First International Congress on Tools for Teaching Logic
The Interactive Learning Environment WinKE for Teaching Deductive Reasoning

Ulrich Endriss
endriss@dcs.kcl.ac.uk
King's College London
United Kingdom

June 2000

Abstract

WinKE is an interactive proof assistant based on the KE calculus, a refutation system which combines features from Smullyan's analytic tableaux and Gentzen's natural deduction. The software has been developed to support teaching logic and deductive reasoning at university level. In the sequel we briefly introduce the proof system KE and give an overview of the main features of the software tool.

1 The KE Calculus

The KE calculus, developed by Mondadori and D'Agostino [2], is similar to the well-known method of analytic tableaux in the sense that a theorem is shown by refuting its complement and the proof search is represented as a tree. Some of the rules, however, differ (and share some similarities with the natural deduction method). In particular, KE has an analytic cut rule, which is called PB (for 'principle of bivalence'). PB is the only branching rule of the system. Disjunctive formulas (usually referred to as 'beta' formulas) are analysed using a matching minor premise (an example would be modus ponens). Elsewhere [1] it has been argued that KE might be particularly suited for teaching logic. The first logic textbook based on KE has been published recently [5].

In [2] it has been shown that KE proofs are essentially shorter than tableaux proofs. In fact, PB corresponds to *semantic branching* and KE's beta rules correspond to what is sometimes called *boolean constraint propagation*, both of which are important optimisation techniques used in automated theorem proving (see for example [3, 4]).

Despite such advantages, constructing a proof tree with even a small number of branches can be very difficult to follow. Also, students cannot really be expected to solve anything but trivial exercises during tutorials or exams. This is not so much due to difficulties inherent in the proof system itself, but a simple consequence of the representational complexity of a proof – whatever method is used. Paper seems not to be a suitable medium for carrying out systematic logical deductions.

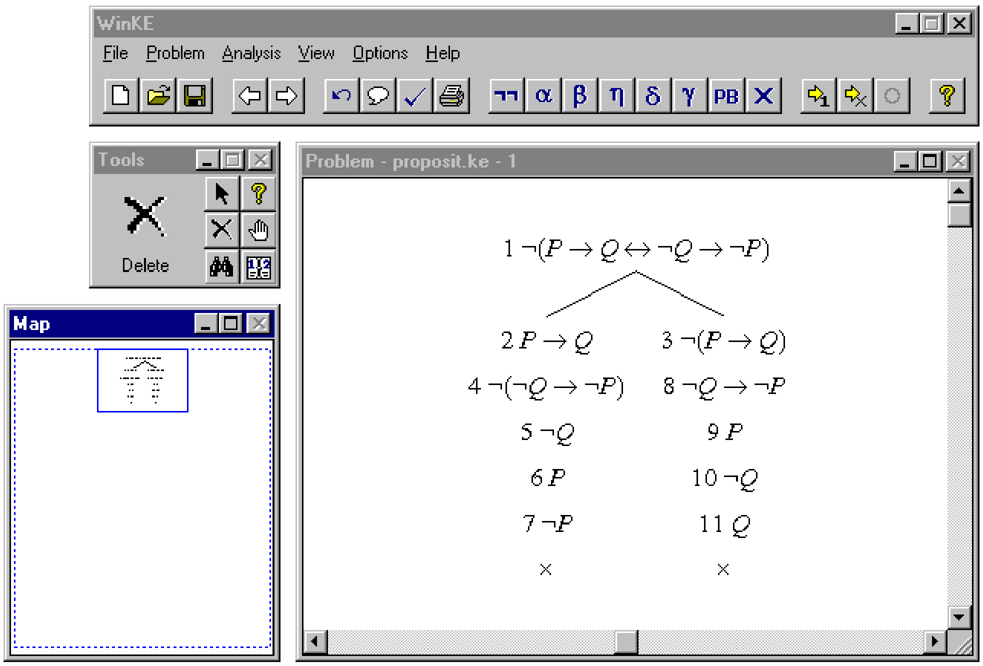

Figure 1: The WinKE interface

2 The WinKE Software

Some of the problems mentioned can be overcome by providing students with appropriate software tools that allow focusing on the important aspects of a problem. The intellectual challenge of finding a proof should still be left with the student, but the computer can assist in handling the sometimes large amounts of data involved. Here it is particularly important that the tool *facilitates* the acquisition of skills (like applying KE) rather than imposing additional difficulties on the student, who first has to learn how to use the program. The interactive learning environment WinKE has been developed to meet these requirements.

The construction of a proof in WinKE is a close simulation of what is commonly done using pen and paper and the system additionally offers a wide variety of useful features. It comprises three teaching modes, namely *supervisor*, *pedagogue*, and *assistant*, to be able to cater for the needs of different kinds of users. The interactive graphical user interface (shown in Fig. 1) consists of four windows: the main window providing menus and buttons for quick access to the most basic functions, the graphic window to display and manipulate KE proof trees, a viewer which displays

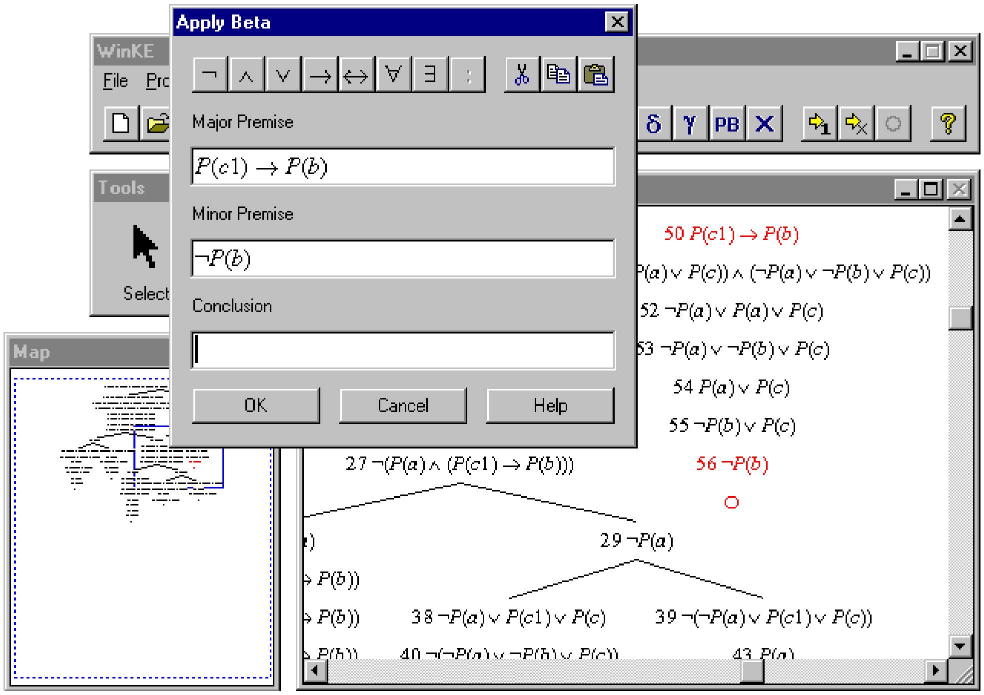

Figure 2: Applying a KE proof rule

a scaled-down view of the virtual drawing board, allowing to focus on a particular portion of it, and a tool box containing several graphic tools.

WinKE may simply serve as a comfortable drawing board for building up and manipulating proof trees, or the correctness of every single deduction step may be checked on-line. Typically, the tool is used to perform a step-by-step deduction. In Fig. 2, for example, the user is about to enter the conclusion of an application of a beta rule to $P(c1) \to P(b)$ and $\neg P(b)$. The program includes a fully automated theorem prover for classical propositional and first-order logic and also allows for the derivation of countermodels for saturated branches. Those countermodels can be visualised in various ways, e.g. as graphs or 'pigeon hole scenarios'. An example for the latter can be seen in Fig. 3.

User support includes bookkeeping facilities, giving hints, and a detailed on-line help system. The system provides several files with example problems, partly taken from [5]. New ones can be edited directly within WinKE. Teachers may configure the system to disable certain functions. You might, for example, not want to give beginners access to the automated deduction feature. The software is running under Windows and has been designed to be easy to learn and use.

Web site. Further information on WinKE and the underlying proof system KE can be found at the WinKE web site at

`http://www.dcs.kcl.ac.uk/staff/endriss/WinKE/`.

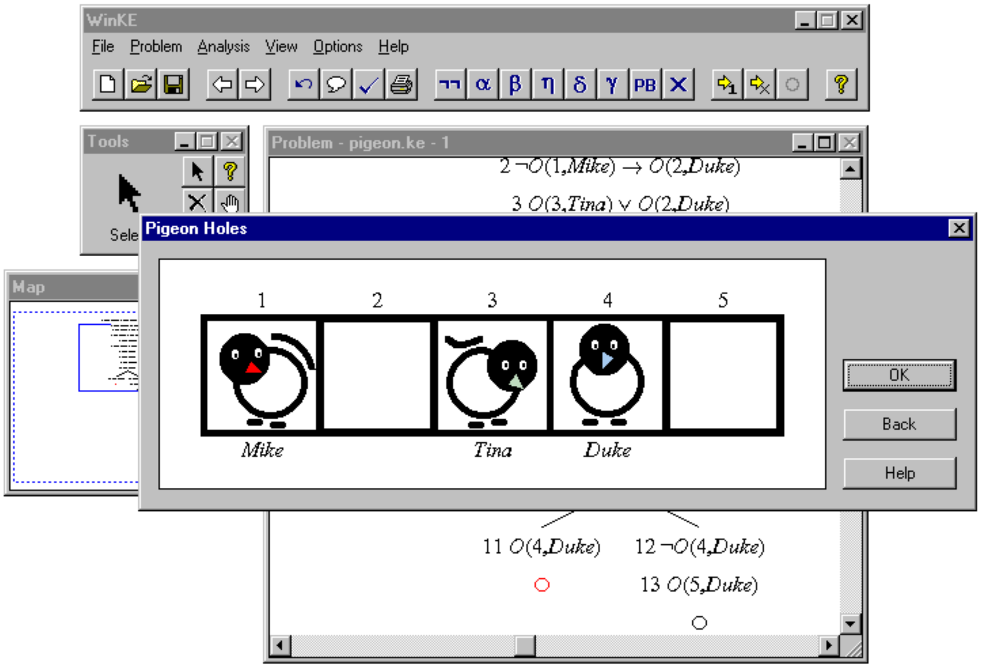

Figure 3: Visualisation of a countermodel for a 'pigeon hole scenario'

Obtaining the software. WinKE is available free of charge to teachers, students, and interested researchers. To obtain your copy, please send email to the author at `endriss@dcs.kcl.ac.uk`.

Acknowledgements. The development of WinKE has been funded by the Academic Centre for Research and Didactic Innovation (CARID) at the University of Ferrara, Italy.

References

[1] K. Broda, M. D'Agostino, and M. Mondadori. A solution to a problem of Popper. In *Karl Popper, Philosopher of Science. Proceedings of the conference The Epistemology of Karl Popper, Cesena, October 1994*. Kluwer Academic Publishers, Forthcoming.

[2] M. D'Agostino and M. Mondadori. The taming of the cut. Classical refutations with analytic cut. *Journal of Logic and Computation*, 4(3):285–319, 1994.

[3] J. W. Freeman. *Improvements to propositional satisfiability search algorithms.* PhD thesis, University of Pennsylvania, 1995.

[4] I. Horrocks and P. F. Patel-Schneider. Optimising description logic subsumption. *Journal of Logic and Computation*, 9(3):267–293, 1999.

[5] M. Mondadori and M. D'Agostino. *Logica*. Edizioni Scolastiche Bruno Mondadori, Milano, 1997.