Automating Natural Deduction for Temporal Logic

Alexander Bolotov*
*University of Westminster
Harrow School of Computer Science
University of Westminster
Watford Road
Harrow HA1 3TP
A.Bolotov@wmin.ac.uk

Oleg Grigoriev†
†Moscow State University
Faculty of Philosophy
Department of Logic
Vorobievi Gory
Moscow 119899
grig@philos.msu.ru

Vasilyi Shangin‡
‡Moscow State University
Faculty of Philosophy
Department of Logic
Vorobievi Gory
Moscow 119899
b-shngn@mail.ru

Abstract

We present our recent work on the construction of natural deduction calculi for temporal logic. We analyse propositional linear-time temporal logic (PLTL) and Computation Tree Logic (CTL) and corresponding proof searching algorithms. The automation of the natural deduction calculi for these temporal logics opens the new prospect to apply our techniques as an automatic reasoning tool in the areas, where the linear-time or branching-time setting is required.

1 Introduction

This work continues our study of natural deduction (ND) proof systems for classical and non-classical logics. The particular approach to build an ND-calculus we are interested in is described in detail in [Bolotov et al. (2004)]. It is a modification of Quine's representation of subordinate proof [Quine (1950)] developed for classical propositional and first-order logic [Bolotov et al. (2004, 2005)] and later extended to a variety of non-classical logics [Bolotov et al. (2006a,b), Makarov (1998)]. All systems, except for the branching-time case, have sound, complete and terminating (except for the first order calculus) proof searching procedures. These results are obtained via adopting a generic proof searching method to a specific setting of the logic in question, thus allowing us to reflect the general nature of an ND as a kind of proof system.

2 Natural Deduction Calculi

Our presentation of the natural deduction proof systems for temporal logics follows the idea of labeled deductive systems [Gabbay (1998)]. We use PLTL and CTL formulae labeled by indices interpreted over the states of the underlying model and relational judgements, which are in turn labeled by indices interpreted over the branches of a tree model in case of CTL. Thus, in the branching-time setting, relational judgements represent an ordering of the states indices. The set of ND elimination and introduction rules for classical propositional logic is extended by the corresponding rules for temporal operators and basic CTL modalities, i.e. pairs $\mathbf{P}T$ where **P** is either of CTL path quantifiers and T is either of temporal operators. While working on the proof-searching algorithms, we found new interesting formulations of some rules. For example, in the case of CTL, a new formulation of the induction rule inspired by Goldblatt (1984), looks very powerful, and we believe can potentially replace several rules for the "until" operator in the original formulation of the system.

3 Proof Searching Algorithm

The proof searching algorithm creates two sequences of formulae, *list_proof*, a set of formulae that constitutes a proof, and *list_goals*, a sequence of formulae that constitutes a set of goals. The underlying searching algorithm is goal directed and represents two main techniques of reasoning. We first try to prove a goal G_n $\in$*list_goals* straightforwardly applying elimination rules to the formulae of *list_proof*. If this gives us the desired goal, then by the nature of the algorithms, we look at the previous goal, G_{n-1} whose structure determines which introduction rule must be applied to obtain the goal G_{n-1} from G_n. However, if this application of relevant searching procedures does not give us a solution, i.e. if aiming at reaching a goal G, we cannot derive in a proof a formula which is identical to G, then $\neg G$ is set up as a new assumption in *list_proof*, and we proceed commencing reasoning by refutation. We call these situations $false - blocks$. Being in such $false - block$, we apply one of the core novel techniques developed for temporal logic setting, allowing us to look how *list_goals* can be updated by considering formulae in *list_proof*.

4 Correctness

We have shown that in the case of PLTL, the proof searching technique guarantees that for any input formula F, the procedure terminates such that if F is valid then we are able to find a natural deduction proof of F, alternatively, we are able to extract a counter-model for F from *list_proof*.

The correctness of the proof searching technique for CTL has been our recent task.

Note that the determination of the application of introduction rules mentioned above is one of the distinguished characteristics of our method and plays significant role in our correctness argument. It prevents us from an "arbitrary" application of an introduction rule, such as introduction of disjunction – if not guided by a searching technique, this rule would allow us to introduce into *list_proof* a formula $A \vee B$ given either of disjuncts, A or B is a member of *list_proof*. Introduction of disjunction, which violates the famous subformula property, has been one of the main reasons for the criticism of natural deduction and for widely spread view within the automated reasoning community that natural systems are not suitable for automation.

5 Related work. Applications and Future Research.

We are not aware of any other proof search algorithm for temporal ND systems. For example, the only other ND constructions for linear-time logic [Indrzejczak (2004)] and branching-time logic [Renteria and Haeusler (2002)] which we are aware of have not been followed by any presentation of the relevant proof searching techniques. On the contrary, the generic nature of our ND method enables its efficient extension for linear and branching-time time logics.

As far as the areas of potential applications are concerned, we consider to utilise proposed natural deduction calculi as a reasoning tool for BDI and normative agents, where the temporal component is essential. This will require extensions of proposed calculi by the rules managing belief and deontic logic operators.

Finally, the complexity analysis will form another part of our future development of this research.

References

A. Bolotov, V. Bocharov, A. Gorchakov, V. Makarov, and V. Shangin. *Let Computer Prove It.* Logic and Computer. Nauka, Moscow, 2004. (In Russian).

A. Bolotov, V. Bocharov, A. Gorchakov, and V. Shangin. Automated first order natural deduction. In *Proceedings of IICAI*, pages 1292–1311, 2005.

A. Bolotov, A. Basukoski, O. Grigoriev, and V. Shangin. Natural deduction calculus for linear-time temporal logic. In *Proceedings of Jelia-2006, LNAI 4160*, 2006a.

A. Bolotov, O. Grigoriev, and V. Shangin. Natural deduction calculus for computation tree logic. In *IEEE John Vincent Atanasoff Symposium on Modern Computing*, pages 175–183, 2006b.

D. Gabbay. Labelled deductive systems, vol.1. *Journal of Logic, Language and Information.*, 7(4):502–506, 1998.

R. Goldblatt. *Logics of Time and Computation.* CSLI, second edition, 1984.

A. Indrzejczak. A labelled natural deduction system for linear temporal logic. *Studia Logica*, 75(3):345–376, 2004.

V. Makarov. Automatic theorem-proving in intuitionistic propositional logic. In *Modern Logic: Theory, History and Applications. Proceedings of the 5th Russian Conference*, StPetersburg, 1998. (In Russian).

W. Quine. On natural deduction. *Journal of Symbolic Logic*, 15:93–102, 1950.

C. Renteria and E. Haeusler. Natural deduction for CTL. *Bulletin of the Section of Logic, Polish Acad. of Sci.*, 31(4): 231–240, 2002.