AUTOMATIC THEOREM PROVING

by Woody Bledsoe, University of Texas at Austin

A talk given before the Advanced Technical Planning Committee of CAU-I, in Dallas, Texas, August 24, 1982.

There is an active group of ATP researchers at UT-Austin under the direction of Woody Bledsoe, Bob Boyer, J. Moore, and Frank Brown, which includes 6 professors and about 12 graduate students. Also, Don Good heads a large group there working on Program Verification (the Gypsy project). Other AI researchers at UT-Austin include Bob Simmons (text understanding), Gordon Novac (Automatic Physics programs), and Elaine Rich (Expert Systems).

Woody Bledsoe and Michael Ballantyne are studying the feasibility of establishing an AI Laboratory at the Woodlands, a new city north of Houston. This laboratory (WAIL), if established, would be funded by the Mitchell Energy Corporations and other corporations in the Houston area. This would be part of HARC (Houston Area Research Center) at the Woodlands, which has already been given 110 acres of land and several million dollars.

A good introduction to ATP can be found in <u>What Can Be Automated</u>?, ed. Bruce Artin, MIT Press 1980, pp. 448-462. This appears as (*) in the list of references given here.

ATP Reference 20 Jan. 1981

OKS

ang and Lee. Symbolic Logic and Mechanical Theorem Proving.
(see List of Reference in this book.)

nald Loveland. Automated Theorem Proving: A Logical Basis. Careful definitions and proofs, especially on Resolution.

ls Nilsson. Principles of Artificial Intelligence. Chaps. 4-6.

A. Robinson. Logic, Form and Function. (The inventor of the term RESOLUTION.) Especially Chapters 11-13.

bert Kowalski. Logic for Problem Solving. (The inventor of Logic Programming, SL-RESOLUTION, Resolution graphs, etc.).

yer and Moore. A Computational Logic. The "Boyer-Moore" system.

igenbaum and Feldman. Computers and Thought. Section 3. Early papers on ATP by: Newell, Simon, and Shaw, and Gelernter and Loveland.

. Siekmann and G. Wrightson. Collected Papers on Automatic Theorem Proving. Forthcoming from Springer. Three volumes. (Martin Davis' history of ATP will start the first volume; W. Bledsoe will write such a history for the third volume.)

INTRODUCTION

'Automatic Theorem Proving" (by W.W.Bledsoe), in What Can Be Automated? (NSF COSERS Study). Ed. Bruce Artin, MIT Press 1980, pp. 448-462.

'Non-resolution Theorem Proving," W. W. Bledsoe, A. I. Journal 9 (1977), 1-35.

OURNALS

International Journal of Artificial Intelligence. (AI Jour.)

Journal of the Association for Computing Machinery. (JACM)

Machine Intelligence (MI-1 - MI-9)

IEEE Transactions on Computers

CONFERENCE REPORTS

Proceedings of the Fourth Workshop on Automated Deduction, Austin, Texas, Feb. 1-3, 1979.

5th Conference on Automated Deduction, Les Arcs, France, July 8-11, 1980.

6th Conference on Automated Deduction, New York, June 7-9, 1982.

Lecture Notes in Compurer Science 87, 138, Springer-Verlag 1980, 1982

Proceedings of IJCAI 1969-1981.

Proceedings of AAAI National Conferences, AAAI, 445 Burgess Dr., Menlo Park, California, 94025.

What is ATP?

Proving theorems Automatically (by computer)

e.g., Pythagorean Theorem

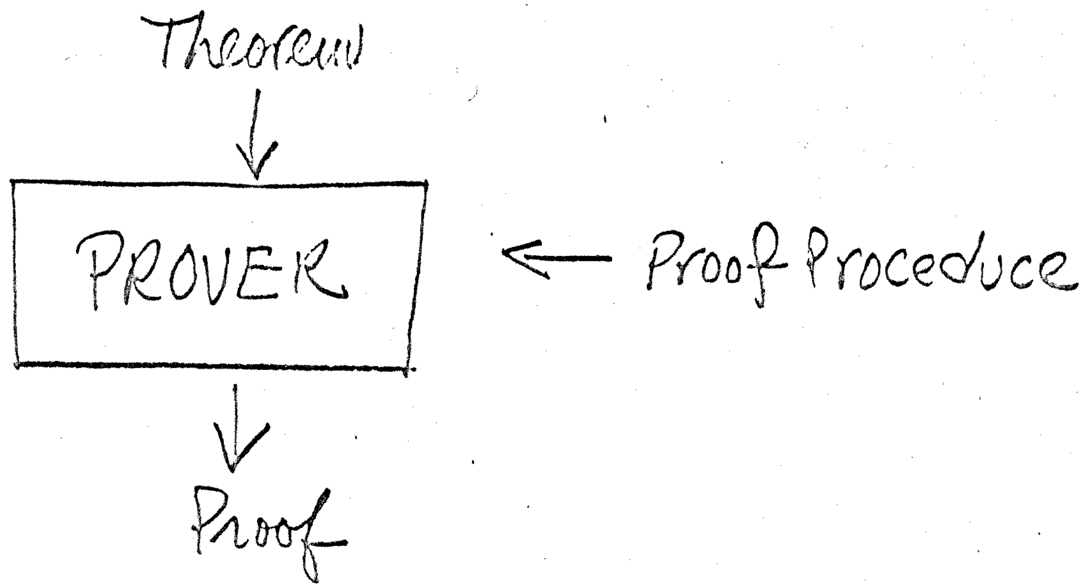
Heine-Borel Theorem

Schroeder Bernslein Theorem, etc.

new theorems

What has been done? Later

How? "



- SOUND : Does not "prove" non-theorems
- COMPLETE : Proves <u>all</u> theorems
  (semi-decision procedure)
  But may never finish on a non-theorem
- DECISION PROCEDURE :
  Can decide whether any formula
  is a theorem or not

Applications of ATP

Program Verification

esp. Man-machine theorem proving and proof checking

Now in use, somewhat (ISI, UT, Stanford, UT, --)

Program Synthesis (Waldinger-Manna, Balzer, ...)

Data Base Inference

Very important, but needs work

See Minker, et al. book (logic and data bases)

Truth maintenance?

Probabilistic inference

Logic Programming (Kowalski, et al.)

Expert Systems (MYCIN, PROLOGUE, PROSPECTOR, etc.)

Mathematics

Proof checking (see slide)

Man-machine (Assistant")

File of theorems

Any automatic decision maker

PROOF-CHECKING

J. Morris - "all" set theory theorems in A. P. Morse's book.

de Bruijn - all of Landau's book.

Boyer-Moore - prime factorization theorem, etc.

PV projects at UT, ISI, Stanford, SRI, ...

Suppes-Kreisel - CAI course in Set Theory to Gödel's Incomp. Th.

.
. Weyhrauch - FOL
.

Excellent application area, but has not been done right.

The user has to bend to the computer (let's change that).

Interesting, challenging, open problem.

Neveln - current APC project at UT.

These three pages are from Reference (*).

One of the earliest ATP programs was Galernter plane geometry prover. For example,

Theorem. Two vertices of a triangle are equidistant from the median to the side determined by those vertices.

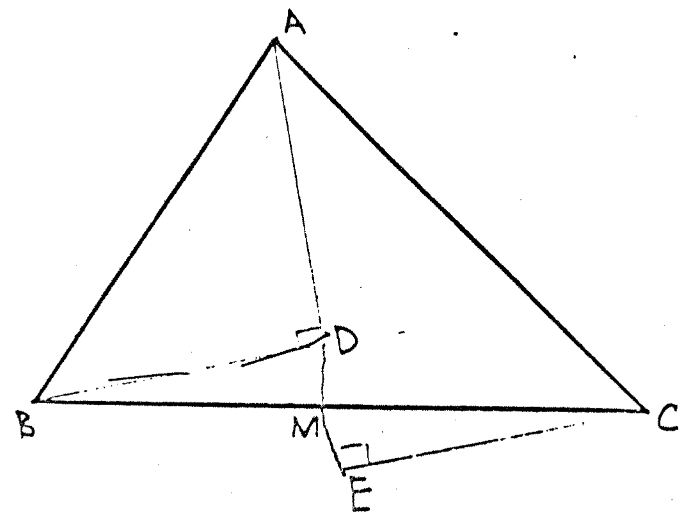


Figure 1

GIVEN: Segment BM = Segment MC, BD⊥AM, CE⊥ME.

GOAL: Segemtn BD = Segment EC.

SOLUTION:

Angle DMB = Angle EMC	Verticle Angles
Angle BDM = Angle CEM	Right Angles are equal
Segment BM = Segment MC	Given
CEM is a triangle	Assumption based on diagram
BDM is a triangle	Assumption based on diagram
Δ CEM $\cong$ Δ BDM	Side-angle-angle
Segment BD = Segment EC	Corresponding elements of congruent triangles

This is the machine's proof though we have omitted some of its steps for simplicity of presentation.

In this proof the machine proceeds ("reasoning backwards") as follows:

Its goal is

G1 Segment BD = Segment EC .

So it consults a list of solutions for this type of goal and finds (among others), that two segments can be proved equal by showing that they are corresponding parts of congruent triangles. Since BD is in Δ BDM and EC is in Δ CEM, it selects the subgoal

G2 Δ CEM ≈ Δ BDM .

Now it consults another list for ways of proving two triangles congruent. It finds: (a) three-sides, (b) side-angle-side, (c) side-angle-angle. It sets the subgoal

G3 "three-sides" for Δ's CEM and BDM .

This fails (after a good deal of work). So it sets the subgoal

G5 "side-angle-angle" .

There are several ways this can be achieved, one of which requires the three sub-goals

G6 Segment BM = Segment MC

G7 Angle DMB = Angle EMC

G8 Angle BDM = Angle CEM .

The machine finds subgoal G6 among its premises. In solving subgoals G7 and G8 it consults a list of methods for making two angles equal, and finds (among others): "verticle angles are equal", and "all right angles are equal." Since it detects from the diagram that angles DMB and EMC are verticle angles, and that angles BDM and CEM are right angles it successfully concludes the proof of subgoals G6 - G8, and therefore G5, G2 and G1. In the second step of the proof when subgoal G2 was selected, the machine could have selected any of the following subgoals:

G2.1	Δ CEA $\approx$ Δ BDA
G2.2	Δ CEA $\approx$ Δ BDM
G2.3	Δ CEM $\approx$ Δ BDA
G2	Δ CEM $\approx$ Δ BDM .

But, by constructing in its memory a "general" diagram of the situation (which is its representation of the drawing in Figure 2), the machine easily checked by measurements that subgoals G2.1 - G2.3 could not be true, but that G2 seems alright. Thus it selected only subgoal G2 and thereby drastically reduced the search time.

This idea of filtering out false subgoals is generalized and used in many areas of automatic theorem proving. For example, in group theory a false subgoal can be discarded by testing it on known groups (such as the Klein four groups).

Mathematics can be divided as follows:

<table>
<tr><td colspan="2">FIRST ORDER LOGIC</td><td rowspan="2">HIGHER ORDER LOGIC</td></tr>
<tr><td>PROPOSITIONAN LOGIC</td><td></td></tr>
</table>

↑ Decision Procedure

Complete (first order logic) — incomplete (higher order logic)

- PROPOSITIONAN LOGIC – Rather trivial

  eg $[P \wedge (P \rightarrow Q) \wedge (\sim Q \vee R) \longrightarrow R]$

  No quantifiers: $(\forall, \exists)$

  Easily handled by Computers. (Tr Tab)

- FIRST ORDER LOGIC – difficult
- HIGHER " " – more "

FIRST ORDER LOGIC

- Examples See next two pages.
- Quantification of individual variables
- This is the challenge of this age, to prove all theorems in first-order logic.
- A complete proof procedure was devised by Herbrand in 1930

Herbrand Procedure

So we are finished?

No. It was too slow!

- Can methods be devised so that computer provers can compete with humans? surpass them?

That's the challenge.

Where are we now?

Theorem. $a \leq l \leq b \wedge a \leq t_0 \leq l$

$\wedge \forall \varepsilon (\varepsilon > 0 \rightarrow \exists r (r < l \wedge \forall s (r \leq s \leq l \rightarrow f(l) \leq f(s) + \varepsilon)))$

$\wedge \forall y (a \leq y < l \rightarrow \exists z (y < z \leq l \wedge \forall t (a \leq t < z \rightarrow f(z) \leq f(t))))$

$\longrightarrow f(l) \leq f(t_0)$

Theorem. The sum of two continuous functions continuous:

$\forall \varepsilon (\varepsilon > 0 \rightarrow \exists \delta (\delta > 0 \wedge \forall y (|x_0 - y| < \delta \rightarrow |f(x_0) - f(y)| \leq \varepsilon)))$

$\wedge \forall \varepsilon (\cdot \quad - \qquad |g(x_0) - g(y)| \leq \varepsilon))$

$\wedge \ \varepsilon_0 > 0 \longrightarrow$

$\exists \delta (\delta > 0 \wedge \forall x (|x_0 - x| < \delta \longrightarrow |[f(x) + g(x)] - [f(x_0) + g(x_0)]| < \varepsilon_0).$

Example: AM5

C1. $\forall x \in [a,b]\ \forall \varepsilon > 0\ \exists r < x\ \forall s (r \le s \le x \rightarrow f(x) \le f(s) + \varepsilon)$

C2. $\quad$ " $\quad$ " $\quad \exists r > x\ \forall s (x \le s \le r \rightarrow$ " $)$

B1. $\forall x \in [a,b]\ (\forall t (a \le t \le x \rightarrow f(t) \le f(x)) \rightarrow x \le l)$

B2. $\forall y (y < l \rightarrow \exists z \in [a,b]\ [\forall t (a \le t \le z \rightarrow f(t) \le f(z)) \wedge y < z \le l]$

-B1. $\forall w \in [a,b]\ \exists g\ \forall x \in [a,b]\ (f(x) \le f(w) \rightarrow g \le x)$

-B2. $\forall w \in [a,b]\ \exists g\ \forall y (g < y \rightarrow \exists z \in [a,b] (f(z) \le f(w) \wedge g \le z < y))$

<u>eorem</u> (AM5) $\quad a \le b \wedge C1 \wedge C2 \wedge LUB1 \wedge LUB2 \wedge GLB1 \wedge GLB2$

$$\longrightarrow \exists u \in [a,b]\ \forall t \in [a,b]\ (f(u) \le f(t)).$$

CHRONOLOGY

1930	"Herbrand Procedure"	
		(also Skolem Presburger, etc.)
/ / / / / /	COMPUTERS / / / / / / / / / /	
1955	Logic Theorist	NSS Rand (Principia Math.)
1959	Geometry Machine	Gelernter
1960	Herbrand Procedure Won't Work. Too Slow.	Gilmore
1960-65	Improved Hilbert Procedure	Davis, Putnam, Prowitz, Russians
"	Wang's System	
1965	RESOLUTION	J. A. Robinson efficient, excitement
1965-70	Refinements of RESOLUTION	
1970	"Natural Deduction Systems"	Bledsoe, Nevins, C. Hewitt, Loveland, etc.
1970's	Both types Applications	

- Newell-Simon-Shaw (Chap. 2 of (1) 38 of 52 theorems)

  Wang

  (all of (1); > 350 Theorems)

  (1) Principia Mathematica (Whitehead & Russell)

  (2) Propositional Logic

- Gelernter

  A number of theorems in plane geometry

  (not requiring constructions)

Some Theorems Proved Automatically

$x^3 = x$ (in a Ring) ⟶ The RING is Commutative

- Unique factorization Theorem (with some input Lemmas)
- The sum of two continuous functions is continuous (δ, ε)
- Intermediate Value Theorem

* f continuous on [a, b] ⟶ f u.cont. on [a, b]
* using non-standard analysis
(+ similar Theorems: Bolzano Weierstrass, etc)

SOME TO PROVE

- Schreoder-Bernstein Theorem

* f cont on [a, b] ⟶ f u.cont on [a, b]
* without non-stand Anal

- Heine Borel Theorem
- Hahn-Banach Theorem

Two main Types of Provers (to be discussed shortly).

- RESOLUTION
- NATURAL DEDUCTION

<u>Two</u> Types of ACTIVITY (of Research)

- Devising Proof procedures and testing them, using Computers
- Proving Completeness results (mathematical)

OR

- Man-machine
- Machine alone

Much of the research in ATP during the last fifteen years has been stimulated by J. A. Robinson's introduction of RESOLUTION in 1965 (see the books by Chang and Lee, Loveland, or Robinson). A succinct easy-to-read, introduction in RESOLUTION is given in Reference (*).

Another kind of ATP research utilizes the "Natural Deduction" Method (see reference (**)).

Natural Deduction is governed by a set of (production) rules. They use the implication symbol "$\rightarrow$". For example,

John is a boy $\rightarrow$ John is a male,

or more generally

$$P \rightarrow Q$$

where P and Q are statements which are either true or false.

Some Natural Deduction Rules

(for the Ground Case - no variables)

$$\frac{P\ ,\ P \rightarrow Q}{Q}$$

$$\frac{P \rightarrow Q,\ P \rightarrow R}{P \rightarrow Q \,\&\, R}$$ AND-SPLIT

$$\frac{R \rightarrow S,\ P \rightarrow Q}{P \,\&\, (Q \rightarrow R) \rightarrow S}$$ BACK CHAIN

$$\frac{P \rightarrow R,\ Q \rightarrow R}{P \text{ or } Q \rightarrow R}$$ CASES

$$\frac{}{P \rightarrow P}$$ match

John is a boy → John is male

P → Q