Lecture Notes on
Constructive Logic: Overview

15-317: Constructive Logic
Frank Pfenning

Lecture 1
August 29, 2017

1 Introduction

According to Wikipedia, logic is the study of the principles of valid inferences and demonstration. From the breadth of this definition it is immediately clear that logic constitutes an important area in the disciplines of philosophy and mathematics. Logical tools and methods also play an essential role in the design, specification, and verification of computer hardware and software. It is these applications of logic in computer science which will be the focus of this course. In order to gain a proper understanding of logic and its relevance to computer science, we will need to draw heavily on the much older logical traditions in philosophy and mathematics. We will discuss some of the relevant history of logic and pointers to further reading throughout these notes. In this introduction, we give only a brief overview of the goal, contents, and approach of this class.

2 Topics

The course is divided into four parts:

I. Proofs as Evidence for Truth

II. Proofs as Programs

III. Proof Search as Computation

IV. Substructural and Modal Logics

Proofs are central in all parts of the course, and give it its constructive nature. In each part, we will exhibit connections between proofs and forms of computations studied in computer science. These connections will take quite different forms, which shows the richness of logic as a foundational discipline at the nexus between philosophy, mathematics, and computer science.

In Part I we establish the basic vocabulary and systematically study propositions and proofs, mostly from a philosophical perspective. The treatment will be rather formal in order to permit an easy transition into computational applications. We will also discuss some properties of the logical systems we develop and strategies for proof search. We aim at a systematic account for the usual forms of logical expression, providing us with a flexible and thorough foundation for the remainder of the course. We will also highlight the differences between constructive and non-constructive reasoning. Exercises in this section will test basic understanding of logical connectives and how to reason with them.

In Part II we focus on constructive reasoning. This means we consider only proofs that describe algorithms. This turns out to be quite natural in the framework we have established in Part I. In fact, it may be somewhat surprising that many proofs in mathematics today are *not* constructive in this sense. Concretely, we find that for a certain fragment of logic, constructive proofs correspond to functional programs and vice versa. More generally, we can extract functional programs from constructive proofs of their specifications. We often refer to constructive reasoning as *intuitionistic*, while non-constructive reasoning is *classical*. Exercises in this part explore the connections between proofs and programs, and between theorem proving and programming.

In Part III we study a different connection between logic and programs where proofs are the result of computation rather than the starting point as in Part II. This gives rise to the paradigm of *logic programming* where the process of computation is one of systematic proof search. Depending on how we search for proofs, different kinds of algorithms can be described at a very high level of abstraction. Exercises in this part focus on exploiting logic programming to implement various algorithms in concrete languages such as Prolog.

In Part IV we study logics with more general and more refined notions of truth. For example, in temporal logic we are concerned with reasoning about truth relative to time. Another example is the modal logic S_5 where we reason about truth in a collection of worlds, each of which is connected to all other worlds. Proofs in this logic can be given an interpretation as dis-

tributed computation. Similarly, *linear logic* is a substructural logic where truth is ephemeral and may change in the process of deduction. As we will see, this naturally corresponds to imperative programming.

3 Goals

There are several related goals for this course. The first is simply that we would like students to gain a good working knowledge of constructive logic and its relation to computation. This includes the translation of informally specified problems to logical language, the ability to recognize correct proofs and construct them.

The second set of goals concerns the transfer of this knowledge to other kinds of reasoning. We will try to illuminate logic and the underlying philosophical and mathematical principles from various points of view. This is important, since there are many different kinds of logics for reasoning in different domains or about different phenomena[1], but there are relatively few underlying philosophical and mathematical principles. Our second goal is to teach these principles so that students can apply them in different domains where rigorous reasoning is required.

A third set of goals relates to specific, important applications of logic in the practice of computer science. Examples are the design of type systems for programming languages, specification languages, or verification tools for various classes of systems. While we do not aim at teaching the use of particular systems or languages, students should have the basic knowledge to quickly learn them, based on the materials presented in this class.

These learning goals present different challenges for students from different disciplines. Lectures, recitations, exercises, and the study of these notes are all necessary components for reaching them. These notes do not cover all aspects of the material discussed in lecture, but provide a point of reference for definitions, theorems, and motivating examples. Recitations are intended to answer students' questions and practice problem solving skills that are critical for the homework assignments. Exercises are a combination of written homework to be handed in at lecture and theorem proving or programming problems to be submitted electronically using the software written in support of the course. A brief tutorial and manual are available with the on-line course material.

[1] for example: classical, intuitionistic, modal, second-order, temporal, belief, linear, relevance, affirmation, ...

4 Intuitionism

We call a logic *constructive* if its proofs describe effective constructions. The emphasis here is on *effective* which is to say that the construction conveyed by a proof can actually be carried out mechanically. In other words, constructive proofs describe algorithms. At first one might think that all proofs describe constructions of this form, and this was historically true for a long time. At some point in the 19th century this direct link between mathematics and computation seemed to get lost. Some mathematicians objected to this and started to develop a foundations of mathematics in which all proofs denote effective constructions.

In order to understand this distinction better, we start with a theorem that illustrates the distinction, the so-called *Banach-Tarski Paradox*.[2]

Theorem 1 *Given a solid ball in 3-dimensional space, there exists a decomposition of the ball into a finite number of disjoint subsets, which can then be put back together in a different say to yield two identical copies of the original ball. Indeed, the reassembly process involves only moving the pieces around and rotating them, without changing their shape. The reconstruction can work with as few as five pieces.*

This is considered paradoxical, since we obviously cannot carry out such a decomposition. The intermediate pieces are in fact non-measurable infinite scatterings of points. The decomposition relies critically on the axiom of choice in set theory, which is highly non-constructive.

This is the kind of theorem (and proof, which we not show here but is sketched in the article) that mathematician L.E.J. Brouwer[3] might have objected to. It is meaningless with respect to our understanding of effective constructions, even if the formalities of its proof are sound. This entails a criticism of Hilbert's program, who posited that at the foundations of mathematics should be a formal system of axioms and inference rules with respect to which we can judge the correctness of mathematical arguments. Brouwer called himself an *intuitionist*, perhaps to contrast himself to Hilbert as a *formalist*.[4] Since intuitionistic logic has subsequently also been formalized (e.g., by Kolmogorov and Heyting), the modern way of framing the opposing sides are *intuitionistic logic* (or arithmetic) and *classical logic* (or arithmetic).

[2] See https://en.wikipedia.org/wiki/Banach-Tarski_paradox

[3] See https://en.wikipedia.org/wiki/L._E._J._Brouwer

[4] For more on this controversy in the foundations of mathematics, see https://en.wikipedia.org/wiki/Brouwer-Hilbert_controversy.

One of the key differences is the interpretation of the existential quantifier. In intuitionistic logic, proving $\exists x.\, A(x)$ entails exhibiting a witness t and a proof of $A(t)$. In classical logic, it is sufficient to show that $\forall x.\, \neg A(x)$ is impossible without exhibiting a witness in the proof.

As example, we consider the following theorem and proof.

Theorem 2 *There are two irrational numbers a and b such that a^b is rational.*

Proof: Consider $\sqrt{2}^{\sqrt{2}}$. There are two cases:

Case: $\sqrt{2}^{\sqrt{2}}$ is rational. Then $a = b = \sqrt{2}$ satisfies the claim.

Case: $\sqrt{2}^{\sqrt{2}}$ is irrational. Then $a = \sqrt{2}^{\sqrt{2}}$ and $b = \sqrt{2}$ satisfy the claim, since $a^b = (\sqrt{2}^{\sqrt{2}})^{\sqrt{2}} = \sqrt{2}^2 = 2$.

□

At this point, the classical mathematician is profoundly happy, since this is an extremely short and elegant proof of a prima facie nontrivial theorem. The intuitionist is profoundly unhappy, since it does not actually exhibit irrational witnesses a and b such that a^b is rational. They might $a = b = \sqrt{2}$, or they might be $a = \sqrt{2}^{\sqrt{2}}$ and $b = \sqrt{2}$. Therefore an intuitionist should reject this proof.

The step which turns out to be incorrect here is to assume that there are two cases (either $\sqrt{2}^{\sqrt{2}}$ is rational or not) without knowing which of the cases hold. More generally, an intuitionist rejects the *law of excluded middle* that any proposition is either true or false (in symbols: $A \vee \neg A$). Concretely, what counts as a constructive proof of $A \vee B$ is either a proof of A or a proof of B. So in addition to existential quantification, the intuitionistic and classical mathematician disagree on the interpretation of disjunction.

However, all is not lost! As an intuitionist, I look at the above proof and say

> *Oh, I understand your proof, but it is for a different theorem! What you have proven is:*
>
> > ***Theorem.*** *If $\sqrt{2}^{\sqrt{2}}$ is rational or not, then there are two irrational numbers a and b such that a^b is rational.*

Surprisingly, as long as we stick to pure logic, or perhaps the theory of natural numbers, any classical proof can be reinterpreted as an intuitionistic proof of a different theorem![5] This suggests that, once we accept that

[5] We may show this interpretation in a future lecture.

intuitionism can in fact also be formalized, intuitionistic and classical and no longer in conflict. Instead, intuitionistic logic is a *generalization* of classical logic in the sense that has a constructive existential quantifier and a constructive disjunction, which is absent from classical logic. At the same time, all classical theorems and proofs can be uniformly imported into intuitionistic logic under some translation.

The intuitionistic interpretation of the proof above yields another question: what is the nature of implication? The intuitionistic interpretation of this particular example clarifies this: the proof of $A \supset B$ consists of a *function* to convert a proof of A into a proof of B. Here, this function proceeds by analyzing the proof of whether $\sqrt{2}^{\sqrt{2}}$ is rational or not. If it is rational, we return the witnesses $a = b = \sqrt{2}$, together with the proof that a^b is rational in this case (which we were in fact given). If it is irrational, we return the witnesses $a = \sqrt{2}^{\sqrt{2}}$ and $b = \sqrt{2}$, together with a (simple equational) proof that $a^b = 2$ in this case.

Through this example, we have already identified three critial intuitionistic principles:

1. An intuitionistic proof of $\exists x.\, A(x)$ exhibits a witness t and a proof of $A(t)$.

2. An intuitionistic proof of $A \vee B$ consists of either a proof of A or a proof of B.

3. An intuitionistic proof of $A \supset B$ contains a construction that transforms a proof of A into a proof of B.

To achieve these, an intuitionist has to reject some classical reasoning principles or axioms. In natural deduction (as discussed in Lecture 2), this is manifest in the single *axiom of excluded middle*.

As a final example, consider the claim:

Theorem 3 *Among all the students in the class, there is a* leader *in the following sense: if he or she has a tattoo, then everyone in the class has a tattoo.*

In logical language, we could formalize this claim as

$$\exists x.\, (\mathsf{has}(x, \mathsf{tattoo}) \supset \forall y.\, \mathsf{has}(y, \mathsf{tattoo}))$$

where the quantifiers range of the students in this class. Here is the (non-constructive!) proof

Proof: Either everyone in the class has a tattoo, or there is at least one student s who does not have a tattoo.

Case: Everyone has a tattoo. Then any x will do[6], because the conclusion of the implication holds.

Case: There is some student s who does not have a tattoo. Then this student s is a leader: since $\mathsf{has}(\mathsf{s}, \mathsf{tattoo})$ is false, the implication $\mathsf{has}(\mathsf{s}, \mathsf{tattoo}) \supset \forall y.\, \mathsf{has}(y, \mathsf{tattoo})$ is true.

□

This is non-constructive, because we use a form of the excluded middle to avoid naming a witness to the existential (which we cannot do without violating students' privacy in unacceptable ways). Actually, as long as the domain of quantification is non-empty (usually assumed in classical logic), this proof has nothing to do with students and tattoos, but the proof above applies to the logical form

$$\exists x.\, (A(x) \supset \forall y.\, A(y))$$

Intuitionistically, we cannot prove this without further assumptions about A.[7]

In the next lecture we will start to look closely at the intuitionistic meaning of the logical connectives and their proof rules, based on the interpretation we sketched in this lecture.

[6] As pointed out by a student, this requires there to be at least one person in the class, which must be the case or that student couldn't have pointed it out.

[7] Exercise: which particular intuitionistically true proposition does the proof above establish?