

A book review.

LOVELAND, DONALD W., Automated Theorem Proving: A Logical Basis., Amsterdam etc., North-Holland Publishing Company, 406 pp., Dfl. 100,-- (Fundamental Studies in Computer Science).

I agreed to review this book in the hope that its study would be an instructive experience, and my highest expectations have been surpassed: the book contains a wealth of information (an average of two definitions per page being a modest estimate). The title covers the contents remarkably well.

The book has six chapters:

1. The role of logical systems. (51 pp.)
 2. Basic resolution. (41 pp.)
 3. Refinements of resolution. (106 pp.)
 4. Subsumption. (63 pp.)
 5. Resolution with equality. (60 pp.)
 6. Resolution and problem reduction format. (58 pp.)
- plus Appendix, References, Table of Symbols and (in view of the extensive terminology introduced, rather incomplete) Index.

Chapter 1 covers the standard material to be expected in an introduction to the topic; remarkable is the author's preference for his own "more semantic, or model-theoretic, approach" to Herbrand's Theorem over Herbrand's "purely syntactic argument", a preference that I don't expect to be shared universally.

Chapter 2 gives a very clear description of the Davis-Putnam Procedure for refutation, and a description of Robinson's Resolution Procedure, which I found somewhat less clear because its essential component "unification" remains very complicated. (In the one and only more ambitious example --(2.6.3)-- of resolution the annotation does not mention the substitutions, with the result that I couldn't read it without the help of pencil and paper.)

The next three chapters discuss at length and in logical depth a variety of strategies and inference rules that have been suggested for the improvement of the efficiency, because a naive mechanical application of Robertson's Resolution Principle leads to unacceptable amounts of manipulation. I said "in logical depth" because the author's main --and legitimate!-- concern is here completeness, i.e. the property that, in spite of the proposed constraints in the application of resolution, the algorithm remains in principle capable of detecting any unsatisfiable formula. The systematic attention paid to this concern is the distinctive feature of this book and justifies the second part of its title.

The last chapter discusses earlier attempts from the perspective gained from the extensive experience with the younger resolution-based attempts at automated theorem proving.

(For my inability to give in a few paragraphs a more adequate coverage of a 400-page book I offer my apologies to the author.)

* * *

One curious aspect of this book deserves to be mentioned: although

its subject matter is the (stylized) application of the first-order predicate calculus, 1) the author hardly uses it himself, and, 2) when he does, he does so clumsily.

Ad 1. All his proofs are entirely verbal as if the first-order predicate calculus did not exist: proofs of several pages running text are not unusual at all. This is the more amazing as his prose is often obscure or wrong. He refers to (p.14) "a distinguished set of elements", although on the next page it is clear that "a set of distinguished elements" was what he meant. He refers (p.118) to "a clause containing all false literals" meaning "a clause containing only false literals". He writes (p.106): "We will be concerned in the following sections with refinements that allow clauses to be ordered. This restricts [should be: "These restrict", EWD.] the possible resolvents that can be generated." How, a logician might ask, can one introduce restrictions by allowing something? He writes (p.15): "Assignment φ is similar mod x_1, \dots, x_n to γ if the assignments agree except for the elements assigned to x_1, \dots, x_n ." This sentence is obscure as we have to guess what "agreeing assignments" are. It is also ridiculous, because in a wide environment this is the only use of the Greek letters φ and γ , and there is never any sense in using an identifier only once. Etc.

Ad 2. He writes (p.365) "... we have shown $\sim G \Rightarrow G$, and, by using the tautology " $(\sim G \Rightarrow G) \Rightarrow G$ ", we have actually established G ." People using the first-order predicate calculus regularly can hardly fail to discover for themselves that the implication sign had better be avoided --it's introduction is responsible for the whole avoidable folklore of the "contrapositive" to which (p.432, etc.) the author devotes several pages-- and, when they encounter it, have learned to regard " $A \Rightarrow B$ " as a clumsy notation for " $\sim A \vee B$ ". Seeing that "we have actually established G " is then done at a glance, without invoking that tautology. By referring (p.343) to "the naturally [sic!] given implication" the author makes upon me the impression of not suspecting that the use of the implication is no more than a bad habit; this is the more amazing because the conceptual streamlining achieved by not using implications is directly reflected in the notational discipline that lies at the root of the resolution procedures that are the core subject matter of his whole book.

* * *

This book has been written with extensive experience and intimate knowledge of automated theorem proving, and thereby provides the reader an interesting look into the kitchen of one of the more respectable (or should we say: "less disreputable"?) branches of Artificial Intelligence. Also from this point of view the book is very interesting.

* * *

I found twelve erroneous usages of the word "alternate" and twelve printing errors, of which several in formulae. The publisher should be blamed for using a type font that does not distinguish between the digit 1 and the letter l. Apart from this blemish, the printing has been done admirably (in Hungary).

Plataanstraat 5
5671 AL NUENEN
The Netherlands

prof.dr.Edsger W.Dijkstra
Burroughs Research Fellow