

Tema DA–1: Introducción a la deducción automática

José A. Alonso Jiménez
Miguel A. Gutiérrez Naranjo

Dpto. de Ciencias de la Computación e Inteligencia Artificial
UNIVERSIDAD DE SEVILLA

¿Qué es el razonamiento automático?

- L. Wos: *What is Automated Reasoning?* (Journal of Automated Reasoning, Vol. 1 (1985) pp. 6–9)
 - Automated reasoning is concerned with the study of using the computer to assist in that part of problem solving requiring reasoning. Some questions arising during the study concern the representation of knowledge, the rules for deriving new knowledge from that which is given, and strategies for controlling the rules. Other questions concern the implementation of the resulting theory and concern various applications for which the corresponding software can be used. Theory, implementation, and application play equally vital and interconnecting roles for automated reasoning in its attempt to reach one of its primary goals – the goal of providing an automated reasoning assistant.

¿Qué es el razonamiento automático?

- Campos de aplicación del R.A.
 - Verificación de sistemas computacionales
 - Programación lógica
 - Programación lógica inductiva
 - Síntesis de programas
 - Inteligencia artificial
 - Representación del conocimiento
 - Planificación
 - Robótica
 - Razonamiento de los agentes inteligentes
 - Procesamiento del lenguaje natural
 - Demostración y descubrimiento de teoremas matemáticos

Historia de la deducción automática

- 1666: Leibniz: *Ars Combinatoria*
- 1847: Boole: *The Mathematical Analysis of Logic*
- 1869: Jevons: máquina lógica
- 1879: Frege: *Begriffsschrift*
- 1892: Peano: *Formulario Mathematico*
- 1910: Whitehead y Russell: *Principia Mathematica*
- 1928: Hilbert y Ackermann: *Grundzüge der theoretischen Logik*
- 1931: Gödel: incompletitud
- 1936: Tarski: concepto de verdad

Historia de la deducción automática

- 1936: Church: indecidibilidad
- 1936: Turing: máquina universal
- 1954: M. Davis: Demostrador aritmético
- 1956: Darmouth: Nacimiento de la Inteligencia Artificial.
- 1956: Newell, Shaw y Simon: *Logic Theorist*
- 1959: Gelernter: máquina geométrica
- 1960: Wang: demostrador basado en el cálculo Gentzen
- 1960: Gilmore: demostrador Skolem
- 1960: Davis y Putnam: procedimiento DP

Historia de la deducción automática

- 1964: Robinson: regla de resolución y unificación
- 1964: Wos: resolución unidad
- 1964: Wos y otros: estrategia del soporte
- 1965: Robinson: regla de hiper-resolución
- 1965: Robinson: subsunción
- 1967: Wos y otros: demodulación
- 1968: Loveland: resolución lineal
- 1969: Guard y otros: SAM (lema)

Historia de la deducción automática

- 1969: Green: QA3, obtención de respuestas
- 1970: Chang: resolución por entradas
- 1970: Wos y Robinson: paramodulación
- 1970: Knuth y Bendix: completación de sistemas de reescritura
- 1970: Bruijin: Automath
- 1971: Fikes y Nilsson: STRIPS: planificación
- 1972: ANL AURA (Automated reasoning assistant)
- 1972: Kowalski: programación lógica
- 1973: Colmerauer: Prolog

Historia de la deducción automática

- 1973: Boyer y Moore: Edinburgh Pure Lisp Theorem Prover
- 1976: Wos y otros: resolución UR
- 1976: Wos y otros: estrategia de pesos
- 1978: Boyer y Moore: A Computational Logic
- 1978: Boyer y Moore: Nqthm
- 1982: ANL: ITP (Interactive Theorem Prover)
- 1983: Rusinoff: teorema de Wilson en
- 1984: AMS *ATP (after 25 years)*
- 1986: Shankar: teorema de incompletitud de Gödel

Historia de la deducción automática

- 1988: McCune: OTTER
- 1989: Kaufmann y Moore: ACL2
- 1991: Shankar y otros: PVS
- 1992: Quaife: *Automted development of fundamental mathematical theories*
- 1992: Kunen–OTTER: axiomas simples de grupos
- 1992: Boyer y Moore: Nqthm–1992
- 1994: McCune: MACE
- 1996: McCune–EQP: Problema de las álgebras de Robbins

Ejemplos de problema

- Rompecabeza de los misioneros y los caníbales

Tres misioneros y tres caníbales están en la orilla derecha de un río. Hay una barca que puede transportar sólo dos personas. Si el número de misioneros nunca puede ser inferior al de caníbales en ninguna de las orillas, ¿cómo pueden cruzar todos a la orilla izquierda?. (La barca no puede cruzar vacía).

- Diseño de circuitos:

Usando cualquier número de puertas AND y OR, pero no más de dos puertas NOT, construir un circuito de acuerdo con la siguiente especificación: Hay tres entradas (e_1 , e_2 y e_3) y tres salidas (s_1 , s_2 y s_3) de forma que se cumplan las siguientes relaciones:

$$s_1 = \text{not}(e_1) \qquad s_2 = \text{not}(e_2) \qquad s_3 = \text{not}(e_3)$$

Ejemplos de problema

- Problema elemental de grupos:

Sea G un grupo y e su elemento neutro. Demostrar que si, para todo x de G , $x^2 = e$, entonces G es conmutativo.

- Formalización

- * Axiomas de grupo:

$$(\forall x)[e.x = x]$$

$$(\forall x)[x.e = x]$$

$$(\forall x)[x^{-1}.x = e]$$

$$(\forall x)[x.x^{-1} = e]$$

$$(\forall x)(\forall y)(\forall z)[(x.y).z = x.(y.z)]$$

- * Hipótesis

$$(\forall x)[x.x = e]$$

- * Conclusión

$$(\forall x)(\forall y)[x.y = y.x]$$

Ejemplos de problema

- Entrada: grupos.in

```
op(400, xfy, *).  
op(300, yf, ^).
```

```
list(usable).  
e * x = x.           % Ax. 1  
x * e = x.           % Ax. 2  
x^ * x = e.          % Ax. 3  
x * x^ = e.          % Ax. 4  
(x * y) * z = x * (y * z). % Ax. 5  
x = x.               % Ax. 6  
x * x = e.  
b * a != a * b.  
end_of_list.
```

```
set(auto2).
```

Ejemplos de problema

- Ejecución

```
otter <grupos.in >grupos.out
```

- Prueba

```
----> UNIT CONFLICT at 0.01 sec ----> 38 [binary,37.1,1.1] $F.
```

```
----- PROOF -----  
1 [] b*a!=a*b.  
3,2 [] e*x=x.  
5,4 [] x*e=x.  
10 [] (x*y)*z=x*(y*z).  
13 [] x*x=e.  
17 [para_into,10.1.1.1,13.1.1,demod,3,flip.1] x*(x*y)=y.  
23 [para_into,10.1.1,13.1.1,flip.1] x*(y*(x*y))=e.  
33 [para_from,23.1.1,17.1.1.2,demod,5,flip.1] x*(y*x)=y.  
37 [para_from,33.1.1,17.1.1.2] x*y=y*x.  
38 [binary,37.1,1.1] $F.  
----- end of proof -----
```

Problema de Robbins

- **Axiomas de Huntington (1933):**
 - (A) $(x + y) + z = x + (y + z)$
 - (C) $x + y = y + x$
 - (H) $n(n(x) + y) + n(n(x) + n(y)) = x$
- **Axioma de Robbins (1933):**
 - (R) $n(n(n(y) + x) + n(x + y)) = x$
- **Teorema:** (A)+(C)+(H) \implies (R)
- **Problema de Robbins:** (A)+(C)+(R) \implies (H)
- **Lemas (Winkler, 1990):**
 - (A)+(C)+(R)+(∃c)(∃d)[c + d = c] \implies (H)
 - (A)+(C)+(R)+(∃c)(∃d)[n(c + d) = n(c)] \implies (H)
- **Teorema (McCune, 1996):**
 - (A)+(C)+(R) \implies (∃c)(∃d)[c + d = c]
- **Entrada a EQP**
 $n(n(n(y) + x) + n(x + y)) = x$.
 $x + y \quad ! = \quad x$.
 $n(x + y) \quad ! = \quad n(x)$.

Problema de Robbins

2	[]	$-(n(x+y)=n(x))$.
3	[]	$n(n(n(x)+y)+n(x+y))=y$.
5	[3,3]	$n(n(n(x+y)+n(x)+y)+y)=n(x+y)$.
6	[3,3]	$n(n(n(n(x)+y)+x+y)+y)=n(n(x)+y)$.
24	[6,3]	$n(n(n(n(x)+y)+x+2y)+n(n(x)+y))=y$.
47	[24,3]	$n(n(n(n(n(x)+y)+x+2y)+n(n(x)+y)+z)+n(y+z))=z$.
48	[24,3]	$n(n(n(n(x)+y)+n(n(x)+y)+x+2y)+y)=n(n(x)+y)$.
146	[48,3]	$n(n(n(n(x)+y)+n(n(x)+y)+x+3y)+n(n(x)+y))=y$.
250	[47,3]	$n(n(n(n(n(x)+y)+x+2y)+n(n(x)+y)+n(y+z)+z)+z)=n(y+z)$.
996	[250,3]	$n(n(n(n(n(n(x)+y)+x+2y)+n(n(x)+y)+n(y+z)+z)+z+u)+n(n(y+z)+u))=u$.
16379	[5,996,3]	$n(n(n(n(x)+x)+3x)+x)=n(n(x)+x)$.
16387	[16379,3]	$n(n(n(n(n(x)+x)+3x)+x+y)+n(n(n(x)+x)+y))=y$.
16388	[16379,3]	$n(n(n(n(x)+x)+4x)+n(n(x)+x))=x$.
16393	[16388,3]	$n(n(n(n(x)+x)+n(n(x)+x)+4x)+x)=n(n(x)+x)$.
16426	[16393,3]	$n(n(n(n(n(x)+x)+n(n(x)+x)+4x)+x+y)+n(n(n(x)+x)+y))=y$.
17547	[146,16387]	$n(n(n(n(n(x)+x)+n(n(x)+x)+4x)+n(n(n(x)+x)+3x)+x)+x)$ $=n(n(n(x)+x)+n(n(x)+x)+4x)$.
17666	[24,16426,17547]	$n(n(n(x)+x)+n(n(x)+x)+4x)=n(n(n(x)+x)+3x)$.

$$n(c+d) = n(c), \quad c = n(n(x)+x)+3x, \quad d = n(n(x)+x)+x$$

Bibliografía

- M. Davis. *The Early History of Automated Deduction* En *Handbook of Automated Reasoning*
(<http://www.cs.nyu.edu/cs/faculty/davism/early.ps>)
- G. Kolata *Computer Math Proof Shows Reasoning Power* (The New York Times, 10 de Diciembre de 1996)
(<http://www.nytimes.com/library/cyber/week/1210math.html>)
- D. MacKenzie *The Automation of Proof: A Historical and Sociological Exploration* (<http://dream.dai.ed.ac.uk/papers/donald>)