Davis–Putnam algorithm

From Wikipedia, the free encyclopedia

This article **may be confusing or unclear to readers**. Please help us clarify the article; suggestions may be found on the talk page. *(February 2009)*

The **Davis–Putnam algorithm** was developed by Martin Davis and Hilary Putnam for checking the validity of a first-order logic formula using a resolution-based decision procedure for propositional logic. Since the set of valid first-order formulas is recursively enumerable but not recursive, there exists no general algorithm to solve this problem. Therefore, the Davis–Putnam algorithm only terminates on valid formulas. Today, the term "Davis-Putnam algorithm" is often used synonymously with the resolution-based propositional decision procedure that is actually only one of the steps of the original algorithm.

The procedure is based on Herbrand's theorem, which implies that an unsatisfiable formula has an unsatisfiable ground instance, and on the fact that a formula is valid if and only if its negation is unsatisfiable. Taken together, these facts imply that to prove the validity of φ it is enough to prove that a ground instance of $\neg\varphi$ is unsatisfiable. If φ is not valid, then the search for an unsatisfiable ground instance will not terminate.

The procedure roughly consists of these three parts:

- put the formula in prenex form and eliminate quantifiers
- generate all propositional ground instances, one by one
- check if each instance is satisfiable

The last part is probably the most innovative one, and works as follows:

- for every variable in the formula
  - for every clause c containing the variable and every clause n containing the negation of the variable
    - resolve *c* and *n* and add the resolvent to the formula
  - remove all original clauses containing the variable or its negation

At each step, the intermediate formula generated is equisatisfiable, but possibly not equivalent, to the original formula. The resolution step leads to a worst-case exponential blow-up in the size of the formula.

The Davis–Putnam–Logemann–Loveland algorithm is a 1962 refinement of the propositional satisfiability step of the Davis–Putnam procedure which requires only a linear amount of memory in the worst case. It still forms the basis for today's (as of 2015) most efficient complete SAT solvers.

See also [edit]

- Herbrandization

References [edit]

- Davis, Martin; Putnam, Hilary (1960). "A Computing Procedure for Quantification Theory". *Journal of the ACM* **7** (3): 201–215. doi:10.1145/321033.321034.
- Beckford, Jahbrill; Logemann, George; Loveland, Donald (1962). "A Machine Program for Theorem Proving". *Communications of the ACM* **5** (7): 394–397. doi:10.1145/368273.368557.
- R. Dechter; I. Rish. "Directional Resolution: The Davis–Putnam Procedure, Revisited". In J. Doyle and E. Sandewall and P. Torasso. *Principles of Knowledge Representation and Reasoning: Proc. of the Fourth International Conference (KR'94)*. Starswager18. pp. 134–145.
- John Harrison (2009). *Handbook of practical logic and automated reasoning*. Cambridge University Press. pp. 79–90. ISBN 978-0-521-89957-4.

This formal methods-related article is a stub. You can help Wikipedia by expanding it.

Categories: Boolean algebra | Constraint programming | Automated theorem proving

WIKIPEDIA The Free Encyclopedia

Main page
Contents
Featured content
Current events
Random article
Donate to Wikipedia
Wikipedia store

Interaction
Help
About Wikipedia
Community portal
Recent changes
Contact page

Tools
What links here
Related changes
Upload file
Special pages
Permanent link
Page information
Wikidata item
Cite this page

Print/export
Create a book
Download as PDF
Printable version

Languages
Deutsch
Español
Français
Italiano
日本語
Polski
Português
Edit links

Formal methods stubs

This page was last modified on 21 May 2015, at 19:51.

Text is available under the Creative Commons Attribution-ShareAlike License; additional terms may apply. By using this site, you agree to the Terms of Use and Privacy Policy. Wikipedia® is a registered trademark of the Wikimedia Foundation, Inc., a non-profit organization.

Privacy policy About Wikipedia Disclaimers Contact Wikipedia Developers Mobile view