Classical Logic is bivalent in that it admits exactly two truth-values: the true and the false. Many-valued logics, in contrast, allow for the consideration of arbitrarily large classes of truth-values. The canonical notion of entailment can be recovered by calling some of these values ‘designated’ and focusing on their preservation from premises to conclusion. A shade of bivalence clearly lurks, then, in the distinction between the values that are designated and those that are not. It is known (cf. [2]) that this residual bivalence allows in fact for an alternative, and in many cases even constructively obtained (cf. [1]), representation of many-valued logics in terms of appropriate bivalent semantics. In this paper we will present a first concrete implementation of the method devised in [1] in order to obtain sound and complete classic-like tableau systems for a very comprehensive class of finite-valued logics. The method is implemented in the functional programming language ML, and our program outputs a text file containing the corresponding theory to be processed by Isabelle, a flexible theorem-proving environment in which it is possible to check meta-results and theorems about the logics under scrutiny. The formulation of different many-valued logics under a common ground —two-signed tableau systems— makes it easier to compare properties of these logics and to appreciate the relations between them.
