int.1 Syntax of Intuitionistic Logic

The syntax of intuitionistic logic is the same as that for propositional logic. In classical propositional logic it is possible to define connectives by others, e.g., one can define \( \varphi \rightarrow \psi \) by \( \neg \varphi \lor \psi \), or \( \varphi \lor \psi \) by \( \neg(\neg \varphi \land \neg \psi) \). Thus, presentations of classical logic often introduce some connectives as abbreviations for these definitions. This is not so in intuitionistic logic, with two exceptions: \( \neg \varphi \) can be—and often is—defined as an abbreviation for \( \varphi \rightarrow \bot \). Then, of course, \( \bot \) must not itself be defined! Also, \( \varphi \leftrightarrow \psi \) can be defined, as in classical logic, as \( (\varphi \rightarrow \psi) \land (\psi \rightarrow \varphi) \).

Formulas of propositional intuitionistic logic are built up from propositional variables and the propositional constant \( \bot \) using logical connectives. We have:

1. A denumerable set \( \text{At}_0 \) of propositional variables \( p_0, p_1, \ldots \)
2. The propositional constant for falsity \( \bot \).
3. The logical connectives: \( \land \) (conjunction), \( \lor \) (disjunction), \( \rightarrow \) (conditional)
4. Punctuation marks: (, ), and the comma.

Definition int.1 (Formula). The set \( \text{Frm}(L_0) \) of formulas of propositional intuitionistic logic is defined inductively as follows:

1. \( \bot \) is an atomic formula.
2. Every propositional variable \( p_i \) is an atomic formula.
3. If \( \varphi \) and \( \psi \) are formulas, then \( (\varphi \land \psi) \) is a formula.
4. If \( \varphi \) and \( \psi \) are formulas, then \( (\varphi \lor \psi) \) is a formula.
5. If \( \varphi \) and \( \psi \) are formulas, then \( (\varphi \rightarrow \psi) \) is a formula.
6. Nothing else is a formula.

In addition to the primitive connectives introduced above, we also use the following defined symbols: \( \neg \) (negation) and \( \leftrightarrow \) (biconditional). Formulas constructed using the defined operators are to be understood as follows:

1. \( \neg \varphi \) abbreviates \( \varphi \rightarrow \bot \).
2. \( \varphi \leftrightarrow \psi \) abbreviates \( (\varphi \rightarrow \psi) \land (\psi \rightarrow \varphi) \).

Although \( \neg \) is officially treated as an abbreviation, we will sometimes give explicit rules and clauses in definitions for \( \neg \) as if it were primitive. This is mostly so we can state practice problems.
Photo Credits

Bibliography