BU CLA MA 531: Computability and Logic

Fall 1995

Handout 2

Assaf J. Kfoury

Tableaux Systems, Gentzen Systems, Natural-Deduction Systems

There are formal proof systems other than Hilbert-style for classical logic, whether propositional, first-order, or higher-order. Among these alternatives are *tableaux*, *Gentzen*, and *natural deduction* systems. None of these are discussed in Enderton's book.

In a Hilbert system, finding a derivation for a wff may be a tricky business, as one has to guess which axiom and which inference rule to use without systematic reliance on the syntax of the given wff. In tableaux, Gentzen, and natural deduction systems, there is less of this kind of guesswork, and derivations (or refutations) are largely syntax-directed. In particular, these alternative systems are more suitable for automatic theorem-proving[1] and allow for elegant presentations of intuitionistic logic[2]. One distinctive feature of these systems, not shared by Hilbert systems, is that they systematically and more efficiently produce a formal proof for a given wff — if it is indeed a valid wff, and if it is not, these systems will sometimes (but not always!) produce a counterexample to its validity. (Of course, we can also use a Hilbert system to produce a formal proof for a valid wff, typically by some exhaustive listing of all possible derivations, but this is not efficient at all!) Actually, in some variants of these alternative proof systems, it becomes very clear that a formal proof is basically a record of "an unsuccessful attempt to produce a counterexample". This is a simple idea, but it seems to be an insight of considerable importance in many applications.[3]

Not to give the wrong impression, there have been attempts to do automatic theorem-proving as well as presentations of intuitionistic logic based on Hilbert systems (of one form or another), but these are generally ill-suited for a computational task and often seem to obscure it by syntactic details. Of course, there are also subjective considerations (e.g. personal taste, familiarity, etc.) in choosing an appropriate proof system to work with. Moreover, although Hilbert systems are "inefficient and barbarously unintuitive",[4] they have advantages. First, the simplicity of their relatively few inference rules makes them suitable for encoding into arithmetic.[5] Second, it is relatively easy to tamper with the axiom schemes of Hilbert systems in order to adapt them to non-classical logics.[6]

---

[1] Automatic theorem-proving is a very active research area in computer science, where the "resolution" methods are essentially based on Gentzen systems or tableaux systems. For further reading on this, see [4] or [9] or [2].

[2] Intuitionistic logic is important in several foundational areas of computer science, notably in relation to typed λ-calculi and programming language theory.

[3] Further elaboration on this last point can be found in Chapter 14 of [9].

[4] Direct quote from [6], page 32.

[5] Without arithmetization of syntax we cannot reach Gödel's Incompleteness Theorem. See Ch. 3 in Enderton's.

[6] One such logic is intuitionism, for which we nevertheless prefer to use a proof system based on natural deduction.

Concerning tableaux systems, Smullyan gives a particularly lucid presentation in [10] (where tableaux are called "analytic tableaux") for classical logic, both propositional and first-order. Bell and Machover in [1], and then more recently Nerode and Shore in [8], also give presentations of classical logic and intuitionistic logic, both propositional and first-order, based on tableaux.

I will not say anything more about tableaux systems, in part because they are really duals of Gentzen systems: Whereas a Gentzen system systematically searches for a formal *proof* in *tree* form, a tableaux system systematically searches for a *refutation* in upside-down tree form. A node in a Gentzen proof tree corresponds to a branch in a tableau (more or less), and the Elimination Theorem for tableaux corresponds to Gentzen's Hauptsatz — for a definition of all these concepts, see the forementioned references (in relation to tableaux sysems) and the references in Handout 3 (in relation to Gentzen systems). One of the best expositions of the relationship between tableaux and Gentzen systems can be found in [10]. The view that tableaux are just a variant of Gentzen systems is discussed in some detail in [11].

Handout 3 and Handout 4 present a Gentzen system and a natural-deduction system, respectively, for classical logic.

Subsequent handouts (Handouts 5, 6, ... as many as time permits) will be devoted to intuitionistic logic. Formal proof systems for intuitionism can be of the Hilbert, or tableaux, or Gentzen, or natural-deduction variety. However, as already indicated, Hilbert systems are the least capable of revealing the computational aspects of the logic and its connections with typed λ-calculi.[7] By the "computational" aspects we mean this: When confronted with a sentence σ, instead of asking "when is σ true?" or "when is σ derivable?", we ask "what is a *proof* of σ?". While tableaux and Gentzen systems are better adapted to the task than Hilbert systems, the simplest way to consider this computational content of intuitionism is to use natural deduction. In particular, using natural deduction, it is easy to exhibit the so-called *Curry-Howard isomorphism* between intuitionistic logics and typed λ-calculi ("wff's and types are the same, proofs and λ-terms are the same"). More can be found in the early chapters of [5] on the beneficial effects of natural deduction.

The proof system in Handouts 5 and 6 (and later) will therefore be the natural-deduction system in Handout 4 appropriately restricted to intuitionistic logic, both propositional and first-order.

[7] A presentation of Hilbert systems for intuitionism can be found in Sections 9.8 and 9.9 of [1], or in Sections 19 and 23 of [7]. One advantage of the presentation in [7] is that it is extendable to a Hilbert system for classical logic by adding just one axiom scheme, namely $(\neg\neg\alpha \to \alpha)$ or $(\neg\alpha \vee \alpha)$, the law of "excluded middle" – see Handout 5.

References

[1] Bell, J., and Machover, M., *A Course in Mathematical Logic*, North-Holland, Amsterdam, 1977.

[2] Chang, C., and Lee, R.C., *Symbolic Logic and Mechanical Theorem Proving*, Academic Press, New York, 1973.

[3] van Dalen, D., *Logic and Structure*, Third Edition, Springer-Verlag, New York, 1994.

[4] Gallier, J.H., *Logic for Computer Science, Foundations of Automatic Theorem Proving*, Harper and Row, New York, 1986.

[5] Girard, J.-Y., Lafont, Y., and Taylor, P., *Proofs and Types*, Cambridge University Press, Cambridge, UK, 1989.

[6] Hodges, W., "Elementary Predicate Logic", in *Handbook of Philosophical Logic*, Vol. 1, ed. Gabbay and Guenthner, pp. 1-132, Reidel, 1983.

[7] Kleene, S.C., *Introduction to Metamathematics*, Van Nostrand, 1952.

[8] Nerode, A and Shore, R.A., *Logic for Applications*, Springer-Verlag, New York, 1993.

[9] Robinson, J.A., *Logic: Form and Function, The Mechanization of Deductive Reasoning*, North-Holland, New York, 1979.

[10] Smullyan, R.M., *First-Order Logic*, Springer-Verlag, New York, 1968.

[11] Sundholm, G., "Systems of Deductions", in *Handbook of Philosophical Logic*, Vol. 1, ed. Gabbay and Guenthner, pp. 133-188, Reidel, 1983.