

E. Börger G. Jäger H. Kleine Büning
M. M. Richter (Eds.)

Computer Science Logic

5th Workshop, CSL '91
Berne, Switzerland, October 7-11, 1991
Proceedings

Springer-Verlag

Berlin Heidelberg New York
London Paris Tokyo
Hong Kong Barcelona
Budapest

Table of Contents

<i>The expressive power of transitive closure and 2-way multihead automata</i> Y. Bargury and J. Makowsky	1
<i>Correctness proof for the WAM with types</i> Ch. Beierle and E. Börger	15
<i>Model checking of persistent Petri nets</i> E. Best and J. Esparza	35
<i>Provability in TBLL: A decision procedure</i> J. Chirimar and J. Lipton	53
<i>How to implement first order formulas in local memory machine models</i> E. Dahlhaus	68
<i>A new approach to abstract data types II. Computation on ADTs as ordinary computation</i> S. Feferman	79
<i>A primitive recursive set theory and AFA: On the logical complexity of the largest bisimulation</i> T. Fernando	96
<i>On bounded theories</i> J. Flum	111
<i>The cutting plane proof system with bounded degree of falsity</i> A. Goerdts	119
<i>Denotational versus declarative semantics for functional programming</i> J. C. González Moreno, M. T. Hortalá González and M. Rodríguez Artalejo	134
<i>On transitive closure logic</i> E. Grädel	149
<i>Some aspects of the probabilistic behavior of variants of resolution</i> P. Heusch and E. Speckenmeyer	164
<i>Safe queries in relational databases with functions</i> J. Hirshfeld	173
<i>Logical inference and polyhedral projection</i> J. N. Hooker	184
<i>Stable logic</i> B. Höslí	201
<i>A transformational methodology for proving termination of logic programs</i> M. R. K. Krishna Rao, D. Kapur and R. K. Shyamasundar	213
<i>Plausibility logic</i> D. Lehmann	227

<i>Towards Kleene algebra with recursion</i>	
H. Leiß	242
<i>Equational specification of abstract types and combinators</i>	
K. Meinke	257
<i>Normal forms in infinite-valued logic: The case of one variable</i>	
D. Mundici	272
<i>A fragment of first order logic adequate for observation equivalence</i>	
H. Oğuztüzün	278
<i>Ordinal processes in comparative concurrency semantics</i>	
S. Pinchinat	293
<i>Logical semantics of modularisation</i>	
G. R. Renardel de Lavalette	306
<i>A cut-elimination procedure designed for evaluating proofs as programs</i>	
U. R. Schmerl	316
<i>Minimal from classical proofs</i>	
H. Schwichtenberg	326
<i>Quantifier hierarchies over word relations</i>	
S. Seibert	329
<i>Complexity results for the default- and the autoepistemic logic</i>	
E. Steffen	339
<i>On completeness for NP via projection translations</i>	
I. A. Stewart	353
<i>Control of ω-automata, Church's problem, and the emptiness problem for tree ω-automata</i>	
J. G. Thistle and W. M. Wonham	367
<i>Comparing the theory of representations and constructive mathematics</i>	
A. S. Troelstra	382
<i>Infinitary queries and their asymptotic probabilities I: Properties definable in transitive closure logic</i>	
J. Tyszkiewicz	396
<i>On completeness of program synthesis systems</i>	
A. Voronkov	411
<i>Proving termination for term rewriting systems</i>	
A. Weiermann	419

Minimal from Classical Proofs

Helmut Schwichtenberg

Mathematisches Institut, Universität München
Theresienstr. 39, D-8000 München 2

We consider the $\rightarrow \forall$ -fragment of first order logic with a distinguished predicate symbol \perp (for falsity); as usual we write $\neg\varphi$ for $\varphi \rightarrow \perp$. Gentzen's natural deduction system for minimal logic in this language consists just of introduction and elimination rules for \rightarrow and \forall . Hence any proof in this system gives rise to a type-free λ -term, possibly with assumption variables. If in addition a proof (and hence also its associated λ -term) is normal, then from its context, i.e. the assignment of assumption formulas to its assumption variables, and from its endformula we can recover all formulas in the proof. This representation of formal proofs seems to be useful: for instance it allows an efficient implementation of normalization by evaluation (cf. [1], [2]).

It is well known that any proof can be transformed into a unique normal form with respect to β -conversion. Using η -expansion we can then construct the *long* normal form, where all minimal formulas are atomic.

We are interested in the problem of how to find proofs in minimal logic, from a somewhat practical point of view.* In particular we want to make use of existing theorem provers based on classical logic. So our problem is to review under what circumstances a classical proof can be converted into a proof in minimal logic, and moreover to describe reasonable algorithms which do this conversion. A good survey of the subject can be found in [3, Chapter 2.3]. Here we add a new result.

Note first that a convenient way to represent classical logic in our setting is to add stability assumptions of the form

$$\text{stab}_P : \forall \vec{x}. \neg\neg P\vec{x} \rightarrow P\vec{x}$$

for all predicate symbols P . For then we can easily derive $\neg\neg\varphi \rightarrow \varphi$ for an arbitrary formula φ , using

$$\begin{aligned} (\neg\neg\psi \rightarrow \psi) &\rightarrow \neg\neg(\varphi \rightarrow \psi) \rightarrow \varphi \rightarrow \psi \\ (\forall x. \neg\neg\varphi \rightarrow \varphi) &\rightarrow \neg\neg\forall x\varphi \rightarrow \forall x\varphi, \end{aligned}$$

which are derivable in our $\rightarrow \forall$ -fragment of minimal logic. Hence by a classical proof of ψ from assumptions $\varphi_1, \dots, \varphi_n$ we mean a proof in minimal logic using stability assumptions in addition to the given assumptions $\varphi_1, \dots, \varphi_n$.

A formula is called *Horn formula* if it has the form $\forall x_1, \dots, x_n. A_1 \rightarrow \dots \rightarrow A_m \rightarrow B$ with A_i and B atomic. It is called *definite Horn formula* if in addition we have $B \neq \perp$. If instead of atomic A_i we allow universally quantified atomic formulas, the result is called a *generalized (definite) Horn formula*.

* At the conference I gave a more general lecture on "Proofs and Programs". Since most of what I have said is already published (in [1] and [2]), this note only elaborates one part of the lecture dealing with a very special aspect of the field.

Theorem 1. *Let $\varphi_1, \dots, \varphi_n$ be generalized Horn formulas. We have a quadratic algorithm transforming a classical proof in long normal form of \perp from $\varphi_1, \dots, \varphi_n$ into a proof in minimal logic of \perp from the same assumptions.*

The proof is by induction on the total number of stability axioms used. Note first that bound assumption variables u in the given normal proof can only occur in the context

$$\text{stab}_{P\bar{r}}(\lambda ud)$$

with u of type $\neg P\bar{r}$ and d of type \perp . The reason for this is that all top formulas different from stability axioms are generalized Horn formulas which never have an implication in the premise of another implication.

Case 1. There is at least one occurrence of a bound assumption variable in the proof. Since we assume our proof to be in long normal form, any of the occurrences of an assumption variable u of type $\neg P\bar{r}$ must be the main premise of an \rightarrow -elimination, i.e. must be in a context ud_1 where u derives $P\bar{r}$. Now choose an uppermost occurrence of a bound assumption variable, i.e. a subderivation ud_1 where d_1 does not contain an occurrence of any bound assumption variable. Since d_1 derives $P\bar{r}$, we can replace the whole subderivation $\text{stab}_{P\bar{r}}(\lambda ud)$ of $P\bar{r}$ (the one where u is bound) by d_1 . Hence we have removed one occurrence of a stability axiom.

Case 2. Otherwise. If there are no more stability axioms in the proof, we are done. If not, choose an uppermost occurrence of a stability axiom, i.e. a subderivation $\text{stab}_{P\bar{r}}(\lambda ud)$ where d does not contain stability axioms. Since we are in case 2 here d also cannot contain free assumption variables which are bound elsewhere in the proof. But since d derives \perp , we can replace the whole proof (which also has \perp as its end formula) by d and hence we are done again.

Note that Theorem 1 is best possible in the sense that it becomes false if we allow an implication in the body of one of the Horn formulas. A counterexample (due to U. Berger) is

$$((P \rightarrow Q) \rightarrow \perp) \rightarrow (P \rightarrow \perp) \rightarrow \perp,$$

which is provable in classical but not in minimal logic. For if it were, we could replace \perp in this proof (which in minimal logic is just another propositional variable) by P , and hence we would obtain a proof in minimal logic of the Peirce formula

$$((P \rightarrow Q) \rightarrow P) \rightarrow P,$$

which is known to be undervivable.

By essentially the same argument we obtain the following variant of Theorem 1 for generalized *definite* Horn formulas:

Theorem 2. *Let $\varphi_1, \dots, \varphi_n$ be generalized definite Horn formulas. We have a quadratic algorithm transforming a classical proof in long normal form of an atomic formula B from $\varphi_1, \dots, \varphi_n$ into a proof in minimal logic of B from the same assumptions.*

The proof is by a simple modification of the argument for Theorem 1. Note that in case 2 it cannot happen that stability axioms occur in the proof since then we would have a derivation d of \perp from definite Horn formulas, which is clearly impossible.

References

1. U. Berger, H. Schwichtenberg: An inverse of the evaluation functional for typed lambda calculus. Proc. 6th IEEE Symp. on Logic in Computer Science, (ed.: R. Vemuri), Los Alamitos: IEEE Computer Society Press, pp. 203–211, 1991
2. H. Schwichtenberg: Proofs as Programs. To appear in: Leeds Proof Theory '90 (ed.: P. Aczel, H. Simmons, S.S. Wainer), Cambridge: University Press, 1992
3. A. Troelstra, D. van Dalen: Constructivism in Mathematics. Vol. I, Amsterdam: North-Holland, 1988