter 2

Propositional Logic

The goal of this chapter is to develop the two principal notions of logic, namely propositions and proofs. There is no universal agreement about the proper foundations for these notions. One approach, which has been particularly successful for applications in computer science, is to understand the meaning of a proposition by understanding its proofs. In the words of Martin-Löf [ML96, Page 27]:

> *The meaning of a proposition is determined by [...] what counts as a verification of it.*

In this chapter we apply Martin-Löf's approach, which follows a rich philosophical tradition, to explain the basic propositional connectives. We will see later that universal and existential quantifiers and types such as natural numbers, lists, or trees naturally fit into the same framework.

2.1 Judgments and Propositions

The cornerstone of Martin-Löf's foundation of logic is a clear separation of the notions of judgment and proposition. A *judgment* is something we may know, that is, an object of knowledge. A judgment is *evident* if we in fact know it.

We make a judgment such as "*it is raining*", because we have evidence for it. In everyday life, such evidence is often immediate: we may look out the window and see that it is raining. In logic, we are concerned with situation where the evidence is indirect: we deduce the judgment by making correct inferences from other evident judgments. In other words: a judgment is evident if we have a proof for it.

The most important judgment form in logic is "A *is true*", where A is a proposition. In order to reason correctly, we therefore need a second judgment form "A *is a proposition*". But there are many others that have been studied extensively. For example, "A *is false*", "A *is true at time* t" (from temporal logic), "*A is necessarily true*" (from modal logic), "*program M has type τ*" (from programming languages), etc.

Returning to the first two judgments, let us try to explain the meaning of conjunction. We write A *prop* for the judgment "*A is a proposition*" and A *true* for the judgment "*A is true*" (presupposing that A *prop*). Given propositions A and B, we want to form the compound proposition "*A and B*", written more formally as $A \wedge B$. We express this in the following inference rule:

$$\frac{A\ \mathit{prop} \qquad B\ \mathit{prop}}{A \wedge B\ \mathit{prop}}\ \wedge F$$

This rule allows us to conclude that $A \wedge B$ *prop* if we already know that A *prop* and B *prop*. In this inference rule, A and B are *schematic variables*, and $\wedge F$ is the name of the rule (which is short for "conjunction formation"). The general form of an inference rule is

$$\frac{J_1\ \ldots\ J_n}{J}\ \mathit{name}$$

where the judgments $J_1, \ldots, J_n$ are called the *premises*, the judgment J is called the *conclusion*. In general, we will use letters J to stand for judgments, while A, B, and C are reserved for propositions.

Once the rule of conjunction formation ($\wedge F$) has been specified, we know that $A \wedge B$ is a proposition, if A and B are. But we have not yet specified what it *means*, that is, what counts as a verification of $A \wedge B$. This is accomplished by the following inference rule:

$$\frac{A\ \mathit{true} \qquad B\ \mathit{true}}{A \wedge B\ \mathit{true}}\ \wedge I$$

Here the name $\wedge I$ stands for "conjunction introduction", since the conjunction is introduced in the conclusion. We take this as specifying the meaning of $A \wedge B$ completely. So what can be deduce if we know that $A \wedge B$ is true? By the above rule, to have a verification for $A \wedge B$ means to have verifications for A and B. Hence the following two rules are justified:

$$\frac{A \wedge B\ \mathit{true}}{A\ \mathit{true}}\ \wedge E_L \qquad\qquad \frac{A \wedge B\ \mathit{true}}{B\ \mathit{true}}\ \wedge E_R$$

The name $\wedge E_L$ stands for "left conjunction elimination", since the conjunction in the premise has been eliminated in the conclusion. Similarly $\wedge E_R$ stands for "right conjunction elimination".

We will later see what precisely is required in order to guarantee that the formation, introduction, and elimination rules for a connective fit together correctly. For now, we will informally argue the correctness of the elimination rules.

As a second example we consider the proposition "*truth*" written as ⊤.

$$\frac{}{\top \; \mathit{prop}}\;\top F$$

Truth should always be true, which means its introduction rule has no premises.

$$\frac{}{\top \; \mathit{true}}\;\top I$$

Consequently, we have no information if we know ⊤ *true*, so there is no elimination rule.

A conjunction of two propositions is characterized by one introduction rule with two premises, and two corresponding elimination rules. We may think of truth as a conjunction of zero propositions. By analogy it should then have one introduction rule with zero premises, and zero corresponding elimination rules. This is precisely what we wrote out above.

2.2 Hypothetical Judgments

Consider the following derivation, for some arbitrary propositions A, B, and C:

$$\cfrac{\cfrac{A \wedge (B \wedge C) \; \mathit{true}}{B \wedge C \; \mathit{true}}\;\wedge E_R}{B \; \mathit{true}}\;\wedge E_L$$

Have we actually proved anything here? At first glance it seems that cannot be the case: B is an arbitrary proposition; clearly we should not be able to prove that it is true. Upon closer inspection we see that all inferences are correct, but the first judgment $A \wedge (B \wedge C)$ has not been justified. We can extract the following knowledge:

> *From the assumption that $A \wedge (B \wedge C)$ is true, we deduce that B must be true.*

This is an example of a *hypothetical judgment*, and the figure above is an *hypothetical derivation.* In general, we may have more than one assumption, so a hypothetical derivation has the form

$$\begin{array}{c} J_1 \quad \cdots \quad J_n \\ \vdots \\ J \end{array}$$

where the judgments $J_1, \ldots, J_n$ are unproven assumptions, and the judgment J is the conclusion. Note that we can always substitute a proof for any hypothesis J_i to eliminate the assumption. We call this the *substitution principle* for hypotheses.

Many mistakes in reasoning arise because dependencies on some hidden assumptions are ignored. When we need to be explicit, we write $J_1, \ldots, J_n \vdash J$ for the hypothetical judgment which is established by the hypothetical derivation above. We may refer to $J_1, \ldots, J_n$ as the antecedents and J as the succedent of the hypothetical judgment.

One has to keep in mind that hypotheses may be used more than once, or not at all. For example, for arbitrary propositions A and B,

$$\dfrac{\dfrac{A \wedge B\ \mathit{true}}{B\ \mathit{true}}\wedge E_R \qquad \dfrac{A \wedge B\ \mathit{true}}{A\ \mathit{true}}\wedge E_L}{B \wedge A\ \mathit{true}}\wedge I$$

can be seen a hypothetical derivation of $A \wedge B\ \mathit{true} \vdash B \wedge A\ \mathit{true}$.

With hypothetical judgments, we can now explain the meaning of implication "*A implies B*" or "*if A then B*" (more formally: $A \supset B$). First the formation rule:

$$\dfrac{A\ \mathit{prop} \qquad\qquad B\ \mathit{prop}}{A \supset B\ \mathit{prop}}\supset F$$

Next, the introduction rule: $A \supset B$ is true, if B is true under the assumption that A is true.

$$\begin{array}{c} \dfrac{}{A\ \mathit{true}}u \\ \vdots \\ \dfrac{B\ \mathit{true}}{A \supset B\ \mathit{true}}\supset I^u \end{array}$$

The tricky part of this rule is the label u. If we omit this annotation, the rule would read

$$\begin{array}{c} A\ \mathit{true} \\ \vdots \\ \dfrac{B\ \mathit{true}}{A \supset B\ \mathit{true}}\supset I \end{array}$$

which would be incorrect: it looks like a derivation of $A \supset B\ \ \mathit{true}$ from the hypothesis $A\ \mathit{true}$. But the assumption $A\ \mathit{true}$ is introduced in the process of proving $A \supset B\ \mathit{true}$; the conclusion should not depend on it! Therefore we label uses of the assumption with a new name u, and the corresponding inference which introduced this assumption into the derivation with the same label u.

As a concrete example, consider the following proof of $A \supset (B \supset (A \wedge B))$.

$$\dfrac{\dfrac{\dfrac{\dfrac{}{A\ \mathit{true}}u \qquad \dfrac{}{B\ \mathit{true}}w}{A \wedge B\ \mathit{true}}\wedge I}{B \supset (A \wedge B)\ \mathit{true}}\supset I^w}{A \supset (B \supset (A \wedge B))\ \mathit{true}}\supset I^u$$

Note that this derivation is not hypothetical (it does not depend on any assumptions). The assumption A *true* labeled u is discharged in the last inference, and the assumption B *true* labeled w is discharged in the second-to-last inference. It is critical that a discharged hypothesis is no longer available for reasoning, and that all labels introduced in a derivation are distinct.

Finally, we consider what the elimination rule for implication should say. By the only introduction rule, having a proof of $A \supset B$ *true* means that we have a hypothetical proof of B *true* from A *true*. By the substitution principle, if we also have a proof of A *true* then we get a proof of B *true*.

$$\frac{A \supset B\ \mathit{true} \qquad A\ \mathit{true}}{B\ \mathit{true}}\supset\! E$$

This completes the rule concerning implication.

With the rules so far, we can write out proofs of simple properties concerning conjunction and implication. The first expresses that conjunction is commutative—intuitively, an obvious property.

$$\dfrac{\dfrac{\dfrac{\dfrac{}{A \wedge B\ \mathit{true}}u}{B\ \mathit{true}}\wedge E_R \qquad \dfrac{\dfrac{}{A \wedge B\ \mathit{true}}u}{A\ \mathit{true}}\wedge E_L}{B \wedge A\ \mathit{true}}\wedge I^u}{(A \wedge B) \supset (B \wedge A)\ \mathit{true}}\supset\! I$$

When we construct such a derivation, we generally proceed by a combination of bottom-up and top-down reasoning. The next example is a distributivity law, allowing us to move implications over conjunctions. This time, we show the partial proofs in each step. Of course, other sequences of steps in proof constructions are also possible.

$$\begin{array}{c}\vdots\\ (A \supset (B \wedge C)) \supset ((A \supset B) \wedge (A \supset C))\ \mathit{true}\end{array}$$

First, we use the implication introduction rule bottom-up.

$$\dfrac{\begin{array}{c}\dfrac{}{A \supset (B \wedge C)\ \mathit{true}}u\\ \vdots\\ (A \supset B) \wedge (A \supset C)\ \mathit{true}\end{array}}{(A \supset (B \wedge C)) \supset ((A \supset B) \wedge (A \supset C))\ \mathit{true}}\supset\! I^u$$

Next, we use the conjunction introduction rule bottom-up.

$$\dfrac{\dfrac{\begin{array}{c}\dfrac{}{A \supset (B \wedge C)\ \mathit{true}}\,u\\ \vdots \\ A \supset B\ \mathit{true}\end{array} \qquad \begin{array}{c}\dfrac{}{A \supset (B \wedge C)\ \mathit{true}}\,u\\ \vdots \\ A \supset C\ \mathit{true}\end{array}}{(A \supset B) \wedge (A \supset C)\ \mathit{true}}\,\wedge I}{(A \supset (B \wedge C)) \supset ((A \supset B) \wedge (A \supset C))\ \mathit{true}}\,\supset I^u$$

We now pursue the left branch, again using implication introduction bottom-up.

$$\dfrac{\dfrac{\dfrac{\begin{array}{c}\dfrac{}{A \supset (B \wedge C)\ \mathit{true}}\,u \quad \dfrac{}{A\ \mathit{true}}\,w\\ \vdots \\ B\ \mathit{true}\end{array}}{A \supset B\ \mathit{true}}\,\supset I^w \qquad \begin{array}{c}\dfrac{}{A \supset (B \wedge C)\ \mathit{true}}\,u\\ \vdots \\ A \supset C\ \mathit{true}\end{array}}{(A \supset B) \wedge (A \supset C)\ \mathit{true}}\,\wedge I}{(A \supset (B \wedge C)) \supset ((A \supset B) \wedge (A \supset C))\ \mathit{true}}\,\supset I^u$$

Note that the hypothesis A *true* is available only in the left branch, but not in the right one: it is discharged at the inference $\supset I^w$. We now switch to top-down reasoning, taking advantage of implication elimination.

$$\dfrac{\dfrac{\dfrac{\begin{array}{c}\dfrac{\dfrac{}{A \supset (B \wedge C)\ \mathit{true}}\,u \quad \dfrac{}{A\ \mathit{true}}\,w}{B \wedge C\ \mathit{true}}\,\supset E\\ \vdots \\ B\ \mathit{true}\end{array}}{A \supset B\ \mathit{true}}\,\supset I^w \qquad \begin{array}{c}\dfrac{}{A \supset (B \wedge C)\ \mathit{true}}\,u\\ \vdots \\ A \supset C\ \mathit{true}\end{array}}{(A \supset B) \wedge (A \supset C)\ \mathit{true}}\,\wedge I}{(A \supset (B \wedge C)) \supset ((A \supset B) \wedge (A \supset C))\ \mathit{true}}\,\supset I^u$$

Now we can close the gap in the left-hand side by conjunction elimination.

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{A \supset (B \wedge C)\ \mathit{true}}\,u \qquad \dfrac{}{A\ \mathit{true}}\,w}{B \wedge C\ \mathit{true}}\supset\! E}{B\ \mathit{true}}\wedge E_L}{A \supset B\ \mathit{true}}\supset\! I^w \qquad \begin{array}{c}\dfrac{}{A \supset (B \wedge C)\ \mathit{true}}\,u \\ \vdots \\ A \supset C\ \mathit{true}\end{array}}{(A \supset B) \wedge (A \supset C)\ \mathit{true}}\wedge I}{(A \supset (B \wedge C)) \supset ((A \supset B) \wedge (A \supset C))\ \mathit{true}}\supset\! I^u
$$

The right premise of the conjunction introduction can be filled in analogously. We skip the intermediate steps and only show the final derivation.

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{A \supset (B \wedge C)\ \mathit{true}}\,u \qquad \dfrac{}{A\ \mathit{true}}\,w}{B \wedge C\ \mathit{true}}\supset\! E}{B\ \mathit{true}}\wedge E_L}{A \supset B\ \mathit{true}}\supset\! I^w \qquad \dfrac{\dfrac{\dfrac{\dfrac{}{A \supset (B \wedge C)\ \mathit{true}}\,u \qquad \dfrac{}{A\ \mathit{true}}\,v}{B \wedge C\ \mathit{true}}\supset\! E}{C\ \mathit{true}}\wedge E_R}{A \supset C\ \mathit{true}}\supset\! I^v}{(A \supset B) \wedge (A \supset C)\ \mathit{true}}\wedge I}{(A \supset (B \wedge C)) \supset ((A \supset B) \wedge (A \supset C))\ \mathit{true}}\supset\! I^u
$$

2.3 Disjunction and Falsehood

So far we have explained the meaning of conjunction, truth, and implication. The disjunction "A *or* B" (written as $A \vee B$) is more difficult, but does not require any new judgment forms.

$$
\dfrac{A\ \mathit{prop} \qquad B\ \mathit{prop}}{A \vee B\ \mathit{prop}}\vee F
$$

Disjunction is characterized by two introduction rules: $A \vee B$ is true, if either A or B is true.

$$
\dfrac{A\ \mathit{true}}{A \vee B\ \mathit{true}}\vee I_L \qquad\qquad \dfrac{B\ \mathit{true}}{A \vee B\ \mathit{true}}\vee I_R
$$

Now it would be incorrect to have an elimination rule such as

$$
\dfrac{A \vee B\ \mathit{true}}{A\ \mathit{true}}\vee E_L?
$$

because even if we know that $A \vee B$ is true, we do not know whether the disjunct A or the disjunct B is true. Concretely, with such a rule we could derive the truth of *every* proposition A as follows:

$$\dfrac{\dfrac{\dfrac{}{B\ \mathit{true}}\,u}{B \supset B\ \mathit{true}}\supset I^u \qquad \dfrac{\dfrac{\dfrac{\dfrac{}{B \supset B\ \mathit{true}}\,w}{A \lor (B \supset B)\ \mathit{true}}\lor I_R}{A\ \mathit{true}}\lor E_L?}{(B \supset B) \supset A\ \mathit{true}}\supset I^w}{A\ \mathit{true}}\supset E$$

Thus we take a different approach. If we know that $A \lor B$ is true, we must consider two cases: A *true* and B *true*. If we can prove a conclusion C *true* in both cases, then C must be true! Written as an inference rule:

$$\dfrac{A \lor B\ \mathit{true} \qquad \begin{array}{c}\dfrac{}{A\ \mathit{true}}\,u\\ \vdots\\ C\ \mathit{true}\end{array} \qquad \begin{array}{c}\dfrac{}{B\ \mathit{true}}\,w\\ \vdots\\ C\ \mathit{true}\end{array}}{C\ \mathit{true}}\lor E^{u,w}$$

Note that we use once again the mechanism of hypothetical judgments. In the proof of the second premise we may use the assumption A *true* labeled u, in the proof of the third premise we may use the assumption B *true* labeled w. Both are discharged at the disjunction elimination rule.

Let us justify the conclusion of this rule more explicitly. By the first premise we know $A \lor B$ *true*. The premises of the two possible introduction rules are A *true* and B *true*. In case A *true* we conclude C *true* by the substitution principle and the second premise: we substitute the proof of A *true* for any use of the assumption labeled u in the hypothetical derivation. The case for B *true* is symmetric, using the hypothetical derivation in the third premise.

Because of the complex nature of the elimination rule, reasoning with disjunction is more difficult than with implication and conjunction. As a simple example, we prove the commutativity of disjunction.

$$\begin{array}{c}\vdots\\ (A \lor B) \supset (B \lor A)\ \mathit{true}\end{array}$$

We begin with an implication introduction.

$$\dfrac{\begin{array}{c}\dfrac{}{A \lor B\ \mathit{true}}\,u\\ \vdots\\ B \lor A\ \mathit{true}\end{array}}{(A \lor B) \supset (B \lor A)\ \mathit{true}}\supset I^u$$

At this point we cannot use either of the two disjunction introduction rules. The problem is that neither B nor A follow from our assumption $A \lor B$! So first we need to distinguish the two cases via the rule of disjunction elimination.

$$\cfrac{\cfrac{\dfrac{}{A \lor B\ \mathit{true}}\,u \qquad \begin{matrix}\dfrac{}{A\ \mathit{true}}\,v \\ \vdots \\ B \lor A\ \mathit{true}\end{matrix} \qquad \begin{matrix}\dfrac{}{B\ \mathit{true}}\,w \\ \vdots \\ B \lor A\ \mathit{true}\end{matrix}}{B \lor A\ \mathit{true}}\,{\lor}E^{v,w}}{(A \lor B) \supset (B \lor A)\ \mathit{true}}\,{\supset}I^{u}$$

The assumption labeled u is still available for each of the two proof obligations, but we have omitted it, since it is no longer needed.

Now each gap can be filled in directly by the two disjunction introduction rules.

$$\cfrac{\cfrac{\dfrac{}{A \lor B\ \mathit{true}}\,u \qquad \dfrac{\dfrac{}{A\ \mathit{true}}\,v}{B \lor A\ \mathit{true}}\,{\lor}I_R \qquad \dfrac{\dfrac{}{B\ \mathit{true}}\,w}{B \lor A\ \mathit{true}}\,{\lor}I_L}{B \lor A\ \mathit{true}}\,{\lor}E^{v,w}}{(A \lor B) \supset (B \lor A)\ \mathit{true}}\,{\supset}I^{u}$$

This concludes the discussion of disjunction. Falsehood (written as $\bot$, sometimes called absurdity) is a proposition that should have no proof! Therefore there are no introduction rules, although we of course have the standard formation rule.

$$\frac{}{\bot\ \mathit{prop}}\,{\bot}F$$

Since there cannot be a proof of $\bot$ *true*, it is sound to conclude the truth of any arbitrary proposition if we know $\bot$ *true*. This justifies the elimination rule

$$\frac{\bot\ \mathit{true}}{C\ \mathit{true}}\,{\bot}E$$

We can also think of falsehood as a disjunction between zero alternatives. By analogy with the binary disjunction, we therefore have zero introduction rules, and an elimination rule in which we have to consider zero cases. This is precisely the $\bot E$ rule above.

From this is might seem that falsehood it useless: we can never prove it. This is correct, except that we might reason from contradictory hypotheses! We will see some examples when we discuss negation, since we may think of the proposition "*not A*" (written $\neg A$) as $A \supset \bot$. In other words, $\neg A$ is true precisely if the assumption A *true* is contradictory because we could derive $\bot$ *true*.

2.4 Notational Definition

The judgments, propositions, and inference rules we have defined so far collectively form a system of *natural deduction*. It is a minor variant of a system introduced by Gentzen [Gen35]. One of his main motivations was to devise rules that model mathematical reasoning as directly as possible, although clearly in much more detail than in a typical mathematical argument.

We now consider how to define negation. So far, the meaning of any logical connective has been defined by its introduction rules, from which we derived its elimination rules. The definitions for all the connectives are *orthogonal*: the rules for any of the connectives do not depend on any other connectives, only on basic judgmental concepts. Hence the meaning of a compound proposition depends only on the meaning of its constituent propositions. From the point of view of understanding logical connectives this is a critical property: to understand disjunction, for example, we only need to understand its introduction rules and not any other connectives.

A frequently proposed introduction rule for "*not A*" (written $\neg A$) is

$$\begin{array}{c} \overline{A\ \mathit{true}}\,u \\ \vdots \\ \dfrac{\bot\ \mathit{true}}{\neg A\ \mathit{true}}\,\neg I^u? \end{array}$$

In words: $\neg A$ is true if the assumption that A is true leads to a contradiction. However, this is not a satisfactory introduction rule, since the premise relies the meaning of $\bot$, violating orthogonality among the connectives. There are several approaches to removing this dependency. One is to introduce a new *judgment*, "*A is false*", and reason explicitly about truth and falsehood. Another employs schematic judgments, which we consider when we introduce universal and existential quantification.

Here we pursue a third alternative: for arbitrary propositions A, we think of $\neg A$ as a syntactic abbreviation for $A \supset \bot$. This is called a *notational definition* and we write

$$\neg A = A \supset \bot.$$

This notational definition is schematic in the proposition A. Implicit here is the formation rule

$$\dfrac{A\ \mathit{prop}}{\neg A\ \mathit{prop}}\,\neg F$$

We allow silent expansion of notational definitions. As an example, we prove that A and $\neg A$ cannot be true simultaneously.

$$\dfrac{\dfrac{\dfrac{\dfrac{}{A \wedge \neg A\ \mathit{true}}\,u}{\neg A\ \mathit{true}}\,{\wedge}E_R \qquad \dfrac{\dfrac{}{A \wedge \neg A\ \mathit{true}}\,u}{A\ \mathit{true}}\,{\wedge}E_L}{\bot\ \mathit{true}}\,{\supset}E}{\neg(A \wedge \neg A)\ \mathit{true}}\,{\supset}I^u$$

We can only understand this derivation if we keep in mind that $\neg A$ stands for $A \supset \bot$, and that $\neg(A \wedge \neg A)$ stands for $(A \wedge \neg A) \supset \bot$.

As a second example, we show the proof that $A \supset \neg\neg A$ is true.

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\neg\ A\ \mathit{true}}\,w \qquad \dfrac{}{A\ \mathit{true}}\,u}{\bot\ \mathit{true}}\,{\supset}E}{\neg\neg A\ \mathit{true}}\,{\supset}I^w}{A \supset \neg\neg A\ \mathit{true}}\,{\supset}I^u$$

Next we consider $A \vee \neg A$, the so-called *"law" of excluded middle*. It claims that every proposition is either true or false. This, however, contradicts our definition of disjunction: we may have evidence neither for the truth of A, nor for the falsehood of A. Therefore we cannot expect $A \vee \neg A$ to be true unless we have more information about A.

One has to be careful how to interpret this statement, however. There are many propositions A for which it is indeed the case that we know $A \vee \neg A$. For example, $\top \vee (\neg\top)$ is clearly true because $\top$ *true*. Similarly, $\bot \vee (\neg\bot)$ is true because $\neg\bot$ is true. To make this fully explicit:

$$\dfrac{\dfrac{}{\top\ \mathit{true}}\,\top I}{\top \vee (\neg\top)\ \mathit{true}}\,{\vee}I_L \qquad\qquad \dfrac{\dfrac{\dfrac{}{\bot\ \mathit{true}}\,u}{\neg\bot\ \mathit{true}}\,{\supset}I^u}{\bot \vee (\neg\bot)\ \mathit{true}}\,{\vee}I_R$$

In mathematics and computer science, many basic relations satisfy the law of excluded middle. For example, we will be able to show that for any two numbers k and n, either $k < n$ or $\neg(k < n)$. However, this requires proof, because for more complex A propositions we may not know if A *true* or $\neg A$ *true*. We will return to this issue later in this course.

At present we do not have the tools to show formally that $A \vee \neg A$ should not be true for arbitrary A. A proof attempt with our generic proof strategy (reason from the bottom up with introduction rules and from the top down with elimination rules) fails quickly, no matter which introduction rule for disjunction we start with.

$$\cfrac{\begin{array}{c}\vdots\\ A\ \mathit{true}\end{array}}{A \lor \neg A\ \mathit{true}}\,{\lor}I_L \qquad \cfrac{\cfrac{\begin{array}{c}\cfrac{}{A\ \mathit{true}}\,u\\ \vdots \\ \bot\ \mathit{true}\end{array}}{\neg A\ \mathit{true}}\,{\supset}I^u}{A\lor\neg A\ \mathit{true}}\,{\lor}I_R$$

We will see that this failure is in fact sufficient evidence to know that $A \lor \neg A$ is not true for arbitrary A.

2.5 Derived Rules of Inference

One popular device for shortening derivations is to introduce *derived rules of inference*. For example,

$$\frac{A \supset B\ \mathit{true} \qquad B \supset C\ \mathit{true}}{A \supset C\ \mathit{true}}$$

is a derived rule of inference. Its derivation is the following:

$$\cfrac{\cfrac{\cfrac{\cfrac{}{A\ \mathit{true}}\,u \qquad A \supset B\ \mathit{true}}{B\ \mathit{true}}\,{\supset}E \qquad B \supset C\ \mathit{true}}{C\ \mathit{true}}\,{\supset}E}{A \supset C\ \mathit{true}}\,{\supset}I^u$$

Note that this is simply a hypothetical derivation, using the premises of the derived rule as assumptions. In other words, a derived rule of inference is nothing but an evident hypothetical judgment; its justification is a hypothetical derivation.

We can freely use derived rules in proofs, since any occurrence of such a rule can be expanded by replacing it with its justification.

A second example of notational definition is logical equivalence "*A if and only if B*" (written $A \equiv B$). We define

$$(A \equiv B) = (A \supset B) \land (B \supset A).$$

That is, two propositions A and B are logically equivalent if A implies B and B implies A. Under this definition, the following become derived rules of inference (see Exercise 2.1). They can also be seen as introduction and elimination rules for logical equivalence (whence their names).

$$\dfrac{\begin{array}{c}\overline{A\ \mathit{true}}\,u\\ \vdots\\ B\ \mathit{true}\end{array}\qquad\begin{array}{c}\overline{B\ \mathit{true}}\,w\\ \vdots\\ A\ \mathit{true}\end{array}}{A\equiv B\ \mathit{true}}\equiv I^{u,w}$$

$$\dfrac{A\equiv B\ \mathit{true}\qquad A\ \mathit{true}}{B\ \mathit{true}}\equiv E_L \qquad\qquad \dfrac{A\equiv B\ \mathit{true}\qquad B\ \mathit{true}}{A\ \mathit{true}}\equiv E_R$$

2.6 Logical Equivalences

We now consider several classes of logical equivalences in order to develop some intuitions regarding the truth of propositions. Each equivalence has the form $A\equiv B$, but we consider only the basic connectives and constants ($\wedge$, $\supset$, $\vee$, $\top$, $\bot$) in A and B. Later on we consider negation as a special case. We use some standard conventions that allow us to omit some parentheses while writing propositions. We use the following operator precedences

$$\neg > \wedge > \vee > \supset > \equiv$$

where $\wedge$, $\vee$, and $\supset$ are right associative. For example

$$\neg A \supset A \vee \neg\neg A \supset \bot$$

stands for

$$(\neg A) \supset ((A \vee (\neg(\neg A))) \supset \bot)$$

In ordinary mathematical usage, $A\equiv B\equiv C$ stands for $(A\equiv B)\wedge(B\equiv C)$; in the formal language we do not allow iterated equivalences without explicit parentheses in order to avoid confusion with propositions such as $(A \equiv A) \equiv \top$.

Commutativity. Conjunction and disjunction are clearly commutative, while implication is not.

(C1) $A \wedge B \equiv B \wedge A\ \mathit{true}$

(C2) $A \vee B \equiv B \vee A\ \mathit{true}$

(C3) $A \supset B$ is not commutative

Idempotence. Conjunction and disjunction are idempotent, while self-implication reduces to truth.

(I1) $A \wedge A \equiv A\ \mathit{true}$

(I2) $A \vee A \equiv A\ \mathit{true}$

(I3) $A \supset A \equiv \top\ \mathit{true}$

Interaction Laws. These involve two interacting connectives. In principle, there are left and right interaction laws, but because conjunction and disjunction are commutative, some coincide and are not repeated here.

(L1) $A \wedge (B \wedge C) \equiv (A \wedge B) \wedge C$ *true*

(L2) $A \wedge \top \equiv A$ *true*

(L3) $A \wedge (B \supset C)$ do not interact

(L4) $A \wedge (B \vee C) \equiv (A \wedge B) \vee (A \wedge C)$ *true*

(L5) $A \wedge \bot \equiv \bot$ *true*

(L6) $A \vee (B \wedge C) \equiv (A \vee B) \wedge (A \vee C)$ *true*

(L7) $A \vee \top \equiv \top$ *true*

(L8) $A \vee (B \supset C)$ do not interact

(L9) $A \vee (B \vee C) \equiv (A \vee B) \vee C$ *true*

(L10) $A \vee \bot \equiv A$ *true*

(L11) $A \supset (B \wedge C) \equiv (A \supset B) \wedge (A \supset C)$ *true*

(L12) $A \supset \top \equiv \top$ *true*

(L13) $A \supset (B \supset C) \equiv (A \wedge B) \supset C$ *true*

(L14) $A \supset (B \vee C)$ do not interact

(L15) $A \supset \bot$ do not interact

(L16) $(A \wedge B) \supset C \equiv A \supset (B \supset C)$ *true*

(L17) $\top \supset C \equiv C$ *true*

(L18) $(A \supset B) \supset C$ do not interact

(L19) $(A \vee B) \supset C \equiv (A \supset C) \wedge (B \supset C)$ *true*

(L20) $\bot \supset C \equiv \top$ *true*

2.7 Summary of Judgments

Judgments.

A prop	A is a proposition
A true	Proposition A is true

Propositional Constants and Connectives. The following table summarizes the introduction and elimination rules for the propositional constants ($\top$, $\bot$) and connectives ($\wedge$, $\supset$, $\vee$). We omit the straightforward formation rules.

Introduction Rules Elimination Rules

$$\dfrac{A\;\mathit{true} \qquad B\;\mathit{true}}{A \wedge B\;\mathit{true}}\wedge I \qquad\qquad \dfrac{A \wedge B\;\mathit{true}}{A\;\mathit{true}}\wedge E_L \quad \dfrac{A \wedge B\;\mathit{true}}{B\;\mathit{true}}\wedge E_R$$

$$\dfrac{}{\top\;\mathit{true}}\top I \qquad\qquad \mathit{no}\ \top E\ \mathit{rule}$$

$$\dfrac{\begin{array}{c}\dfrac{}{A\;\mathit{true}}u\\ \vdots \\ B\;\mathit{true}\end{array}}{A\supset B\;\mathit{true}}\supset I^u \qquad\qquad \dfrac{A\supset B\;\mathit{true} \qquad A\;\mathit{true}}{B\;\mathit{true}}\supset E$$

$$\dfrac{A\;\mathit{true}}{A\vee B\;\mathit{true}}\vee I_L \quad \dfrac{B\;\mathit{true}}{A\vee B\;\mathit{true}}\vee I_R \qquad\qquad \dfrac{A\vee B\;\mathit{true} \quad \begin{array}{c}\dfrac{}{A\;\mathit{true}}u\\ \vdots \\ C\;\mathit{true}\end{array} \quad \begin{array}{c}\dfrac{}{B\;\mathit{true}}w\\ \vdots \\ C\;\mathit{true}\end{array}}{C\;\mathit{true}}\vee E^{u,w}$$

$$\mathit{no}\ \bot I\ \mathit{rule} \qquad\qquad \dfrac{\bot\;\mathit{true}}{C\;\mathit{true}}\bot E$$

Notational Definitions. We use the following notational definitions.

$\neg A$	$=$	$A \supset \bot$	not A
$A \equiv B$	$=$	$(A \supset B) \wedge (B \supset A)$	A if and only if B

2.8 A Linear Notation for Proofs

The two-dimensional format for rules of inference and deductions is almost universal in the literature on logic. Unfortunately, it is not well-suited for writing actual proofs of complex propositions, because deductions become very unwieldy. Instead with use a linearized format explained below. Furthermore, since logical symbols are not available on a keyboard, we use the following concrete syntax for propositions:

$A \equiv B$	`A <=> B`	A if and only if B
$A \supset B$	`A => B`	A implies B
$A \vee B$	`A	B`	A or B
$A \wedge B$	`A & B`	A and B
$\neg A$	`~ A`	not A

The operators are listed in order of increasing binding strength, and implication (`=>`), disjunction (`|`), and conjunction (`&`) associate to the right, just like the corresponding notation from earlier in this chapter.

The linear format is mostly straightforward. A proof is written as a sequence of judgments separated by semi-colon '`;`'. Later judgements must follow from earlier ones by simple applications of rules of inference. Since it can easily be verified that this is the case, explicit justifications of inferences are omitted. Since the only judgment we are interested in at the moment is the truth of a proposition, the judgment "*A true*" is abbreviated simply as "*A*".

The only additional notation we need is for hypothetical proofs. A hypothetical proof

$$\begin{array}{c} A \ \mathit{true} \\ \vdots \\ C \ \mathit{true} \end{array}$$

is written as `[A;...;C]`.

In other words, the hypothesis A is immediately preceded by a square bracket ('`[`'), followed by the lines representing the hypothetical proof of C, followed by a closing square bracket ('`]`'). So square brackets are used to delimit the scope of an assumption. If we need more than hypothesis, we nest this construct as we will see in the example below.

As an example, we consider the proof of $(A \supset B) \land (B \supset C) \supset (A \supset C)$ *true*. We show each stage in the proof during its natural construction, showing both the mathematical and concrete syntax, except that we omit the judgment "*true*" to keep the size of the derivation manageable. We write '`...`' to indicate that the following line has not yet been justified.

$$\begin{array}{c} \vdots \\ (A \supset B) \land (B \supset C) \supset (A \supset C) \end{array}$$

```
...
(A => B) & (B => C) => (A => C);
```

The first bottom-up step is an implication introduction. In the linear form, we use our notation for hypothetical judgments.

$$\dfrac{\begin{array}{c} \overline{(A \supset B) \land (B \supset C)}\,u \\ \vdots \\ A \supset C \end{array}}{(A \supset B) \land (B \supset C) \supset (A \supset C)} \supset I^u$$

```
[ (A => B) & (B => C);
  ...
  A => C ];
(A => B) & (B => C) => (A => C);
```

Again, we proceed via an implication introduction. In the mathematical notation, the hypotheses are shown next to each other. In the linear notation, the second hypothesis `A` is nested inside the first, also making both of them available to fill the remaining gap in the proof.

$$\dfrac{\dfrac{\begin{array}{c}\dfrac{}{(A\supset B)\wedge(B\supset C)}u \qquad \dfrac{}{A}w\\ \vdots \\ C\end{array}}{A\supset C}{\supset}I^w}{(A\supset B)\wedge(B\supset C)\supset(A\supset C)}{\supset}I^u$$

```
[ (A => B) & (B => C);
  [ A;
    ...
    C ];
  A => C ];
(A => B) & (B => C) => (A => C);
```

Now that the conclusion is atomic and cannot be decomposed further, we reason downwards from the hypotheses. In the linear format, we write the new line `A => B;` immediately below the hypothesis, but we could also have inserted it directly below `A;`. In general, the requirement is that the lines representing the premise of an inference rule must all come before the conclusion. Furthermore, lines cannot be used outside the hypothetical proof in which they appear, because their proof could depend on the hypothesis.

$$\dfrac{\dfrac{\begin{array}{c}\dfrac{\dfrac{}{(A\supset B)\wedge(B\supset C)}u}{A\supset B}{\wedge}E_L \qquad \dfrac{}{A}w\\ \vdots \\ C\end{array}}{A\supset C}{\supset}I^w}{(A\supset B)\wedge(B\supset C)\supset(A\supset C)}{\supset}I^u$$

```
[ (A => B) & (B => C);
  A => B;
  [ A;
    ...
    C ];
  A => C ];
(A => B) & (B => C) => (A => C);
```

Nex we apply another straightforward top-down reasoning step. In this case, there is no choice on where to insert `B;`.

$$\dfrac{\dfrac{\begin{array}{c}\dfrac{\dfrac{\dfrac{}{(A\supset B)\wedge(B\supset C)}u}{A\supset B}{\wedge}E_L \qquad \dfrac{}{A}w}{B}{\supset}E\\ \vdots \\ C\end{array}}{A\supset C}{\supset}I^w}{(A\supset B)\wedge(B\supset C)\supset(A\supset C)}{\supset}I^u$$

```
[ (A => B) & (B => C);
  A => B;
  [ A;
    B;
    ...
    C ];
  A => C ];
(A => B) & (B => C) => (A => C);
```

For the last two steps, we align the derivations vertically. The are both top-down steps (conjunction elimination followed by implication elimination).

$$\begin{array}{c}
\dfrac{\dfrac{}{(A \supset B) \wedge (B \supset C)}\,u}{B \supset C}\,\wedge E_R
\qquad
\dfrac{\dfrac{\dfrac{}{(A \supset B) \wedge (B \supset C)}\,u}{A \supset B}\,\wedge E_L \qquad \dfrac{}{A}\,w}{B}\,\supset E \\
\vdots \\
\dfrac{\dfrac{C}{A \supset C}\,\supset I^w}{(A \supset B) \wedge (B \supset C) \supset (A \supset C)}\,\supset I^u
\end{array}$$

```
[ (A => B) & (B => C);
  A => B;
  B => C;
  [ A;
    B;
    ...
    C ];
  A => C ];
(A => B) & (B => C) => (A => C);
```

In the step above we notice that subproofs may be shared in the linearized format, while in the tree format they appear more than once. In this case it is only the hypothesis $(A \supset B) \wedge (B \supset C)$ which is shared.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{(A \supset B) \wedge (B \supset C)}\,u}{B \supset C}\,\wedge E_R
\qquad
\dfrac{\dfrac{\dfrac{}{(A \supset B) \wedge (B \supset C)}\,u}{A \supset B}\,\wedge E_L \qquad \dfrac{}{A}\,w}{B}\,\supset E}{C}\,\supset E}{A \supset C}\,\supset I^w}{(A \supset B) \wedge (B \supset C) \supset (A \supset C)}\,\supset I^u$$

```
[ (A => B) & (B => C);
  A => B;
  B => C;
  [ A;
    B;
    C ];
  A => C ];
(A => B) & (B => C) => (A => C);
```

In the last step, the linear derivation only changed in that we noticed that `C` already follows from two other lines and is therefore justified.

For other details of concrete syntax and usage of the proof-checking program available for this course, please refer to the on-line documentation available through the course home page.

2.9 Normal Deductions

The strategy we have used so far in proof search is easily summarized: we reason with introduction rules from the bottom up and with elimination rules from the top down, hoping that the two will meet in the middle. This description is somewhat vague in that it is not obvious how to apply it to complex rules such as disjunction elimination which involve formulas other than the principal one whose connective is eliminated.

To make this precise we introduce two new judgments

$A \uparrow$	A has a normal proof
$A \downarrow$	A has a neutral proof

We are primarily interest in normal proofs, which are those that our strategy can find. Neutral proofs represent an auxiliary concept (sometimes called an *extraction proof*) necessary for the definition of normal proofs.

We will define these judgments via rules, trying to capture the following intuitions:

1. A normal proof is either neutral, or proceeds by applying introduction rules to other normal proofs.

2. A neutral proof proceeds by applying elimination rules to hypotheses or other neutral proofs.

By construction, every A which has a normal (or neutral) proof is true. The converse, namely that every true A has a normal proof also holds, but is not at all obvious. We may prove this property later on, at least for a fragment of the logic.

First, a general rule to express that every neutral proof is normal.

$$\frac{A \downarrow}{A \uparrow} \downarrow\uparrow$$

Conjunction. The rules for conjunction are easily annotated.

$$\frac{A \uparrow \qquad B \uparrow}{A \wedge B \uparrow} \wedge I \qquad \frac{A \wedge B \downarrow}{A \downarrow} \wedge E_L \qquad \frac{A \wedge B \downarrow}{B \downarrow} \wedge E_R$$

Truth. Truth only has an introduction rule and therefore no neutral proof constructor.

$$\frac{}{\top \uparrow} \top I$$

Implication. Implication first fixes the idea that hypotheses are neutral, so the introduction rule refers to both normal and neutral deductions.

$$\dfrac{\begin{array}{c}\overline{A\downarrow}\,u\\ \vdots\\ B\uparrow\end{array}}{A\supset B\uparrow}\supset I^u \qquad\qquad \dfrac{A\supset B\downarrow \qquad A\uparrow}{B\downarrow}\supset E$$

The elimination rule is more difficult to understand. The principal premise (with the connective "⊃" we are eliminating) should have a neutral proof. The resulting derivation will once again be neutral, but we can only require the second premise to have a normal proof.

Disjunction. For disjunction, the introduction rules are straightforward. The elimination rule requires again the requires the principal premise to have a neutral proof. An the assumptions introduced in both branches are also neutral. In the end we can conclude that we have a normal proof of the conclusion, if we can find a normal proof in each premise.

$$\dfrac{A\uparrow}{A\vee B\uparrow}\vee I_L \qquad \dfrac{B\uparrow}{A\vee B\uparrow}\vee I_R \qquad \dfrac{A\vee B\downarrow \quad \begin{array}{c}\overline{A\downarrow}\,u\\ \vdots\\ C\uparrow\end{array} \quad \begin{array}{c}\overline{B\downarrow}\,w\\ \vdots\\ C\uparrow\end{array}}{C\uparrow}\vee E^{u,w}$$

Falsehood. Falsehood is analogous to the rules for disjunction. But since there are no introduction rules, there are no cases to consider in the elimination rule.

$$\dfrac{\bot\downarrow}{C\uparrow}\bot E$$

All the proofs we have seen so far in these notes are normal: we can easily annotate them with arrows using only the rules above. The following is an example of a proof which is not normal.

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\neg A\ \mathit{true}}w \qquad \dfrac{\dfrac{\dfrac{}{A\ \mathit{true}}u \quad \dfrac{}{\neg A\ \mathit{true}}w}{A\ \wedge \neg A\ \mathit{true}}\wedge I}{A\ \mathit{true}}\wedge E_L}{\perp\ \mathit{true}}\supset E}{\neg A \supset \perp\ \mathit{true}}\supset I^w}{A \supset \neg A \supset \perp\ \mathit{true}}\supset I^u$$

If we follow the process of annotation, we fail at only one place as indicated below.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\neg A\downarrow}w \qquad \dfrac{\dfrac{\dfrac{\dfrac{}{A\downarrow}u \quad \dfrac{}{\neg A\downarrow}w}{A\ \wedge \neg A\ ?}\wedge I}{A\downarrow}\wedge E_L}{A\uparrow}\downarrow\uparrow}{\perp\downarrow}\supset E}{\perp\uparrow}\downarrow\uparrow}{\neg A \supset \perp\uparrow}\supset I^w}{A \supset \neg A \supset \perp\uparrow}\supset I^u$$

The situation that prevents this deduction from being normal is that we introduce a connective (in this case, $A \wedge \neg A$) and then immediately eliminate it. This seems like a detour—why do it at all? In fact, we can just replace this little inference with the hypothesis $A\downarrow$ and obtain a deduction which is now normal.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\neg A\downarrow}w \qquad \dfrac{\dfrac{}{A\downarrow}u}{A\uparrow}\downarrow\uparrow}{\perp\downarrow}\supset E}{\perp\uparrow}\downarrow\uparrow}{\neg A \supset \perp\uparrow}\supset I^w}{A \supset \neg A \supset \perp\uparrow}\supset I^u$$

It turns out that the only reason a deduction may not be normal is an introduction followed by an elimination, and that we can always simplify such a derivation to (eventually) obtain a normal one. This process of simplification is directly connected to computation in a programming language. We only need to fix a particular simplification strategy. Under this interpretation, a proof corresponds to a program, simplification of the kind above corresponds to computation, and a normal proof corresponds to a value. It is precisely this correspondence which is the central topic of the next chapter.

We close this chapter with our first easy meta-theorem, that is, a theorem *about* a logical system rather than within it. We show that if a the proposition A has a normal proof then it must be true. In order to verify this, we also need the auxiliary property that if A has a neutral proof, it is true.

Theorem 2.1 (Soundness of Normal Proofs) *For natural deduction with logical constants* $\wedge$, $\supset$, $\vee$, $\top$ *and* $\bot$ *we have:*

1. *If* $A\uparrow$ *then* A *true, and*

2. *if* $A\downarrow$ *then* A *true.*

Proof: We replace every judgment $B\uparrow$ and $B\downarrow$ in the deduction of $A\uparrow$ or $A\downarrow$ by B *true* and B *true*. This leads to correct derivation that A *true* with one exception: the rule

$$\frac{B\downarrow}{B\uparrow}\downarrow\uparrow$$

turns into

$$\frac{B\ \mathit{true}}{B\ \mathit{true}}$$

We can simply delete this "inference" since premise and conclusion are identical. □

2.10 Exercises

Exercise 2.1 *Show the derivations for the rules* $\equiv I$, $\equiv E_L$ *and* $\equiv E_R$ *under the definition of* $A\equiv B$ *as* $(A\supset B)\wedge(B\supset A)$.

Bibliography

[CGP99] E.M. Clarke, Orna Grumberg, and Doron Peled. *Model Checking.* MIT Press, Cambridge, Massachusetts, 1999.

[CR36] Alonzo Church and J.B. Rosser. Some properties of conversion. *Transactions of the American Mathematical Society*, 39(3):472–482, May 1936.

[Dav96] Rowan Davies. A temporal logic approach to binding-time analysis. In E. Clarke, editor, *Proceedings of the Eleventh Annual Symposium on Logic in Computer Science*, pages 184–195, New Brunswick, New Jersey, July 1996. IEEE Computer Society Press.

[Gen35] Gerhard Gentzen. Untersuchungen über das logische Schließen. *Mathematische Zeitschrift*, 39:176–210, 405–431, 1935. English translation in M. E. Szabo, editor, *The Collected Papers of Gerhard Gentzen*, pages 68–131, North-Holland, 1969.

[Har95] John Harrison. Binary decision diagrams as a HOL derived rule. *The Computer Journal*, 38:162–170, 1995.

[How80] W. A. Howard. The formulae-as-types notion of construction. In J. P. Seldin and J. R. Hindley, editors, *To H. B. Curry: Essays on Combinatory Logic, Lambda Calculus and Formalism*, pages 479–490. Academic Press, 1980. Hitherto unpublished note of 1969, rearranged, corrected, and annotated by Howard.

[HR00] Michael R.A. Huth and Mark D. Ryan. *Logic in Computer Science: Modelling and reasoning about systems.* Cambridge University Press, 2000.

[ML80] Per Martin-Löf. Constructive mathematics and computer programming. In *Logic, Methodology and Philosophy of Science VI*, pages 153–175. North-Holland, 1980.

[ML96] Per Martin-Löf. On the meanings of the logical constants and the justifications of the logical laws. *Nordic Journal of Philosophical Logic*, 1(1):11–60, 1996.

[Oka99] Chris Okasaki. Red-black trees in a functional setting. *Journal of Functional Programming*, 9(4):471–477, July 1999.

[XP98] Hongwei Xi and Frank Pfenning. Eliminating array bound checking through dependent types. In Keith D. Cooper, editor, *Proceedings of the Conference on Programming Language Design and Implementation (PLDI'98)*, pages 249–257, Montreal, Canada, June 1998. ACM Press.

[XP99] Hongwei Xi and Frank Pfenning. Dependent types in practical programming. In A. Aiken, editor, *Conference Record of the 26th Symposium on Principles of Programming Languages (POPL'99)*, pages 214–227. ACM Press, January 1999.