Naraschewski, Wolfgang; **Nipkow, Tobias**

Type inference verified: Algorithm $\mathcal{W}$ in Isabelle/HOL. (English) Zbl 0927.03025

Giménez, Eduardo (ed.) et al., Types for proofs and programs. International workshop TYPES '96, Aussois, France, December 15–19, 1996. Selected papers. Berlin: Springer. Lect. Notes Comput. Sci. 1512, 317-332 (1998).

Most functional programming languages have a common core: the simply typed λ-calculus enriched with let-expressions, i.e. local definitions of polymorphic values. This language is called Mini-ML. The set of well-typed Mini-ML expressions is inductively defined by a set of inference rules. One of the key properties of Mini-ML is that every well-typed expression has a most general type. The computation of the most general type is called type inference. It was first studied by *R. Hindley* [Trans. Am. Math. Soc. 146, 29-60 (1969; Zbl 0196.01501)] in the context of combinatory logic and later independently by *R. Milner* [J. Comput. Syst. Sci. 17, 348-375 (1978; Zbl 0388.68003)] for Mini-ML. Milner's type inference algorithm is known as algorithm $\mathcal{W}$. L. Damas proved the completeness of $\mathcal{W}$.

This paper presents the first machine-checked proof of correctness and completeness of $\mathcal{W}$. It is an extension of the work by *D. Nazareth* and *T. Nipkow* [Lect. Notes Comput. Sci. 1125, 331-346 (1996)], who treated the monomorphic case (no let-expressions). A partial verification of $\mathcal{W}$ in the proof checker Coq is reported by *C. Dubois* and *V. Ménissier-Morain* [in: J. von Wright et al. (eds.), Suppl. Proc. 9th Int. Conf. Theorem Proving in Higher-Order Logic, Turku Centre Comput. Sci., 15-30 (1996)]. Our paper provides the definition of (due to lack of space: almost) all concepts, the key lemmas, but almost no proofs. The complete development is accessible via http://www4.in.tum.de/~nipkow/.

For the entire collection see [Zbl 0898.00024].

MSC:

03B35 Mechanization of proofs and logical operations
03B40 Combinatory logic and lambda calculus
68T15 Theorem proving (deduction, resolution, etc.) (MSC2010)

Keywords:

simply typed lambda calculus; Mini-ML; computation of the most general type; type inference algorithm; machine-checked proof of correctness and completeness

Software:

Coq; Isabelle; Isabelle/HOL; mini-ML