The finite stages of inductive definitions *

Robert F. Stärk**

Department of Computer and Information Science, University of Pennsylvania, Philadelphia, PA 19104

Summary. In general, the least fixed point of a positive elementary inductive definition over the Herbrand universe is Π_1^1 and has no computational meaning. The finite stages, however, are computable, since validity of equality formulas in the Herbrand universe is decidable. We set up a formal system BID for the finite stages of positive elementary inductive definitions over the Herbrand universe and show that the provably total functions of the system are exactly that of Peano arithmetic. The formal system BID contains the so-called inductive extension of a logic program as a special case. This first-order theory can be used to prove termination and correctness properties of pure Prolog programs, since notions like negation-as-failure and left-termination can be turned into positive inductive definitions.

1. Why inductive definitions over the Herbrand universe?

In traditional logic programming, the semantics of a program is always given by the least fixed point of a monotonic operator over the Herbrand universe. The first example is the well-known van Emden-Kowalski operator for definite Horn clause programs in [25]. This operator is defined by a purely existential formula and is therefore continuous. The least fixed point of the operator is recursively enumerable. Moreover, the finite stages of the inductive definition are exactly what is computed by SLD-resolution.

In [11], Fitting has generalized the van Emden-Kowalski operator using three-valued logic to programs which may also contain negation in the bodies of the clauses. Although Fitting's operator is still monotonic it is no longer continuous. It follows from Blair [2] and Kunen [14] that the least fixed point of this operator can be Π_1^1-complete and that the closure ordinal can be ω_1^{CK} even for definite Horn clause programs.

The finite stages of Fitting's operator, however, are decidable and correspond to what is computed by SLDNF-resolution. This has been shown by Kunen in [15] for allowed logic programs and by the author in [21] for mode-correct programs. The class of allowed programs is considered as too restrictive in general. The class of mode-correct programs, however, contains most programs of practical interest, since a programmer has always modes in mind when he writes a program. Moreover, every allowed program is also mode-correct.

---

* This paper is in its final form and no similar paper has been or is being submitted elsewhere.

** Research supported by the Swiss National Science Foundation. This article has been written at the Department of Mathematics, Stanford University.