Machine-Oriented Reasoning

Cláudia Nalon

http://www.cic.unb.br/∼nalon

nalon@unb.br

Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

CADE-27, Natal, 2019

▷ Logic
Logic
Calculus
Formally
Metatheorical Properties
Notes
The unavoidable slide
Semantics
The Early Days
DPLL
Resolution

Logic

▷ Logic
▷ Logic
Calculus
Formally
Metatheorical Properties
Notes
The unavoidable slide
Semantics
The Early Days
DPLL
Resolution

- Logic is a formal language together with a relationship, often denoted by "$\models$", between a set of formulae and a formula:

$$\Gamma \models \varphi$$

- Formulae related by this relationship to the empty set are of special interest. If

$$\emptyset \models \varphi \text{ or } \models \varphi$$

  we say φ is valid.
- For classical logic, the deduction theorem ensures that if

$$\{\gamma_1, \ldots, \gamma_n\} \models \varphi$$

  then

$$\models (\gamma_1 \rightarrow \ldots (\gamma_{n-1} \rightarrow (\gamma_n \rightarrow \varphi)))$$

Calculus

▷ Logic
Logic
▷ Calculus
Formally
Metatheorical Properties
Notes
The unavoidable slide
Semantics
The Early Days
DPLL
Resolution

- A logic language together with a syntactically defined relation, often denoted by "$\vdash$", is called a calculus.

$$\Gamma \vdash \varphi$$

- Again, formulae related by this relationship to the empty set are of special interest. If

$$\emptyset \vdash \varphi \text{ or } \vdash \varphi$$

we say φ is a theorem.

▷ Logic
Logic
Calculus
▷ Formally
Metatheorical Properties
Notes
The unavoidable slide
Semantics
The Early Days
DPLL
Resolution

- Formally, a calculus is defined as a pair $\langle \mathcal{A}, \mathcal{R} \rangle$, where $\mathcal{A}$ is a set of formulae and $\mathcal{R}$ is a set of inference rules.
- An inference rule is usually given as:

$$\frac{\begin{array}{c}\gamma_1\\ \vdots \\ \gamma_n\end{array}}{\varphi}$$

where γ_i are the premises and φ is the conclusion. The application of the inference rule to the premises *produces* the conclusion. We also say that the conclusion is *derived* from the premises (by the inference rule).

Metatheorical Properties

▷ Logic
Logic
Calculus
Formally
▷ Metatheorical Properties
Notes
The unavoidable slide
Semantics
The Early Days
DPLL
Resolution

- Strong soundness:

$$\text{if } \Gamma \vdash^{C}_{\mathcal{L}} \varphi \text{ then } \Gamma \models_{\mathcal{L}} \varphi$$

- Weak soundness:

$$\text{if } \vdash^{C}_{\mathcal{L}} \varphi \text{ then } \models_{\mathcal{L}} \varphi$$

- Strong completeness:

$$\text{if } \Gamma \models_{\mathcal{L}} \varphi \text{ then } \Gamma \vdash^{C}_{\mathcal{L}} \varphi$$

- Weak completeness:

$$\text{if } \models_{\mathcal{L}} \varphi \text{ then } \vdash^{C}_{\mathcal{L}} \varphi$$

- Consistency: if there is no φ such that both $\vdash^{C}_{\mathcal{L}} \varphi$ and $\vdash^{C}_{\mathcal{L}} \neg\varphi$.

▷ Logic
Logic
Calculus
Formally
Metatheorical Properties
▷ Notes
The unavoidable slide
Semantics
The Early Days
DPLL
Resolution

- Because of the deduction theorem, weak properties are often enough to show a calculus has good properties. This is why automated reasoning is often referred to as *theorem proving*.
- Those good properties are not enough to ensure that a calculus is suitable for implementation.

▷ Logic
Logic
Calculus
Formally
Metatheorical Properties
Notes
▷ The unavoidable slide
Semantics
The Early Days
DPLL
Resolution

The set of logical symbols is given by the union of the following sets:

1. $\mathcal{P} = \{P^{n_P}, Q^{n_Q}, R^{n_R}, \ldots, P_1^{n_{P_1}}, Q_1^{n_{Q_1}}, R_1^{n_{R_1}}, \ldots\}$;
2. $\mathcal{F} = \{f^{n_f}, g^{n_g}, h^{n_h}, \ldots, f_1^{n_{f_1}}, g_1^{n_{g_1}}, h_1^{n_{h_1}}, \ldots\}$;
3. $\mathcal{C} = \{a, b, c, \ldots, a_1, b_1, c_1, \ldots\}$;
4. $\mathcal{V} = \{x, y, z, \ldots, x_1, y_1, z_1, \ldots\}$;
5. $\{\forall, \exists\}$
6. $\{\neg, \vee, \wedge, \rightarrow, \leftrightarrow\}$;
7. (,), and ,.

The set of well-formed formulae in the first-order language, denoted by $\mathsf{WFF}_{\mathcal{L}_{FO}}$, is the least set obtained by recursively applying the following:

- □ $P^n(t_1, \ldots, t_n) \in \mathsf{WFF}_{\mathcal{L}_{FO}}$, where $P^n \in \mathcal{P}$, for $t_i \in \mathcal{T}$, $0 \leq i \leq n$, $n \in \mathbb{N}$;
- □ if $\varphi, \psi \in \mathsf{WFF}_{\mathcal{L}_{FO}}$ and $x \in \mathcal{V}$, then $\neg\varphi$, $(\varphi \vee \psi)$, $(\varphi \wedge \psi)$, $(\varphi \rightarrow \psi)$, $(\varphi \leftrightarrow \psi)$, $\forall x\varphi$ and $\exists x\varphi \in \mathsf{WFF}_{\mathcal{L}_{FO}}$.

Obs.: We'll restrict ourselves to sentences, that is, well-formed formulae with no free variables.

▷ Logic
Logic
Calculus
Formally
Metatheorical Properties
Notes
The unavoidable slide
▷ Semantics
The Early Days
DPLL
Resolution

An *interpretation* $\mathbb{M}$ for $(\mathcal{P}, \mathcal{F})$ consists of:

- □ a non-empty set $\mathcal{A}$ (*universe*);
- □ a function $f^{\mathbb{M}} : A^n \longrightarrow A$, for each $f^n \in \mathcal{F}$;
- □ a relation $P^{\mathbb{M}} \subseteq A^n$, for each $P^n \in \mathcal{P}$.

Let $\mathbb{M} = (\mathcal{A}, \{f^{\mathbb{M}}\}_{f \in \mathcal{F}}, \{P^{\mathbb{M}}\}_{P \in \mathcal{P}})$ be an interpretation for $(\mathcal{P}, \mathcal{F})$, $x \in \mathcal{V}$ and φ, ψ in WFF$_{\mathcal{L}_{FO}}$:

1. $\mathbb{M} \models P(t_1, \ldots, t_n)$ iff $(t_1, \ldots, t_n) \in P^{\mathbb{M}}$;
2. $\mathbb{M} \models \neg\varphi$ iff $\mathbb{M} \not\models (\varphi)$;
3. $\mathbb{M} \models \varphi \wedge \psi$ iff $\mathbb{M} \models \varphi$ e $\mathbb{M} \models \psi$;
4. $\mathbb{M} \models \varphi \vee \psi$ iff $\mathbb{M} \models \varphi$ or $\mathbb{M} \models \psi$; ;
5. $\mathbb{M} \models \varphi \rightarrow \psi$ iff $\mathbb{M} \models \neg\varphi \vee \psi$;
6. $\mathbb{M} \models \forall x \varphi$ iff $\mathbb{M} \models \varphi[a \backslash x]$ for all $a \in \mathcal{A}$;
7. $\mathbb{M} \models \exists x \varphi$ iff $\mathbb{M} \models \varphi[a \backslash x]$ for some $a \in \mathcal{A}$.

Logic

▷ The Early Days

In the beginning there was darkness...
Presburger Procedure
The Great Triumph
The Logic Theory Machine
Example - Substitution Method
Example - Detachment Method
Example - Chaining
Comparing LTM with the British Museum
Performance and Completeness
Wang and Gao
Example
Success?
Paul Gilmore and Herbert Gelernter
Dag Prawitz
Example: $\forall x P(x) \to \exists y P(y)$
Some notes

DPLL

Resolution

The Early Days

In the beginning there was darkness...

Logic

▷ The Early Days

In the beginning there was ▷ darkness...

Presburger Procedure

The Great Triumph

The Logic Theory Machine

Example - Substitution Method

Example - Detachment Method

Example - Chaining

Comparing LTM with the British Museum

Performance and Completeness

Wang and Gao

Example

Success?

Paul Gilmore and Herbert Gelernter

Dag Prawitz

Example: $\forall x P(x) \rightarrow \exists y P(y)$

Some notes

DPLL

Resolution

- The development of automated reasoning tools started shorter after computers were available: the first Turing-complete programmable digital computer, the ENIAC, was operational in 1946; in 1954, a program written by Martin Davis produces a proof for the Presburger arithmetic (published in 1957).
- George Collins implemented parts of Tarski's decision procedure for elementary algebra on an IBM 704. Also published in 1957.
- Most procedures relied on (naive) enumerations of proofs and/or models.
- First results were not encouraging: even short proofs for known theorems could not be carried out by computers. In the words of Martin Davis on his program: "Its great triumph was to prove that the sum of two even numbers is even."

Presburger Procedure

Logic
▷ The Early Days
In the beginning there was darkness...
▷ Presburger Procedure
The Great Triumph
The Logic Theory Machine
Example - Substitution Method
Example - Detachment Method
Example - Chaining
Comparing LTM with the British Museum
Performance and Completeness
Wang and Gao
Example
Success?
Paul Gilmore and Herbert Gelernter
Dag Prawitz
Example: $\forall x P(x) \to \exists y P(y)$
Some notes
DPLL
Resolution

- In 1929, Presburger shows that the first order theory of addition on the natural numbers is decidable [Pre29].
- Axioms for the Presburger arithmetic are:

  1. $\neg(0 = x + 1)$
  2. $x + 1 = y + 1 \to x = y$
  3. $x + 0 = x$
  4. $x + (y + 1) = (x + y) + 1$
  5. $P(0) \wedge \forall x(P(x) \to P(x + 1)) \to \forall y P(y)$

- The procedure consists of quantifier elimination on formulae in antiprenex disjunctive normal form, whose implementation on a JOHNNIAC was reported in [Dav57].

The Great Triumph

Logic
▷ The Early Days
In the beginning there was darkness...
Presburger Procedure
▷ The Great Triumph
The Logic Theory Machine
Example - Substitution Method
Example - Detachment Method
Example - Chaining
Comparing LTM with the British Museum
Performance and Completeness
Wang and Gao
Example
Success?
Paul Gilmore and Herbert Gelernter
Dag Prawitz
Example: $\forall x P(x) \to \exists y P(y)$
Some notes
DPLL
Resolution

- The JOHNNIAC at Princeton:
  - John von Neumann Numerical Integrator and Automatic Computer, Rand Corp.
  - Vacuum tube machine.
  - 1024 40-bit word machine (yes, this is only 5KB!).
  - MFTBF (Mean Flight Time Between Failures) of 10 minutes.
  - Add time: 50ms (that's about 0.02 MIPS).
- Fischer and Rabin showed that complexity is double-exponential [FR74]. Upper bound is triple-exponential [Opp78].

The Logic Theory Machine

Logic
▷ The Early Days
In the beginning there was darkness...
Presburger Procedure
The Great Triumph
▷ The Logic Theory Machine
Example - Substitution Method
Example - Detachment Method
Example - Chaining
Comparing LTM with the British Museum
Performance and Completeness
Wang and Gao
Example
Success?
Paul Gilmore and Herbert Gelernter
Dag Prawitz
Example: $\forall x P(x) \rightarrow \exists y P(y)$
Some notes
DPLL
Resolution

- It was devised and implemented by Newell, Shaw, and Simon [NS56, NS57, NSS57].
- It was implemented on a JOHNNIAC at the Rand Corporation, using a programming language devised by the authors.
- The calculus was based on that of the Principia: five axioms together with substitution and modus ponens. It also includes a rule for replacement of definitions. The proof is based on the construction of a tree, which handles space by using heuristics associated to detachment, backward and forward chaining. If we want to prove φ:
  - find a sentence ψ such that $\psi \rightarrow \varphi$ is provable; reduce the proof of φ to the proof of ψ (detachment);
  - if φ is of the form $\psi \rightarrow \chi$, find $\psi \rightarrow \chi'$ which is provable; reduce the proof of φ to the proof of $\chi' \rightarrow \chi$ (chaining);
  - if φ is of the form $\psi \rightarrow \chi$, find $\chi' \rightarrow \chi$ which is provable; reduce the proof of φ to the proof of $\psi \rightarrow \chi'$ (chaining).

Example - Substitution Method

Logic
▷ The Early Days
In the beginning there was darkness...
Presburger Procedure
The Great Triumph
The Logic Theory Machine
▷ Example - Substitution Method
Example - Detachment Method
Example - Chaining
Comparing LTM with the British Museum
Performance and Completeness
Wang and Gao
Example
Success?
Paul Gilmore and Herbert Gelernter
Dag Prawitz
Example: $\forall x P(x) \to \exists y P(y)$
Some notes
DPLL
Resolution

1.01	$(p \to q) = (\neg p \vee q)$
1.2	$(p \vee p) \to p$
1.3	$p \to (q \vee p)$
1.4	$(p \vee q) \to (q \vee p)$
1.5	$((p \vee q) \vee r) \to ((q \vee p) \vee r)$
1.6	$(p \to q) \to ((r \vee p) \to (r \vee q))$

Example - Substitution Method

Logic
▷ The Early Days
In the beginning there was darkness...
Presburger Procedure
The Great Triumph
The Logic Theory Machine
Example - Substitution
▷ Method
Example - Detachment Method
Example - Chaining
Comparing LTM with the British Museum
Performance and Completeness
Wang and Gao
Example
Success?
Paul Gilmore and Herbert Gelernter
Dag Prawitz
Example: $\forall x P(x) \rightarrow \exists y P(y)$
Some notes
DPLL
Resolution

1.01	$(p \rightarrow q) = (\neg p \vee q)$
1.2	$(p \vee p) \rightarrow p$
1.3	$p \rightarrow (q \vee p)$
1.4	$(p \vee q) \rightarrow (q \vee p)$
1.5	$((p \vee q) \vee r) \rightarrow ((q \vee p) \vee r)$
1.6	$(p \rightarrow q) \rightarrow ((r \vee p) \rightarrow (r \vee q))$

2.01	$?(p \rightarrow \neg p) \rightarrow \neg p$

Logic

▷ **The Early Days**

In the beginning there was darkness...
Presburger Procedure
The Great Triumph
The Logic Theory Machine
▷ Example - Substitution Method
Example - Detachment Method
Example - Chaining
Comparing LTM with the British Museum
Performance and Completeness
Wang and Gao
Example
Success?
Paul Gilmore and Herbert Gelernter
Dag Prawitz
Example: $\forall x P(x) \to \exists y P(y)$
Some notes

DPLL

Resolution

1.01	$(p \to q) = (\neg p \vee q)$	
1.2	$(p \vee p) \to p$	
1.3	$p \to (q \vee p)$	
1.4	$(p \vee q) \to (q \vee p)$	
1.5	$((p \vee q) \vee r) \to ((q \vee p) \vee r)$	
1.6	$(p \to q) \to ((r \vee p) \to (r \vee q))$	
2.01	$?(p \to \neg p) \to \neg p$	
1.2	$!(A \vee A) \to A$	$[1.2, \text{similarity}]$

Example - Substitution Method

Logic
▷ The Early Days
In the beginning there was darkness...
Presburger Procedure
The Great Triumph
The Logic Theory Machine
Example - Substitution
▷ Method
Example - Detachment Method
Example - Chaining
Comparing LTM with the British Museum
Performance and Completeness
Wang and Gao
Example
Success?
Paul Gilmore and Herbert Gelernter
Dag Prawitz
Example: $\forall x P(x) \to \exists y P(y)$
Some notes
DPLL
Resolution

1.01	$(p \to q) = (\neg p \vee q)$	
1.2	$(p \vee p) \to p$	
1.3	$p \to (q \vee p)$	
1.4	$(p \vee q) \to (q \vee p)$	
1.5	$((p \vee q) \vee r) \to ((q \vee p) \vee r)$	
1.6	$(p \to q) \to ((r \vee p) \to (r \vee q))$	
2.01	$?(p \to \neg p) \to \neg p$	
1.2	$!(A \vee A) \to A$	$[1.2, \text{similarity}]$
$1.2'$	$!(\neg B \vee \neg B) \to \neg B$	$[1.2, \text{substitution}]$

Example - Substitution Method

Logic
▷ The Early Days
In the beginning there was darkness...
Presburger Procedure
The Great Triumph
The Logic Theory Machine
Example - Substitution
▷ Method
Example - Detachment Method
Example - Chaining
Comparing LTM with the British Museum
Performance and Completeness
Wang and Gao
Example
Success?
Paul Gilmore and Herbert Gelernter
Dag Prawitz
Example: $\forall x P(x) \rightarrow \exists y P(y)$
Some notes
DPLL
Resolution

1.01	$(p \rightarrow q) = (\neg p \vee q)$	
1.2	$(p \vee p) \rightarrow p$	
1.3	$p \rightarrow (q \vee p)$	
1.4	$(p \vee q) \rightarrow (q \vee p)$	
1.5	$((p \vee q) \vee r) \rightarrow ((q \vee p) \vee r)$	
1.6	$(p \rightarrow q) \rightarrow ((r \vee p) \rightarrow (r \vee q))$	
2.01	$?(p \rightarrow \neg p) \rightarrow \neg p$	
1.2	$!(A \vee A) \rightarrow A$	[1.2, similarity]
$1.2'$	$!(\neg B \vee \neg B) \rightarrow \neg B$	[1.2, substitution]
$1.2''$	$!(B \rightarrow \neg B) \rightarrow \neg B$	[$1.2'$, replacement]

Example - Substitution Method

Logic
▷ The Early Days
In the beginning there was darkness...
Presburger Procedure
The Great Triumph
The Logic Theory Machine
Example - Substitution ▷ Method
Example - Detachment Method
Example - Chaining
Comparing LTM with the British Museum
Performance and Completeness
Wang and Gao
Example
Success?
Paul Gilmore and Herbert Gelernter
Dag Prawitz
Example: $\forall x P(x) \to \exists y P(y)$
Some notes
DPLL
Resolution

1.01	$(p \to q) = (\neg p \lor q)$	
1.2	$(p \lor p) \to p$	
1.3	$p \to (q \lor p)$	
1.4	$(p \lor q) \to (q \lor p)$	
1.5	$((p \lor q) \lor r) \to ((q \lor p) \lor r)$	
1.6	$(p \to q) \to ((r \lor p) \to (r \lor q))$	
2.01	$?(p \to \neg p) \to \neg p$	
1.2	$!(A \lor A) \to A$	[1.2, similarity]
$1.2'$	$!(\neg B \lor \neg B) \to \neg B$	[1.2, substitution]
$1.2''$	$!(B \to \neg B) \to \neg B$	[$1.2'$, replacement]
$1.2'''$	$!(p \to \neg p) \to \neg p$	[$1.2''$, replacement]

Example - Detachment Method

$2.01 \quad ?(p \to q) \to ((q \to r) \to (p \to r))$

Example - Detachment Method

2.01 $?(p \to q) \to ((q \to r) \to (p \to r))$

2.04 $!(A \to (B \to C)) \to (B \to (A \to C))$ [theorem]

Example - Detachment Method

2.01 $?(p \to q) \to ((q \to r) \to (p \to r))$

2.04 $!(A \to (B \to C)) \to (B \to (A \to C))$ [theorem]

2.04′ $!((q \to r) \to ((p \to q) \to (p \to r))) \to ((p \to q) \to ((q \to r) \to (p \to r)))$ [2.04, substitution]

Example - Detachment Method

$$2.01 \quad ?(p \to q) \to ((q \to r) \to (p \to r))$$

$$2.04 \quad !(A \to (B \to C)) \to (B \to (A \to C)) \qquad \text{[theorem]}$$

$$2.04' \quad !((q \to r) \to ((p \to q) \to (p \to r))) \to ((p \to q) \to ((q \to r) \to (p \to r)))$$

$$[2.04, \text{substitution}]$$

$$2.04'L \quad ?(q \to r) \to ((p \to q) \to (p \to r))$$

Example - Detachment Method

2.01	$?(p \to q) \to ((q \to r) \to (p \to r))$	
2.04	$!(A \to (B \to C)) \to (B \to (A \to C))$	[theorem]
$2.04'$	$!((q \to r) \to ((p \to q) \to (p \to r))) \to ((p \to q) \to ((q \to r) \to (p \to r)))$	[2.04, substitution]
$2.04'L$	$?(q \to r) \to ((p \to q) \to (p \to r))$	
2.05	$!(A \to B) \to ((C \to A) \to (C \to B))$	[theorem]

Example - Detachment Method

2.01	$?(p \to q) \to ((q \to r) \to (p \to r))$	
2.04	$!(A \to (B \to C)) \to (B \to (A \to C))$	[theorem]
$2.04'$	$!((q \to r) \to ((p \to q) \to (p \to r))) \to ((p \to q) \to ((q \to r) \to (p \to r)))$	[2.04, substitution]
$2.04'L$	$?(q \to r) \to ((p \to q) \to (p \to r))$	
2.05	$!(A \to B) \to ((C \to A) \to (C \to B))$	[theorem]
$2.05'$	$!(q \to r) \to ((p \to q) \to (p \to r))$	[2.05, substitution]

Example - Detachment Method

$$
\begin{array}{rll}
2.01 & ?(p \to q) \to ((q \to r) \to (p \to r)) & \\
 & & \\
2.04 & !(A \to (B \to C)) \to (B \to (A \to C)) & [\textsf{theorem}] \\
2.04' & !((q \to r) \to ((p \to q) \to (p \to r))) \to ((p \to q) \to ((q \to r) \to (p \to r))) & \\
 & & [2.04, \textsf{substitution}] \\
2.04'L & ?(q \to r) \to ((p \to q) \to (p \to r)) & \\
2.05 & !(A \to B) \to ((C \to A) \to (C \to B)) & [\textsf{theorem}] \\
2.05' & !(q \to r) \to ((p \to q) \to (p \to r)) & \\
 & & [2.05, \textsf{substitution}] \\
2.01 & !(p \to q) \to ((q \to r) \to (p \to r)) & \\
 & & [2.05', 2.04', \textsf{detachment}]
\end{array}
$$

Example - Detachment Method

2.01	$?(p \to q) \to ((q \to r) \to (p \to r))$	
2.04	$!(A \to (B \to C)) \to (B \to (A \to C))$	[theorem]
$2.04'$	$!((q \to r) \to ((p \to q) \to (p \to r))) \to ((p \to q) \to ((q \to r) \to (p \to r)))$	[2.04, substitution]
$2.04'L$	$?(q \to r) \to ((p \to q) \to (p \to r))$	
2.05	$!(A \to B) \to ((C \to A) \to (C \to B))$	[theorem]
$2.05'$	$!(q \to r) \to ((p \to q) \to (p \to r))$	[2.05, substitution]
2.01	$!(p \to q) \to ((q \to r) \to (p \to r))$	[$2.05'$, $2.04'$, detachment]

Example - Chaining

Logic

▷ The Early Days
In the beginning there was darkness...
Presburger Procedure
The Great Triumph
The Logic Theory Machine
Example - Substitution Method
Example - Detachment Method
▷ Example - Chaining
Comparing LTM with the British Museum
Performance and Completeness
Wang and Gao
Example
Success?
Paul Gilmore and Herbert Gelernter
Dag Prawitz
Example: $\forall x P(x) \rightarrow \exists y P(y)$
Some notes

DPLL

Resolution

To prove $A \rightarrow C$, find B such that:

$$((A \rightarrow B) \wedge (B \rightarrow C)) \rightarrow (A \rightarrow C)$$

Example - Chaining

Logic

▷ The Early Days
In the beginning there was darkness...
Presburger Procedure
The Great Triumph
The Logic Theory Machine
Example - Substitution Method
Example - Detachment Method
▷ Example - Chaining
Comparing LTM with the British Museum
Performance and Completeness
Wang and Gao
Example
Success?
Paul Gilmore and Herbert Gelernter
Dag Prawitz
Example: $\forall x P(x) \rightarrow \exists y P(y)$
Some notes

DPLL

Resolution

To prove $A \rightarrow C$, find B such that:

$$((A \rightarrow B) \wedge (B \rightarrow C)) \rightarrow (A \rightarrow C)$$

$$1.2 \quad (p \vee p) \rightarrow p$$

$$2.08 \quad ?p \rightarrow p$$

Example - Chaining

Logic
▷ The Early Days
In the beginning there was darkness...
Presburger Procedure
The Great Triumph
The Logic Theory Machine
Example - Substitution Method
Example - Detachment Method
▷ Example - Chaining
Comparing LTM with the British Museum
Performance and Completeness
Wang and Gao
Example
Success?
Paul Gilmore and Herbert Gelernter
Dag Prawitz
Example: $\forall x P(x) \rightarrow \exists y P(y)$
Some notes
DPLL
Resolution

To prove $A \rightarrow C$, find B such that:

$$((A \rightarrow B) \wedge (B \rightarrow C)) \rightarrow (A \rightarrow C)$$

$$1.2 \quad (p \vee p) \rightarrow p$$

$$2.08 \quad ?p \rightarrow p$$

$$2.07 \quad !A \rightarrow (A \vee A) \quad \text{[theorem]}$$

Example - Chaining

Logic
▷ The Early Days
In the beginning there was darkness...
Presburger Procedure
The Great Triumph
The Logic Theory Machine
Example - Substitution Method
Example - Detachment Method
▷ Example - Chaining
Comparing LTM with the British Museum
Performance and Completeness
Wang and Gao
Example
Success?
Paul Gilmore and Herbert Gelernter
Dag Prawitz
Example: $\forall x P(x) \to \exists y P(y)$
Some notes
DPLL
Resolution

To prove $A \to C$, find B such that:

$$((A \to B) \land (B \to C)) \to (A \to C)$$

$$1.2 \quad (p \lor p) \to p$$

$$2.08 \quad ?p \to p$$

$$2.07 \quad !A \to (A \lor A) \quad [\text{theorem}]$$
$$2.07' \quad !p \to (p \lor p) \quad [2.07, \text{substitution}]$$

Example - Chaining

Logic
▷ The Early Days
In the beginning there was darkness...
Presburger Procedure
The Great Triumph
The Logic Theory Machine
Example - Substitution Method
Example - Detachment Method
▷ Example - Chaining
Comparing LTM with the British Museum
Performance and Completeness
Wang and Gao
Example
Success?
Paul Gilmore and Herbert Gelernter
Dag Prawitz
Example: $\forall x P(x) \rightarrow \exists y P(y)$
Some notes
DPLL
Resolution

To prove $A \rightarrow C$, find B such that:

$$((A \rightarrow B) \wedge (B \rightarrow C)) \rightarrow (A \rightarrow C)$$

$$1.2 \quad (p \vee p) \rightarrow p$$

$$2.08 \quad ?p \rightarrow p$$

2.07	$!A \rightarrow (A \vee A)$	[theorem]
$2.07'$	$!p \rightarrow (p \vee p)$	$[2.07,$ substitution$]$
$2.07''$	$!p \rightarrow p$	$[2.07', 1.2,$ chaining$]$

Comparing LTM with the British Museum

Logic
▷ The Early Days
In the beginning there was darkness...
Presburger Procedure
The Great Triumph
The Logic Theory Machine
Example - Substitution Method
Example - Detachment Method
Example - Chaining
▷ Comparing LTM with the British Museum
Performance and Completeness
Wang and Gao
Example
Success?
Paul Gilmore and Herbert Gelernter
Dag Prawitz
Example: $\forall x P(x) \rightarrow \exists y P(y)$
Some notes
DPLL
Resolution

- The British Museum algorithm enumerates all theorems in a forward manner from axioms to the proof (if there is one).
- The Logic Theory Machine works in a backward manner in order to find sub-problems from the desired axioms or known theorems.

> (...) working backward may be analogous to the ease with which a needle can find its way out of a haystack, compared with the difficulty of someone finding the lone needle in the haystack. [NSS57]

- The Logic Theory Machine could prove 38 of the first 52 theorems on Chapter 2 of the Principia (the remainder 14 problems could not be solved because of memory limitations).
- The program could use results previously proved: there was *learning* in theorem proving already!
- Backward reasoning is still one of the most used techniques in automated reasoning.

Logic
▷ The Early Days
In the beginning there was darkness...
Presburger Procedure
The Great Triumph
The Logic Theory Machine
Example - Substitution Method
Example - Detachment Method
Example - Chaining
Comparing LTM with the British Museum
▷ Performance and Completeness
Wang and Gao
Example
Success?
Paul Gilmore and Herbert Gelernter
Dag Prawitz
Example: $\forall x P(x) \rightarrow \exists y P(y)$
Some notes
DPLL
Resolution

- The JOHNNIAC at Rand: by 1956, tubes had already been replaced by transistors, main memory had been expanded to 4096 words and a drum, the external storage device, of 12K had been added; MFTBF was around 100 hours.
- It took ten seconds to prove:

$$(2.01) \qquad (p \rightarrow \neg p) \rightarrow \neg p.$$

- It took 12 minutes, using all 38 theorems already proved, to prove

$$(2.45) \qquad \neg(p \vee q) \rightarrow \neg p.$$

- Failure to produce a proof was reported after 23 minutes for:

$$(2.31) \qquad p \vee (q \vee r) \rightarrow (p \vee q) \vee r.$$

- The Logic Theory Machine is not a complete procedure (because of the way it deals with substitution and how it uses heuristics to guide the search for a proof).

Wang and Gao

Logic
▷ The Early Days
In the beginning there was darkness...
Presburger Procedure
The Great Triumph
The Logic Theory Machine
Example - Substitution Method
Example - Detachment Method
Example - Chaining
Comparing LTM with the British Museum
Performance and Completeness
▷ Wang and Gao
Example
Success?
Paul Gilmore and Herbert Gelernter
Dag Prawitz
Example: $\forall x P(x) \rightarrow \exists y P(y)$
Some notes
DPLL
Resolution

- It must be emphasised that the Logic Theory Machine was not designed for efficiency: it tries to mimic the way humans produce proofs.
- A calculus designed for efficiency was that of Wang: it's a two-sided sequent system, whose inference rules remove one operator each time.
- In the case of a split, both sequents must be proved.
- This work started at IBM in the Summer of 1958 and then moved to the Bell Labs in 1959-1960 [Wan60a, Wan60b, Wan61].
- Three programs were developed:
  - One for propositional logic;
  - One for the decidable part of predicate logic;
  - One for all predicate logic.

Example

Logic

▷ The Early Days
In the beginning there was darkness...
Presburger Procedure
The Great Triumph
The Logic Theory Machine
Example - Substitution Method
Example - Detachment Method
Example - Chaining
Comparing LTM with the British Museum
Performance and Completeness
Wang and Gao
▷ Example
Success?
Paul Gilmore and Herbert Gelernter
Dag Prawitz
Example: $\forall x P(x) \rightarrow \exists y P(y)$
Some notes

DPLL

Resolution

$p, \neg q \wedge r$	$\Rightarrow$	$p \wedge r$	[Initial Sequent]
$p, \neg q, r$	$\Rightarrow$	$p \wedge q$	[Left conjunction removal]
p, r	$\Rightarrow$	$q, p \wedge r$	[Left negation removal]

$p, r \quad \rightarrow \quad q, p$
[Right conjunction removal]

$p, r \quad \rightarrow \quad q, r$
[Right conjunction removal]

Success?

Logic
▷ The Early Days
In the beginning there was darkness...
Presburger Procedure
The Great Triumph
The Logic Theory Machine
Example - Substitution Method
Example - Detachment Method
Example - Chaining
Comparing LTM with the British Museum
Performance and Completeness
Wang and Gao
Example
▷ Success?
Paul Gilmore and Herbert Gelernter
Dag Prawitz
Example: $\forall x P(x) \rightarrow \exists y P(y)$
Some notes
DPLL
Resolution

- The first program, the one dedicated to propositional logic, took 37 minutes to prove the 220 theorems from the Principia.
- Techniques, based on pattern matching, were used during enumeration of substitutions (somewhat anticipating the results from Prawitz).

It is also worth noting that other authors have responded heavily to the heuristics proposed by Newell, Shaw, and Simon. See, for instance,

- B. Dunham and R. Fridshal and G. L. Sward: "A non-heuristic program for proving elementary logical theorems" [DFS59].
- The procedure used rules as splitting and pure literal reduction, which is later used by Davis and Putnam in their satisfiability procedure.

published under the chapter "Pattern recognition and machine learning" in the First International Conference on Information Processing, in Paris.

Paul Gilmore and Herbert Gelernter

Logic

▷ The Early Days

In the beginning there was darkness...

Presburger Procedure

The Great Triumph

The Logic Theory Machine

Example - Substitution Method

Example - Detachment Method

Example - Chaining

Comparing LTM with the British Museum

Performance and Completeness

Wang and Gao

Example

Success?

▷ Paul Gilmore and Herbert Gelernter

Dag Prawitz

Example: $\forall x P(x) \rightarrow \exists y P(y)$

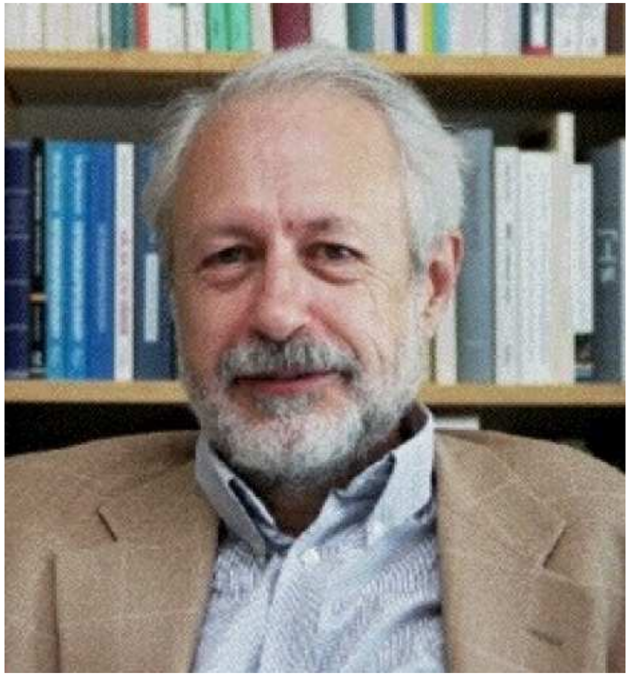
Some notes

DPLL

Resolution

- IBM Research at the Lamb Estate in Croton-on-Hudson, NY.
- Implemented Beth's semantic tableaux for first-order logic (although it was actually closer to Hintikka's method) in 1958 [Gel59, Gil60, Gil70].
- They lately got to know – and acknowledge the fact – that theirs were not the first implementation of a procedure for first-order logic (IBM 704).
  - Take as input a negated first-order formula in prenex form with matrix in disjunctive normal form.
  - Search for constants is about the same that the one proposed by Prawitz.

Dag Prawitz



- Prawitz developed a general calculus for predicate logic based on contradiction.
  - A derivation is a sequence of pairs: $(\phi, F), \ldots$
  - Each pair in the sequence is derived from the previous by application of "usual" inference rules for formulae without quantifiers.
  - A derivation from a quantified formula relies on some enumeration of the constants.
  - If the derivation contains two pairs (ψ, T) and (ψ, F), then a contradiction is found. Thus, the original formula is valid.

Example: $\forall x P(x) \rightarrow \exists y P(y)$

Logic

▷ The Early Days
In the beginning there was darkness...
Presburger Procedure
The Great Triumph
The Logic Theory Machine
Example - Substitution Method
Example - Detachment Method
Example - Chaining
Comparing LTM with the British Museum
Performance and Completeness
Wang and Gao
Example
Success?
Paul Gilmore and Herbert Gelernter
Dag Prawitz
▷ Example: $\forall x P(x) \rightarrow \exists y P(y)$
Some notes

DPLL

Resolution

□ $(\forall x P(x) \rightarrow \exists y P(y), F)$

Example: $\forall x P(x) \to \exists y P(y)$

Logic

▷ **The Early Days**

In the beginning there was darkness...
Presburger Procedure
The Great Triumph
The Logic Theory Machine
Example - Substitution Method
Example - Detachment Method
Example - Chaining
Comparing LTM with the British Museum
Performance and Completeness
Wang and Gao
Example
Success?
Paul Gilmore and Herbert Gelernter
Dag Prawitz
▷ **Example:** $\forall x P(x) \to \exists y P(y)$
Some notes

DPLL

Resolution

- $(\forall x P(x) \to \exists y P(y), F)$
- $(\forall x P(x) \to \exists y P(y), F), (\forall x P(x), T), (\exists y P(y), F)$

Logic
▷ The Early Days
In the beginning there was darkness...
Presburger Procedure
The Great Triumph
The Logic Theory Machine
Example - Substitution Method
Example - Detachment Method
Example - Chaining
Comparing LTM with the British Museum
Performance and Completeness
Wang and Gao
Example
Success?
Paul Gilmore and Herbert Gelernter
Dag Prawitz
▷ Example: $\forall x P(x) \to \exists y P(y)$
Some notes
DPLL
Resolution

- $(\forall x P(x) \to \exists y P(y), F)$
- $(\forall x P(x) \to \exists y P(y), F), (\forall x P(x), T), (\exists y P(y), F)$
- By the semantics of the universal operator, then for any constant c, we have that $(P(c), T)$. We consider an enumeration of all constants, $c_1, c_2, \ldots$, and take any constant:

Example: $\forall x P(x) \rightarrow \exists y P(y)$

Logic

▷ The Early Days

In the beginning there was darkness...

Presburger Procedure

The Great Triumph

The Logic Theory Machine

Example - Substitution Method

Example - Detachment Method

Example - Chaining

Comparing LTM with the British Museum

Performance and Completeness

Wang and Gao

Example

Success?

Paul Gilmore and Herbert Gelernter

Dag Prawitz

▷ Example: $\forall x P(x) \rightarrow \exists y P(y)$

Some notes

DPLL

Resolution

- $(\forall x P(x) \rightarrow \exists y P(y), F)$
- $(\forall x P(x) \rightarrow \exists y P(y), F), (\forall x P(x), T), (\exists y P(y), F)$
- By the semantics of the universal operator, then for any constant c, we have that $(P(c), T)$. We consider an enumeration of all constants, $c_1, c_2, \ldots$, and take any constant:
- $(P(c_1), T)$

Logic

▷ The Early Days

In the beginning there was darkness...

Presburger Procedure

The Great Triumph

The Logic Theory Machine

Example - Substitution Method

Example - Detachment Method

Example - Chaining

Comparing LTM with the British Museum

Performance and Completeness

Wang and Gao

Example

Success?

Paul Gilmore and Herbert Gelernter

Dag Prawitz

▷ Example: $\forall x P(x) \rightarrow \exists y P(y)$

Some notes

DPLL

Resolution

- $(\forall x P(x) \rightarrow \exists y P(y), F)$
- $(\forall x P(x) \rightarrow \exists y P(y), F), (\forall x P(x), T), (\exists y P(y), F)$
- By the semantics of the universal operator, then for any constant c, we have that $(P(c), T)$. We consider an enumeration of all constants, $c_1, c_2, \ldots$, and take any constant:
- $(P(c_1), T)$
- By the semantics of the existential operator, $(\exists y P(y), F)$ implies that for any constant c, we have that $P(c)$ is false. We select a constant that has been used before:

Example: $\forall x P(x) \to \exists y P(y)$

Logic
▷ The Early Days
In the beginning there was darkness...
Presburger Procedure
The Great Triumph
The Logic Theory Machine
Example - Substitution Method
Example - Detachment Method
Example - Chaining
Comparing LTM with the British Museum
Performance and Completeness
Wang and Gao
Example
Success?
Paul Gilmore and Herbert Gelernter
Dag Prawitz
▷ Example: $\forall x P(x) \to \exists y P(y)$
Some notes
DPLL
Resolution

- $(\forall x P(x) \to \exists y P(y), F)$
- $(\forall x P(x) \to \exists y P(y), F), (\forall x P(x), T), (\exists y P(y), F)$
- By the semantics of the universal operator, then for any constant c, we have that $(P(c), T)$. We consider an enumeration of all constants, $c_1, c_2, \ldots$, and take any constant:
- $(P(c_1), T)$
- By the semantics of the existential operator, $(\exists y P(y), F)$ implies that for any constant c, we have that $P(c)$ is false. We select a constant that has been used before:
- $(P(c_1), F)$

Logic
▷ The Early Days
In the beginning there was darkness...
Presburger Procedure
The Great Triumph
The Logic Theory Machine
Example - Substitution Method
Example - Detachment Method
Example - Chaining
Comparing LTM with the British Museum
Performance and Completeness
Wang and Gao
Example
Success?
Paul Gilmore and Herbert Gelernter
Dag Prawitz
Example: $\forall x P(x) \rightarrow \exists y P(y)$
▷ Some notes
DPLL
Resolution

Some notes

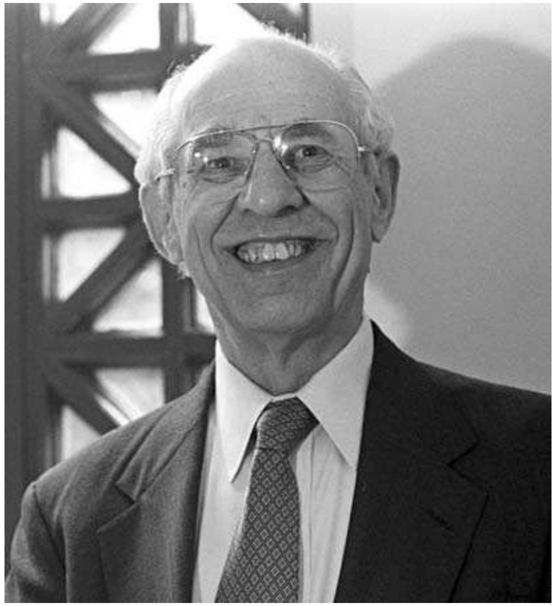
- This is a tableau!
- Constant selection is slightly more complicated than what was outlined in the example.
- Prawitz coded the procedure in a programming language he devised.
- His father, Håkan Prawtiz, hand-translated the code into machine code in 1957: FACIT EDB (transistorised computer), 2049 40-bits machine words, 8192 words drum.
- In 1958, Neri Voghera tested the procedure in a number of examples.
  - Outlined at the first IFIP, in Paris, 1960.
  - Published in 1960 [PPV60].
- The search for constants in Prawitz procedure were far from optimal.
- Prawitz introduced the use of *unification* in 1960: the method relied on the use of meta-variables which were replaced when *needed*, instead of using a fixed sequence.

Logic
The Early Days
▷ DPLL
Davis-Putnam
Davis-Putnam-Logemann-Loveland
Prenex Normal Form
Skolem Functions
Example
Herbrand Universe
Examples
Herbrand Base
Herbrand Interpretations
Formally
Example
Ground Instances of Clauses
Satisfiability of Ground Clauses
Unsatisfiability
Saturation
Saturation and Herbrand Interpretations
Herbrand's Theorems
DPLL
Example
Checking for consistency
Examples
More examples
Only using rule III
First-Order Example
DPLL

DPLL

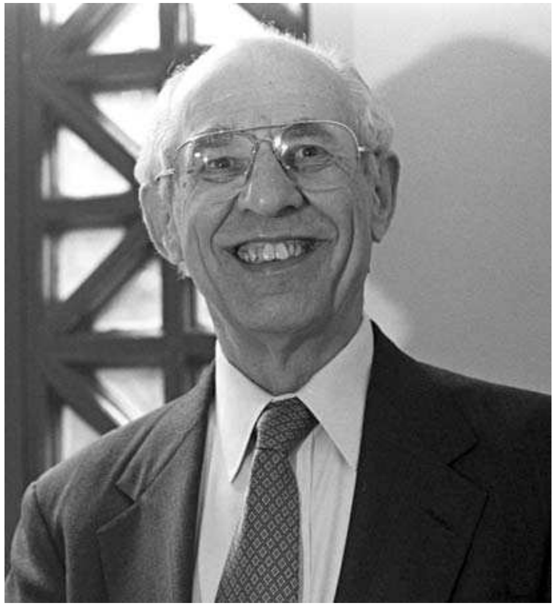
Logic
The Early Days
▷ DPLL
▷ Davis-Putnam
Davis-Putnam-Logemann-Loveland
Prenex Normal Form
Skolem Functions
Example
Herbrand Universe
Examples
Herbrand Base
Herbrand Interpretations
Formally
Example
Ground Instances of Clauses
Satisfiability of Ground Clauses
Unsatisfiability
Saturation
Saturation and Herbrand Interpretations
Herbrand's Theorems
DPLL
Example
Checking for consistency
Examples
More examples
Only using rule III
First-Order Example
DPLL

None of the mentioned procedures allowed functional symbols. Davis-Putnam, 1960, introduces Skolem functions and the Herbrand universe into the world of automated deduction. It also introduces clausal form. Finally, they proposed the unit resolution rule (which is there called "rule for the elimination of one-literal clauses") [DP60, DLL62].



Davis-Putnam-Logemann-Loveland

Logic
The Early Days
▷ DPLL
Davis-Putnam
▷ Davis-Putnam-Logemann-Loveland
Prenex Normal Form
Skolem Functions
Example
Herbrand Universe
Examples
Herbrand Base
Herbrand Interpretations
Formally
Example
Ground Instances of Clauses
Satisfiability of Ground Clauses
Unsatisfiability
Saturation
Saturation and Herbrand Interpretations
Herbrand's Theorems
DPLL
Example
Checking for consistency
Examples
More examples
Only using rule III
First-Order Example



Prenex Normal Form

Logic
The Early Days
▷ DPLL
Davis-Putnam
Davis-Putnam-Logemann-Loveland
▷ Prenex Normal Form
Skolem Functions
Example
Herbrand Universe
Examples
Herbrand Base
Herbrand Interpretations
Formally
Example
Ground Instances of Clauses
Satisfiability of Ground Clauses
Unsatisfiability
Saturation
Saturation and Herbrand Interpretations
Herbrand's Theorems
DPLL
Example
Checking for consistency
Examples
More examples
Only using rule III
First-Order Example

$$\varphi \stackrel{\text{def}}{=} Q_1 x_1 \ldots Q_n x_n \psi$$

where

- $\longrightarrow$ φ is a sentence: no free variables.
- $\longrightarrow$ Each Q_i, $1 \leq i \leq n$, is a quantifier.
- $\longrightarrow$ ψ is in Negation Normal Form (NNF).
- $\longrightarrow$ ψ is the matrix of φ.
- $\longrightarrow$ x_i, $1 \leq i \leq n$, is free in ψ.

Skolem Functions

Logic
The Early Days
▷ DPLL
Davis-Putnam
Davis-Putnam-Logemann-Loveland
Prenex Normal Form
▷ Skolem Functions
Example
Herbrand Universe
Examples
Herbrand Base
Herbrand Interpretations
Formally
Example
Ground Instances of Clauses
Satisfiability of Ground Clauses
Unsatisfiability
Saturation
Saturation and Herbrand Interpretations
Herbrand's Theorems
DPLL
Example
Checking for consistency
Examples
More examples
Only using rule III
First-Order Example
DPLL

Let

$$\varphi \stackrel{\text{def}}{=} Q_1 x_1 \ldots Q_n x_n \psi$$

be a formula in Prenex Normal Form where Q_i is the leftmost existential quantifier. If φ is satisfiable, then

$$\varphi' \stackrel{\text{def}}{=} Q_1 x_1 \ldots Q_{i-1} x_{i-1} Q_{i+1} x_{i+1} \ldots Q_n x_n \psi[x_i \mapsto f^{i-1}(x_1, \ldots, x_{i-1})]$$

where f^{i-1} is a new functional symbol, is satisfiable.

Obs.: The transformation is *satisfiability preserving*: if there is a model $\mathbb{M}$ for φ, then there is a model $\mathbb{M}$' for φ'.
Note that this is enough for proof methods based on contradiction: if there is no model $\mathbb{M}$' for φ', then there is no model $\mathbb{M}$ for φ.

Example

Logic

The Early Days

▷ DPLL

Davis-Putnam
Davis-Putnam-Logemann-Loveland
Prenex Normal Form
Skolem Functions
▷ Example
Herbrand Universe
Examples
Herbrand Base
Herbrand Interpretations
Formally
Example
Ground Instances of Clauses
Satisfiability of Ground Clauses
Unsatisfiability
Saturation
Saturation and Herbrand Interpretations
Herbrand's Theorems
DPLL
Example
Checking for consistency
Examples
More examples
Only using rule III
First-Order Example
DPLL

$$\neg(\forall x(P(x) \to Q(x)) \to (\forall x P(x) \to \forall x Q(x)))$$

Example

Logic
The Early Days
▷ DPLL
Davis-Putnam
Davis-Putnam-Logemann-Loveland
Prenex Normal Form
Skolem Functions
▷ Example
Herbrand Universe
Examples
Herbrand Base
Herbrand Interpretations
Formally
Example
Ground Instances of Clauses
Satisfiability of Ground Clauses
Unsatisfiability
Saturation
Saturation and Herbrand Interpretations
Herbrand's Theorems
DPLL
Example
Checking for consistency
Examples
More examples
Only using rule III
First-Order Example
DPLL

$$\neg(\forall x(P(x) \to Q(x)) \to (\forall x P(x) \to \forall x Q(x)))$$
$$\neg(\forall x(P(x) \to Q(x)) \to (\forall y P(y) \to \forall z Q(z)))$$

Example

Logic
The Early Days
▷ DPLL
Davis-Putnam
Davis-Putnam-Logemann-Loveland
Prenex Normal Form
Skolem Functions
▷ Example
Herbrand Universe
Examples
Herbrand Base
Herbrand Interpretations
Formally
Example
Ground Instances of Clauses
Satisfiability of Ground Clauses
Unsatisfiability
Saturation
Saturation and Herbrand Interpretations
Herbrand's Theorems
DPLL
Example
Checking for consistency
Examples
More examples
Only using rule III
First-Order Example
DPLL

$$\neg(\forall x(P(x) \to Q(x)) \to (\forall x P(x) \to \forall x Q(x)))$$
$$\neg(\forall x(P(x) \to Q(x)) \to (\forall y P(y) \to \forall z Q(z)))$$
$$\forall x(\neg P(x) \vee Q(x)) \wedge \forall y P(y) \wedge \neg \forall z Q(z)$$

Example

Logic
The Early Days
▷ DPLL
Davis-Putnam
Davis-Putnam-Logemann-Loveland
Prenex Normal Form
Skolem Functions
▷ Example
Herbrand Universe
Examples
Herbrand Base
Herbrand Interpretations
Formally
Example
Ground Instances of Clauses
Satisfiability of Ground Clauses
Unsatisfiability
Saturation
Saturation and Herbrand Interpretations
Herbrand's Theorems
DPLL
Example
Checking for consistency
Examples
More examples
Only using rule III
First-Order Example
DPLL

$$\neg(\forall x(P(x) \to Q(x)) \to (\forall x P(x) \to \forall x Q(x)))$$
$$\neg(\forall x(P(x) \to Q(x)) \to (\forall y P(y) \to \forall z Q(z)))$$
$$\forall x(\neg P(x) \lor Q(x)) \land \forall y P(y) \land \neg\forall z Q(z)$$
$$\forall x(\neg P(x) \lor Q(x)) \land \forall y P(y) \land \exists z \neg Q(z)$$

Example

Logic
The Early Days
▷ DPLL
Davis-Putnam
Davis-Putnam-Logemann-Loveland
Prenex Normal Form
Skolem Functions
▷ Example
Herbrand Universe
Examples
Herbrand Base
Herbrand Interpretations
Formally
Example
Ground Instances of Clauses
Satisfiability of Ground Clauses
Unsatisfiability
Saturation
Saturation and Herbrand Interpretations
Herbrand's Theorems
DPLL
Example
Checking for consistency
Examples
More examples
Only using rule III
First-Order Example
DPLL

$$\neg(\forall x(P(x) \to Q(x)) \to (\forall x P(x) \to \forall x Q(x)))$$
$$\neg(\forall x(P(x) \to Q(x)) \to (\forall y P(y) \to \forall z Q(z)))$$
$$\forall x(\neg P(x) \vee Q(x)) \wedge \forall y P(y) \wedge \neg\forall z Q(z)$$
$$\forall x(\neg P(x) \vee Q(x)) \wedge \forall y P(y) \wedge \exists z \neg Q(z)$$
$$\forall x \forall y \exists z((\neg P(x) \vee Q(x)) \wedge P(y) \wedge \neg Q(z))$$

Example

Logic
The Early Days
▷ DPLL
Davis-Putnam
Davis-Putnam-Logemann-Loveland
Prenex Normal Form
Skolem Functions
▷ Example
Herbrand Universe
Examples
Herbrand Base
Herbrand Interpretations
Formally
Example
Ground Instances of Clauses
Satisfiability of Ground Clauses
Unsatisfiability
Saturation
Saturation and Herbrand Interpretations
Herbrand's Theorems
DPLL
Example
Checking for consistency
Examples
More examples
Only using rule III
First-Order Example
DPLL

$$\neg(\forall x(P(x) \rightarrow Q(x)) \rightarrow (\forall x P(x) \rightarrow \forall x Q(x)))$$
$$\neg(\forall x(P(x) \rightarrow Q(x)) \rightarrow (\forall y P(y) \rightarrow \forall z Q(z)))$$
$$\forall x(\neg P(x) \vee Q(x)) \wedge \forall y P(y) \wedge \neg\forall z Q(z)$$
$$\forall x(\neg P(x) \vee Q(x)) \wedge \forall y P(y) \wedge \exists z \neg Q(z)$$
$$\forall x \forall y \exists z((\neg P(x) \vee Q(x)) \wedge P(y) \wedge \neg Q(z))$$
$$\forall x \forall y((\neg P(x) \vee Q(x)) \wedge P(y) \wedge \neg Q(f^2(x, y)))$$

Logic
The Early Days
▷ DPLL
Davis-Putnam
Davis-Putnam-Logemann-Loveland
Prenex Normal Form
Skolem Functions
Example
▷ Herbrand Universe
Examples
Herbrand Base
Herbrand Interpretations
Formally
Example
Ground Instances of Clauses
Satisfiability of Ground Clauses
Unsatisfiability
Saturation
Saturation and Herbrand Interpretations
Herbrand's Theorems
DPLL
Example
Checking for consistency
Examples
More examples
Only using rule III
First-Order Example
DPLL

Let $\mathcal{S}$ be a set of clauses. The Herbrand Universe $\mathcal{H}$ associated with $\mathcal{S}$ is a set of ground terms constructed as follows:

- if there is f^0 which occurs in $\mathcal{S}$, then $f^0 \in \mathcal{H}$; otherwise, let $\mathcal{H} = \{a\}$;
- if $t_1, \ldots, t_n$ are terms in $\mathcal{H}$ and f^n occurs in $\mathcal{S}$, then $f^n(t_1, \ldots, t_n) \in \mathcal{H}$.

That is, the Herbrand Universe (for a set) is the set of all ground terms (based on that set).

Examples

Logic
The Early Days
▷ DPLL
Davis-Putnam
Davis-Putnam-Logemann-Loveland
Prenex Normal Form
Skolem Functions
Example
Herbrand Universe
▷ Examples
Herbrand Base
Herbrand Interpretations
Formally
Example
Ground Instances of Clauses
Satisfiability of Ground Clauses
Unsatisfiability
Saturation
Saturation and Herbrand Interpretations
Herbrand's Theorems
DPLL
Example
Checking for consistency
Examples
More examples
Only using rule III
First-Order Example
DPLL

- No functions: $\mathcal{S} = \{P(x) \vee Q(x), P(x), \neg Q(x)\}$

$$\mathcal{H} = \{a\}$$

- One function: $\mathcal{S} = \{P(x) \vee Q(x), P(x), \neg Q(f(x))\}$

$$\mathcal{H} = \{a, f(a), f(f(a)), f(f(f(a))), \ldots\}$$

- One function (our working example):
$\mathcal{S} = \{P(x) \vee Q(x), P(y), \neg Q(f(x,y))\}$

$$\mathcal{H} = \{a, f(a,a), f(a, f(a,a)), f(f(a,a),a), f(f(a,a), f(a,a)), \ldots\}$$

- Two functions: $\mathcal{S} = \{P(x) \vee Q(x), P(f(x)), \neg Q(g(x))\}$

$$\mathcal{H} = \{a, f(a), g(a), f(f(a)), f(g(a)), g(f(a)), g(g(a)), \ldots\}$$

Herbrand Base

Let $\mathcal{S}$ be a set of clauses and $\mathcal{H}$ its Herbrand universe. The Herbrand base $\mathcal{B}$ for $\mathcal{S}$ is the set of all ground atoms $P^n(t_1, \ldots, t_n)$, where P^n occurs in $\mathcal{S}$ and t_i is in $\mathcal{H}$, for $i = 1, \ldots, n$.
Examples:

- No functions: $\mathcal{S} = \{P(x) \vee Q(x), P(x), \neg Q(x)\}, \mathcal{H} = \{a\}$

$$\{P(a), Q(a)\}$$

- One function:
  $\mathcal{S} = \{P(x) \vee Q(x), P(x), \neg Q(f(x))\}, \mathcal{H} = \{a, f(a), f(f(a)), f(f(f(a))), \ldots\}$

$$\{P(a), Q(a), P(f(a)), Q(f(a)), P(f(f(a))), Q(f(f(a))), \ldots\}$$

- For our working example: $\mathcal{S} = \{\neg P(x) \vee Q(x), P(y), \neg Q(f^2(x, y))\}$,
  $\mathcal{H} = \{a, f^2(a, a), f^2(a, f^2(a, a)), f^2(f^2(a, a), a), f^2(f^2(a, a), f^2(a, a)), \ldots\}$

$$\{P(a), Q(a), P(f^2(a, a)), Q(f^2(a, a)), P(f^2(a, f(a, a))), P(f^2(f(a, a), a)), P(f^2(f(a, a), f^2(a, a)))$$

Logic
The Early Days
▷ DPLL
Davis-Putnam
Davis-Putnam-Logemann-Loveland
Prenex Normal Form
Skolem Functions
Example
Herbrand Universe
Examples
Herbrand Base
▷ Herbrand Interpretations
Formally
Example
Ground Instances of Clauses
Satisfiability of Ground Clauses
Unsatisfiability
Saturation
Saturation and Herbrand Interpretations
Herbrand's Theorems
DPLL
Example
Checking for consistency
Examples
More examples
Only using rule III
First-Order Example
DPLL

Let $\mathcal{S}$ be a set of clauses and $\mathcal{H}$ be its Herbrand universe.

- A *Herbrand interpretation* provides:
  - A domain: $\mathcal{H}$
  - An assignment for constants to an element of the domain;
  - An assignment for functional symbols to an element of the domain;
  - An assignment for predicate symbols to $\{\bot, \top\}$

Formally

Logic
The Early Days
▷ DPLL
Davis-Putnam
Davis-Putnam-Logemann-Loveland
Prenex Normal Form
Skolem Functions
Example
Herbrand Universe
Examples
Herbrand Base
Herbrand Interpretations
▷ Formally
Example
Ground Instances of Clauses
Satisfiability of Ground Clauses
Unsatisfiability
Saturation
Saturation and Herbrand Interpretations
Herbrand's Theorems
DPLL
Example
Checking for consistency
Examples
More examples
Only using rule III
First-Order Example
DPLL

Let $\mathcal{S}$ be a set of clauses, $\mathcal{H}$ be its Herbrand universe and $\mathcal{B}$ be a Herbrand base for $\mathcal{S}$. Let $\mathcal{I} = \langle \mathcal{H}, \mathcal{B} \rangle$. We say that $\mathcal{I}$ is a *Herbrand interpretation* if $t^{\mathcal{B}} = t$ for all terms t.
That is,

- If c is a constant, then $c^{\mathcal{I}} = c$;
- If f is a n-ary functional symbol, then $f^{\mathcal{I}}(h_1, \ldots, h_n) = f(h_1, \ldots, h_n)$.
- There are no restrictions for predicate symbols.
  If $\mathcal{B} = \{P_1, \ldots, P_n, \ldots\}$, then take $\mathcal{I} = \{m_1, \ldots, m_n, \ldots\}$, where $m_i = P_i$ or $m_i = \neg P_i$.

Example

$$\mathcal{S} = \{P(x) \vee Q(x), P(y), \neg Q(f^2(x,y))\}$$

$$\mathcal{H} = \{a, f^2(a,a), f^2(a, f^2(a,a)), f^2(f^2(a,a),a), f^2(f^2(a,a), f^2(a,a)), \ldots\}$$

$$\{P(a), Q(a), P(f^2(a,a)), Q(f^2(a,a)), P(f^2(a,f(a))), P(f^2(f(a),a)), P(f^2(f(a), f^2(a,a))), \ldots$$

Possible interpretations:

$\mathcal{I}_1 = \{P(a), Q(a), P(f^2(a,a)), Q(f^2(a,a)), P(f^2(a, f^2(a,a))), P(f^2(f^2(a,a),a)), \ldots\}$

$\mathcal{I}_2 = \{\neg P(a), Q(a), P(f^2(a,a)), Q(f^2(a,a)), P(f^2(a, f^2(a,a))), P(f^2(f^2(a,a),a)), \ldots\}$

$\mathcal{I}_3 = \{P(a), \neg Q(a), P(f^2(a,a)), Q(f^2(a,a)), P(f^2(a, f^2(a,a))), P(f^2(f^2(a,a),a)), \ldots\}$

$\ldots$

Ground Instances of Clauses

Logic
The Early Days
▷ DPLL
Davis-Putnam
Davis-Putnam-Logemann-Loveland
Prenex Normal Form
Skolem Functions
Example
Herbrand Universe
Examples
Herbrand Base
Herbrand Interpretations
Formally
Example
▷ Ground Instances of Clauses
Satisfiability of Ground Clauses
Unsatisfiability
Saturation
Saturation and Herbrand Interpretations
Herbrand's Theorems
DPLL
Example
Checking for consistency
Examples
More examples
Only using rule III
First-Order Example
DPLL

Let $\mathcal{S}$ be a set of clauses and $\mathcal{H}$ its Herbrand universe. A ground instance of a clause C, $C \in \mathcal{S}$, for a Herbrand Base $\mathcal{H}$ (of $\mathcal{S}$) is the clause $C' = C[t_{ji} \backslash x_i]$, for all x_i occurring in C and $t_{ji} \in \mathcal{H}$.
Examples:

- No functions: $\mathcal{S} = \{P(x) \vee Q(x), P(x), \neg Q(x)\}, \mathcal{H} = \{a\}$, $\mathcal{B} = \{P(a), Q(a)\}$
  - $P(a) \vee Q(a), P(a), \neg Q(a)$ are the only ground instances for $\mathcal{B}$
- For our working example: $\mathcal{S} = \{\neg P(x) \vee Q(x), P(y), \neg Q(f^2(x,y))\}$, $\mathcal{H} =$ $\{a, f^2(a,a), f^2(a, f^2(a,a)), f^2(f^2(a,a),a), f^2(f^2(a,a), f^2(a,a)), \ldots\}$, $\mathcal{B} = \{\{P(a), Q(a), P(f^2(a,a)), Q(f^2(a,a)), \ldots\}$
  - $P(a) \vee Q(a)$
  - $P(f^2(a,a)) \vee Q(f^2(a,a))$
  - $P(f^2(a, f^2(a,a))) \vee Q(f^2(a, f^2(a,a)))$
  - $\ldots$

Satisfiability of Ground Clauses

Logic
The Early Days
▷ DPLL
Davis-Putnam
Davis-Putnam-Logemann-Loveland
Prenex Normal Form
Skolem Functions
Example
Herbrand Universe
Examples
Herbrand Base
Herbrand Interpretations
Formally
Example
Ground Instances of Clauses
▷ Satisfiability of Ground Clauses
Unsatisfiability
Saturation
Saturation and Herbrand Interpretations
Herbrand's Theorems
DPLL
Example
Checking for consistency
Examples
More examples
Only using rule III
First-Order Example
DPLL

Let $\mathcal{S}$ be a set of clauses and a $\mathcal{I}$ be a Herbrand interpretation for $\mathcal{S}$.

- Let C' be a ground instance of a clause C in $\mathcal{S}$. We say that $\mathcal{I}$ satisfies C', denoted by $\mathcal{I} \models C'$, if $C' \cap \mathcal{I} \neq \emptyset$ (note that clauses are regarded as set of literals).
- We say that $\mathcal{I}$ satisfies C in $\mathcal{S}$ if, and only if, for all ground clauses C' of C, we have that $\mathcal{I} \models C'$.

Example: $\mathcal{S} = \{\neg P(x) \vee Q(f(x))\}$

$\mathcal{I} = \{P(a), \neg Q(a), P(f(a)), Q(f(a)), \neg Q(f(f(a))), \ldots\}$

$\mathcal{I} \models \neg P(a) \vee Q(f(a))$
$\mathcal{I} \not\models \neg P(f(a)) \vee Q(f(f(a)))$

Unsatisfiability

Logic
The Early Days
▷ DPLL
Davis-Putnam
Davis-Putnam-Logemann-Loveland
Prenex Normal Form
Skolem Functions
Example
Herbrand Universe
Examples
Herbrand Base
Herbrand Interpretations
Formally
Example
Ground Instances of Clauses
Satisfiability of Ground Clauses
▷ Unsatisfiability
Saturation
Saturation and Herbrand Interpretations
Herbrand's Theorems
DPLL
Example
Checking for consistency
Examples
More examples
Only using rule III
First-Order Example
DPLL

Let $\mathcal{S}$ be a set of clauses. $\mathcal{S}$ is unsatisfiable if and only if for every Herbrand interpretation $\mathcal{I}$ for $\mathcal{S}$ there is a ground clause C' of C in $\mathcal{S}$ such that $\mathcal{I} \not\models C'$.

Example: $\{\neg P(x), P(a)\}$

There are only two possible Herbrand interpretations:

- $\mathcal{I}_1 = \{P(a)\}$: does not satisfy the first clause;
- $\mathcal{I}_2 = \{\neg P(a)\}$: does not satisfy the second clause.

Saturation

Logic
The Early Days
▷ DPLL
Davis-Putnam
Davis-Putnam-Logemann-Loveland
Prenex Normal Form
Skolem Functions
Example
Herbrand Universe
Examples
Herbrand Base
Herbrand Interpretations
Formally
Example
Ground Instances of Clauses
Satisfiability of Ground Clauses
Unsatisfiability
▷ Saturation
Saturation and Herbrand Interpretations
Herbrand's Theorems
DPLL
Example
Checking for consistency
Examples
More examples
Only using rule III
First-Order Example
DPLL

Let $\mathcal{S}$ be a set of clauses. Let $\mathcal{T}$ be a set of terms. By $\mathcal{T}(\mathcal{S})$, we denote the saturation of $\mathcal{S}$ over $\mathcal{T}$, which is the set of all ground clauses obtainable from $\mathcal{S}$ by (uniformly) replacing variables by members of $\mathcal{T}$.

Saturation

Logic
The Early Days
▷ DPLL
Davis-Putnam
Davis-Putnam-Logemann-Loveland
Prenex Normal Form
Skolem Functions
Example
Herbrand Universe
Examples
Herbrand Base
Herbrand Interpretations
Formally
Example
Ground Instances of Clauses
Satisfiability of Ground Clauses
Unsatisfiability
▷ Saturation
Saturation and Herbrand Interpretations
Herbrand's Theorems
DPLL
Example
Checking for consistency
Examples
More examples
Only using rule III
First-Order Example
DPLL

Let $\mathcal{S}$ be a set of clauses. Let $\mathcal{T}$ be a set of terms. By $\mathcal{T}(\mathcal{S})$, we denote the saturation of $\mathcal{S}$ over $\mathcal{T}$, which is the set of all ground clauses obtainable from $\mathcal{S}$ by (uniformly) replacing variables by members of $\mathcal{T}$.

Example: $\mathcal{S} = \{\neg P(x) \vee Q(x), P(y), \neg Q(f^2(x,y))\}$

$$\mathcal{H} = \{a, f^2(a,a), f^2(a, f^2(a,a)), f^2(f^2(a,a),a), f^2(f^2(a,a), f^2(a,a)), \ldots\}$$

$$\begin{aligned}\mathcal{H}(\mathcal{S}) = \{ \quad & \neg P(a) \vee Q(a), P(a), \neg Q(f^2(a,a)),\\ & \neg P(a) \vee Q(a), P(f^2(a,a)), \neg Q(f^2(a, f^2(a,a))),\\ & \neg P(f^2(a,a)) \vee Q(f^2(a,a)), P(a), \neg Q(f^2(f^2(a,a),a)),\\ & \neg P(f^2(a,a)) \vee Q(f^2(a,a)), P(f^2(a,a)), \neg Q(f^2(f^2(a,a), f^2(a,a))),\\ & \ldots\}\end{aligned}$$

Logic
The Early Days
▷ DPLL
Davis-Putnam
Davis-Putnam-Logemann-Loveland
Prenex Normal Form
Skolem Functions
Example
Herbrand Universe
Examples
Herbrand Base
Herbrand Interpretations
Formally
Example
Ground Instances of Clauses
Satisfiability of Ground Clauses
Unsatisfiability
▷ Saturation
Saturation and Herbrand Interpretations
Herbrand's Theorems
DPLL
Example
Checking for consistency
Examples
More examples
Only using rule III
First-Order Example
DPLL

Let $\mathcal{S}$ be a set of clauses. Let $\mathcal{T}$ be a set of terms. By $\mathcal{T}(\mathcal{S})$, we denote the saturation of $\mathcal{S}$ over $\mathcal{T}$, which is the set of all ground clauses obtainable from $\mathcal{S}$ by (uniformly) replacing variables by members of $\mathcal{T}$.

Example: $\mathcal{S} = \{\neg P(x) \vee Q(x), P(y), \neg Q(f^2(x, y))\}$

$$\mathcal{H} = \{a, f^2(a,a), f^2(a, f^2(a,a)), f^2(f^2(a,a), a), f^2(f^2(a,a), f^2(a,a)), \ldots\}$$

$$\begin{aligned} \mathcal{H}(\mathcal{S}) = \{ \quad & \neg P(a) \vee Q(a), P(a), \neg Q(f^2(a,a)), \\ & \neg P(a) \vee Q(a), P(f^2(a,a)), \neg Q(f^2(a, f^2(a,a))), \\ & \neg P(f^2(a,a)) \vee Q(f^2(a,a)), P(a), \neg Q(f^2(f^2(a,a), a)), \\ & \neg P(f^2(a,a)) \vee Q(f^2(a,a)), P(f^2(a,a)), \neg Q(f^2(f^2(a,a), f^2(a,a))), \\ & \ldots\} \end{aligned}$$

Logic
The Early Days
▷ DPLL
Davis-Putnam
Davis-Putnam-Logemann-Loveland
Prenex Normal Form
Skolem Functions
Example
Herbrand Universe
Examples
Herbrand Base
Herbrand Interpretations
Formally
Example
Ground Instances of Clauses
Satisfiability of Ground Clauses
Unsatisfiability
Saturation
▷ Saturation and Herbrand Interpretations
Herbrand's Theorems
DPLL
Example
Checking for consistency
Examples
More examples
Only using rule III
First-Order Example
DPLL

- A set of ground literals which does not include a complementary pair is called a *model* (or interpretation).
- If $\mathcal{I}$ is a model and $\mathcal{S}$ is a set of ground clauses, then $\mathcal{I}$ is a model of $\mathcal{S}$ if, for all clauses C in $\mathcal{S}$, C contains a member of $\mathcal{I}$.
- In general, if $\mathcal{S}$ is any set of clauses and $\mathcal{H}$ is the Herbrand universe of $\mathcal{S}$, then $\mathcal{I}$ is a model of $\mathcal{S}$ just in case $\mathcal{I}$ is a model of $\mathcal{H}(\mathcal{S})$.

Logic
The Early Days
▷ DPLL
Davis-Putnam
Davis-Putnam-Logemann-Loveland
Prenex Normal Form
Skolem Functions
Example
Herbrand Universe
Examples
Herbrand Base
Herbrand Interpretations
Formally
Example
Ground Instances of Clauses
Satisfiability of Ground Clauses
Unsatisfiability
Saturation
Saturation and Herbrand Interpretations
▷ Herbrand's Theorems
DPLL
Example
Checking for consistency
Examples
More examples
Only using rule III
First-Order Example

If $\mathcal{S}$ is any finite set of clauses and $\mathcal{H}$ is its Herbrand Universe, then $\mathcal{S}$ is unsatisfiable if and only if some finite subset of $\mathcal{H}(\mathcal{S})$ is unsatisfiable.

Note that this theorem suggests the following refutation procedure:

1. Select $\mathcal{P}_0, \mathcal{P}_1, \mathcal{P}_2, \ldots$ as finite subsets of the Herbrand universe $\mathcal{H}$ of $\mathcal{S}$, such that $\mathcal{P}_i \subseteq \mathcal{P}_{i+1}$, for all $i \geq 0$ and such that $\bigcup_{i=0}^{\infty} \mathcal{P}_i = \mathcal{H}$.
2. Check $\mathcal{P}_0(\mathcal{S}), \mathcal{P}_1(\mathcal{S}), \mathcal{P}_2(\mathcal{S}), \ldots$ for satisfiability.
3. By Herbrand's theorem, for some i, $\mathcal{P}_i(\mathcal{S})$ is unsatisfiable.

Level saturation procedures take $\mathcal{H}_0, \mathcal{H}_1, \mathcal{H}_2, \ldots$ to be the finite sets over which the set of clauses is saturated, where each $\mathcal{H}_{i+1}$ contains all the terms in $\mathcal{H}_i$ or whose arguments are in $\mathcal{H}_i$.
Finding *proof sets*, i.e. finite unsatisfiable subsets which are minimal (in the sense that every proper subset is satisfiable) requires guidance (heuristics).

Logic
The Early Days
▷ DPLL
Davis-Putnam
Davis-Putnam-Logemann-Loveland
Prenex Normal Form
Skolem Functions
Example
Herbrand Universe
Examples
Herbrand Base
Herbrand Interpretations
Formally
Example
Ground Instances of Clauses
Satisfiability of Ground Clauses
Unsatisfiability
Saturation
Saturation and Herbrand Interpretations
Herbrand's Theorems
▷ DPLL
Example
Checking for consistency
Examples
More examples
Only using rule III
First-Order Example
DPLL

In order to prove φ

- Take the Skolemised prenex normal form of φ, where the matrix is in Conjunctive Normal Form, obtaining φ'
- **Repeat**
  Generate the Herbrand sets by level and check for unsatisfiability
  until a contradiction has been found.

Example

Logic
The Early Days
▷ DPLL
Davis-Putnam
Davis-Putnam-Logemann-Loveland
Prenex Normal Form
Skolem Functions
Example
Herbrand Universe
Examples
Herbrand Base
Herbrand Interpretations
Formally
Example
Ground Instances of Clauses
Satisfiability of Ground Clauses
Unsatisfiability
Saturation
Saturation and Herbrand Interpretations
Herbrand's Theorems
DPLL
▷ Example
Checking for consistency
Examples
More examples
Only using rule III
First-Order Example
DPLL

$$\forall x_1 \exists x_2 \forall x_3 R(x_1, x_2, x_3)$$

$$\forall x_1 \forall x_3 R(x_1, f(x_1), x_3)$$

$$R(a, f(a), a)$$

$$R(a, f(a), f(a))$$

$$R(f(a), f(f(a)), a)$$

$$R(f(a), f(f(a)), f(a))$$

$$\vdots$$

Checking for consistency

Logic
The Early Days
▷ DPLL
Davis-Putnam
Davis-Putnam-Logemann-Loveland
Prenex Normal Form
Skolem Functions
Example
Herbrand Universe
Examples
Herbrand Base
Herbrand Interpretations
Formally
Example
Ground Instances of Clauses
Satisfiability of Ground Clauses
Unsatisfiability
Saturation
Saturation and Herbrand Interpretations
Herbrand's Theorems
DPLL
Example
▷ Checking for consistency
Examples
More examples
Only using rule III
First-Order Example
DPLL

1. Unit resolution (which is there called "Rule for the Elimination of One-Literal Clauses"):
   (a) if p, $\neg p$ are in $\mathcal{S}$, $\mathcal{S} = \{\bot\}$.
   (b) if p in $\mathcal{S}$, then
   $\mathcal{S} = \mathcal{S} \setminus \{C \mid p \in C\} \setminus \{D \mid \neg p \in D\} \cup \{D \mid D \cup \{\neg p\} \in \mathcal{S}\}$.
   (c) $\neg p$ in $\mathcal{S}$, then
   $\mathcal{S} = \mathcal{S} \setminus \{C \mid \neg p \in C\} \setminus \{D \mid p \in D\} \cup \{D \mid D \cup \{p\} \in \mathcal{S}\}$.
2. Pure Literal Elimination (which is there called "Affirmative-Negative Rule"): if l occurs only positively or negatively, then $\mathcal{S} = \mathcal{S} \setminus \{D \mid l \in D\}$.
3. Rule for Eliminating Atomic Formulas: if we can obtain (by factoring) $\mathcal{S} = \{(\varphi \vee p), (\psi \vee \neg p), R\}$, where R is free of p, then $\mathcal{S} = \{(\varphi \vee \psi), R\}$

Examples

Logic
The Early Days
▷ DPLL
Davis-Putnam
Davis-Putnam-Logemann-Loveland
Prenex Normal Form
Skolem Functions
Example
Herbrand Universe
Examples
Herbrand Base
Herbrand Interpretations
Formally
Example
Ground Instances of Clauses
Satisfiability of Ground Clauses
Unsatisfiability
Saturation
Saturation and Herbrand Interpretations
Herbrand's Theorems
DPLL
Example
Checking for consistency
▷ Examples
More examples
Only using rule III
First-Order Example
DPLL

$$p \vee q \vee \neg r, p \vee \neg q, \neg p, r$$

$p \vee q \vee \neg r, p \vee \neg q, \neg p, r$

$p \vee q, p \vee \neg q, \neg p$

$q, \neg q$

$\bot$

$$p \vee q, \neg q, \neg p \vee q \vee \neg r$$

$p \vee q, \neg q, \neg p \vee q \vee \neg r$

$p \vee q, \neg q$

$p \vee q, \neg q$

p

$\emptyset$

More examples

Logic
The Early Days
▷ DPLL
Davis-Putnam
Davis-Putnam-Logemann-Loveland
Prenex Normal Form
Skolem Functions
Example
Herbrand Universe
Examples
Herbrand Base
Herbrand Interpretations
Formally
Example
Ground Instances of Clauses
Satisfiability of Ground Clauses
Unsatisfiability
Saturation
Saturation and Herbrand Interpretations
Herbrand's Theorems
DPLL
Example
Checking for consistency
Examples
▷ More examples
Only using rule III
First-Order Example
DPLL

$$p \vee \neg q, \neg p \vee q, q \vee \neg r, \neg q \vee \neg r$$

$p \vee \neg q, \neg p \vee q, q \vee {\color{red}\neg r}, \neg q \vee {\color{red}\neg r}$

$p \vee \neg q, \neg p \vee q$

$q \vee \neg q$

$\emptyset$

Only using rule III

Logic

The Early Days

▷ DPLL

Davis-Putnam
Davis-Putnam-Logemann-Loveland
Prenex Normal Form
Skolem Functions
Example
Herbrand Universe
Examples
Herbrand Base
Herbrand Interpretations
Formally
Example
Ground Instances of Clauses
Satisfiability of Ground Clauses
Unsatisfiability
Saturation
Saturation and Herbrand Interpretations
Herbrand's Theorems
DPLL
Example
Checking for consistency
Examples
More examples
▷ Only using rule III
First-Order Example
DPLL

$$p \lor r, p \lor \neg s, \neg p \lor s, \neg p \lor \neg r, s \lor \neg r, \neg s \lor r$$

$$(r \land \neg s) \lor p, (\neg r \land s) \lor \neg p, s \lor \neg r, \neg s \lor r$$

$$(r \land \neg s) \lor (\neg r \land s), s \lor \neg r, \neg s \lor r$$

$$s \lor r, \neg s \lor \neg r, s \lor \neg r, \neg s \lor r$$

$$(s \land \neg s) \lor r, (s \land \neg s) \lor \neg r$$

$$(s \land \neg s)$$

$$s, \neg s = \bot$$

First-Order Example

Logic
The Early Days
▷ DPLL
Davis-Putnam
Davis-Putnam-Logemann-Loveland
Prenex Normal Form
Skolem Functions
Example
Herbrand Universe
Examples
Herbrand Base
Herbrand Interpretations
Formally
Example
Ground Instances of Clauses
Satisfiability of Ground Clauses
Unsatisfiability
Saturation
Saturation and Herbrand Interpretations
Herbrand's Theorems
DPLL
Example
Checking for consistency
Examples
More examples
Only using rule III
▷ First-Order Example

$$\exists x \exists y \forall z ((F(x,y) \rightarrow (F(y,z) \wedge F(z,z)))$$
$$\wedge ((F(x,y) \wedge G(x,y)) \rightarrow (G(x,z) \wedge G(z,z))))$$

$$F(x,y),$$
$$\neg F(y, f(x,y)) \vee \neg F(f(x,y), f(x,y)) \vee G(x,y),$$
$$\neg F(y, f(x,y)) \vee \neg F(f(x,y), f(x,y)) \vee \neg G(x, f(x,y)) \vee \neg G(f(x,y), f(x,y))$$

It took 25 lines of instantiation using the enumeration procedure described before, where nesting of functional symbols goes up to 25. Then, unit resolution can be repeatedly applied and a contradiction is found.

DPLL

Logic
The Early Days
▷ DPLL
Davis-Putnam
Davis-Putnam-Logemann-Loveland
Prenex Normal Form
Skolem Functions
Example
Herbrand Universe
Examples
Herbrand Base
Herbrand Interpretations
Formally
Example
Ground Instances of Clauses
Satisfiability of Ground Clauses
Unsatisfiability
Saturation
Saturation and Herbrand Interpretations
Herbrand's Theorems
DPLL
Example
Checking for consistency
Examples
More examples
Only using rule III
First-Order Example
▷ DPLL

- Rule for Eliminating Atomic Formulas: if we can obtain (by factoring) $\mathcal{S} = \{(\varphi \vee p), (\psi \vee \neg p), R\}$, where R is free of p, then $\mathcal{S} = \{(\varphi \vee \psi), R\}$ (1960).
- Splitting (Rule for Eliminating Atomic Formulas)*: if we can obtain (by factoring) $\mathcal{S} = \{(\varphi \vee p), (\psi \vee \neg p), R\}$, where R is free of p, then $\mathcal{S} = \{(\varphi \wedge R), (\psi \wedge R)\}$ (1962).
- One of the formulae is put on the tape and processed; if it is not satisfiable, then the other formula is tested for satisfiability.
- The formula

$$\exists x \exists y \forall z ((F(x,y) \rightarrow (F(y,z) \wedge F(z,z))) \wedge ((F(x,y) \wedge G(x,y)) \rightarrow (G(x,z) \wedge G(z,z)))$$

was proved under two minutes after 60 instantiation lines.

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
... and then came light!
The Resolution Principle
Resolution, at last
Unification
Another Example
The upshot
Literal Unification
Occurs Check
Most General Unifier
Unification - Example
Propositional Resolution
The algorithm
Tautology Elimination
First-Order Resolution
Resolution Method
Soundness and Completeness
Subsumption
Historical References

Resolution

John Alan Robinson

- "Theories of Meaning Implicit in the British Empiricists Locke, Berkeley and Hume" (Cambridge, 1952).
- "Causality, Probability and Testimony" (PhD, University of Oregon, University of Princeton, 1956).
- DuPont, operations research.

- In 1960, while holding a postdoc at the University of Pittsburgh, he accepts a tenure from Rice University, but decides to take a Summer research position at the Applied Mathematics Division of the Argonne National Laboratory (Chicago).
- His appointment at Argonne was to implement the Davis-Putnam procedure on an IBM 704.
- This work is described in a report (1961) which is in revised form in a paper from the Journal of the ACM [Rob63].

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
▷ Argonne
1962-1963
... and then came light!
The Resolution Principle
Resolution, at last
Unification
Another Example
The upshot
Literal Unification
Occurs Check
Most General Unifier
Unification - Example
Propositional Resolution
The algorithm
Tautology Elimination
First-Order Resolution
Resolution Method
Soundness and Completeness
Subsumption
Historical References

In 1960, he implements the DP procedure in FORTRAN (together with George Robinson).

> I spent the summer [of 1961] happily... running this thing. Yes, it worked just like they [Davis and Putnam] said. However, the examples that one would like to try next blew it up, and we ran into combinatorial explosion of the instantiation, 'try all the instances, enumerate all the instances.' So I learnt that summer that was not exactly an elegant way to go.

In 1961, he implements Prawitz procedure in FORTRAN.

> The work at Argonne really was dominating my life, because I started looking at other ways to go in this first-order procedure stuff, and stumbled across the work of Prawitz, who was the one who reached some form of unification, calculating instances rather than trying them all.

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
▷ 1962-1963
... and then came light!
The Resolution Principle
Resolution, at last
Unification
Another Example
The upshot
Literal Unification
Occurs Check
Most General Unifier
Unification - Example
Propositional Resolution
The algorithm
Tautology Elimination
First-Order Resolution
Resolution Method
Soundness and Completeness
Subsumption
Historical References

William F. Miller, the boss at the Applied Mathematics Division in Argonne, sets up a team for the theorem-proving research project:

- John Alan Robinson.
- George Robinson, who earlier had helped with the implementation of the DP procedure.
- Larry Wos, a mathematician at the Lab.
- During those two summer projects, Robinson rediscovered unification and the resolution rule, which was submitted in 1963, sitting on the desk of some reviewer for a year, and finally published in 1965.

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
▷ … and then came light!
The Resolution Principle
Resolution, at last
Unification
Another Example
The upshot
Literal Unification
Occurs Check
Most General Unifier
Unification - Example
Propositional Resolution
The algorithm
Tautology Elimination
First-Order Resolution
Resolution Method
Soundness and Completeness
Subsumption
Historical References

- On the unification procedure, Robinson says that he was "absolutely inspired by Prawitz". He uses a version similar to that of Herbrand's unification algorithm in his paper. Herbrand's procedure is definitional rather than the one devised by Prawitz, which is algorithmic.
- Prawitz and Robinson were not aware of this part of Herbrand's paper (although both cite the paper).
- The resolution rule had also appeared before:
  - it was first discovered in 1937, by Blake [Bla37];
  - then rediscovered by Quine (there called "consensus rule"), in 1955 [vOQ55];
  - and it was proposed for theorem proving in 1962 at Harvard University by Dunham and North [DN63].

The Resolution Principle

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
... and then came light!
▷ The Resolution Principle
Resolution, at last
Unification
Another Example
The upshot
Literal Unification
Occurs Check
Most General Unifier
Unification - Example
Propositional Resolution
The algorithm
Tautology Elimination
First-Order Resolution
Resolution Method
Soundness and Completeness
Subsumption
Historical References

> Traditionally, a single step in deduction has been required, for pragmatic and psychological reasons, to be simple enough, broadly speaking, to be apprehended as correct by a human being in a single intellectual act. No doubt this custom originated in the desire that each single step of a deduction should be indubitable, even though the deductions as whole may consist of a long chain of such steps.

Resolution was far more complex, as it "condones single inferences which are often beyond the ability of human to grasp (other than discursively)." Resolution is *machine-oriented*: it was shown to be complete and it was designed for efficiency. Thus, although indubitable, resolution wasn't meant for human reading.

Resolution, at last

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
... and then came light!
The Resolution Principle
▷ Resolution, at last
Unification
Another Example
The upshot
Literal Unification
Occurs Check
Most General Unifier
Unification - Example
Propositional Resolution
The algorithm
Tautology Elimination
First-Order Resolution
Resolution Method
Soundness and Completeness
Subsumption
Historical References

$$[\text{CUT}] \quad \frac{\begin{array}{ccc} \varphi & \Rightarrow & \psi \\ \psi & \Rightarrow & \chi \end{array}}{\begin{array}{ccc} \varphi & \Rightarrow & \chi \end{array}}$$

Resolution, at last

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
... and then came light!
The Resolution Principle
▷ Resolution, at last
Unification
Another Example
The upshot
Literal Unification
Occurs Check
Most General Unifier
Unification - Example
Propositional Resolution
The algorithm
Tautology Elimination
First-Order Resolution
Resolution Method
Soundness and Completeness
Subsumption
Historical References

$$[\text{CUT}]\quad \frac{\begin{array}{ccc}\varphi & \Rightarrow & \psi\\ \psi & \Rightarrow & \chi\end{array}}{\begin{array}{ccc}\varphi & \Rightarrow & \chi\end{array}}$$

$$[\text{RESOLUTION}]\quad \frac{\begin{array}{ccc}(\varphi & \vee & \psi)\\ (\neg\psi & \vee & \chi)\end{array}}{\begin{array}{ccc}(\varphi & \vee & \chi)\end{array}}$$

Resolution, at last

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
... and then came light!
The Resolution Principle
▷ Resolution, at last
Unification
Another Example
The upshot
Literal Unification
Occurs Check
Most General Unifier
Unification - Example
Propositional Resolution
The algorithm
Tautology Elimination
First-Order Resolution
Resolution Method
Soundness and Completeness
Subsumption
Historical References

$$[\text{CUT}] \quad \frac{\begin{array}{ccc}\varphi & \Rightarrow & \psi \\ \psi & \Rightarrow & \chi\end{array}}{\begin{array}{ccc}\varphi & \Rightarrow & \chi\end{array}}$$

$$[\text{RESOLUTION}] \quad \frac{\begin{array}{ccc}(\varphi & \vee & \psi) \\ (\neg\psi & \vee & \chi)\end{array}}{\begin{array}{ccc}(\varphi & \vee & \chi)\sigma\end{array}}$$

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
... and then came light!
The Resolution Principle
Resolution, at last
▷ Unification
Another Example
The upshot
Literal Unification
Occurs Check
Most General Unifier
Unification - Example
Propositional Resolution
The algorithm
Tautology Elimination
First-Order Resolution
Resolution Method
Soundness and Completeness
Subsumption
Historical References

$$[\text{RESOLUTION}] \quad \frac{\begin{array}{ccc}(\varphi & \vee & \psi) \\ (\neg\psi' & \vee & \chi)\end{array}}{(\varphi \quad \vee \quad \chi)\sigma}$$

where $\sigma = \text{unifier}(\psi, \psi')$

Example:

$$\begin{array}{lll} 1. & \neg Rich(x) \vee Happy(x) & \\ 2. & Rich(Bill) & \\ 3. & Happy(Bill) & [RES, 1, 2, Bill \backslash x] \end{array}$$

Another Example

1. $F(x, y)$
2. $\neg F(z, f_1(w, z)) \vee \neg F(f_1(w, z), f_1(w, z)) \vee G(w, z)$
3. $\neg F(t, f_1(u, t)) \vee \neg F(f_1(u, t), f_1(u, t)) \vee \neg G(u, f_1(u, t)) \vee \neg G(f_1(u, t), f_1(u, t))$
4. $\neg F(f_1(w, z), f_1(w, z)) \vee G(w, z)$
   $[Res, 1, 2, z \backslash x, f_1(w, z) \backslash y]$
5. $G(w, z)$
   $[Res, 1, 5, z \backslash x, f_1(w, z) \backslash y]$
6. $\neg F(f_1(w, z), f_1(w, z)) \vee \neg G(w, f_1(w, z)) \vee \neg G(f_1(x, y), f_1(x, y))$
   $[Res, 1, 3, z \backslash x, f_1(w, z) \backslash y]$
7. $\neg G(t, f_1(u, t)) \vee \neg G(f_1(u, t), f_1(u, t))$
   $[Res, 1, 6, t \backslash x, f_1(u, t) \backslash y]$
8. $\neg G(t, f_1(u, t)) \vee \neg G(f_1(u, t), f_1(u, t))$
   $[Res, 1, 7, t \backslash x, f_1(u, t) \backslash y]$
9. $\neg G(f_1(u, t), f_1(u, t))$
   $[Res, 8, 5, t \backslash w, f_1(u, t) \backslash z]$
10. $\square$
   $[Res, 9, 5, f_1(u, t) \backslash w, f_1(u, t) \backslash z]$

The upshot

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
... and then came light!
The Resolution Principle
Resolution, at last
Unification
Another Example
▷ The upshot
Literal Unification
Occurs Check
Most General Unifier
Unification - Example
Propositional Resolution
The algorithm
Tautology Elimination
First-Order Resolution
Resolution Method
Soundness and Completeness
Subsumption
Historical References

- Argonne was then the world champion in theorem-proving!
- Refinements were mostly introduced by Robinson and Wos.
- From 1967 till 1970, Robinson publishes around 90 papers on theorem-proving.

Literal Unification

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
... and then came light!
The Resolution Principle
Resolution, at last
Unification
Another Example
The upshot
▷ Literal Unification
Occurs Check
Most General Unifier
Unification - Example
Propositional Resolution
The algorithm
Tautology Elimination
First-Order Resolution
Resolution Method
Soundness and Completeness
Subsumption
Historical References

- Let φ be a literal. We denote by φ^i the symbol at the $i-th$ position of φ. If $i > |\varphi|$, then φ^i is undefined.
- Let φ and ψ be literals with $\varphi \neq \psi$. Let i be the least number such that both φ^i and ψ^i are defined and such that $\varphi^i \neq \psi^i$. The pair (φ^i, ψ^i) is called the *disagreement pair* of φ and ψ.

Example:

$$\begin{array}{rcl} \varphi & = & P(g_1(c), f_1(a, g_1(x), g_2(a, g_1(b)))) \\ \psi & = & P(g_1(c), f_1(a, g_1(x), g_2(f_2(x, y), z))) \end{array}$$

Disagreement pair: $(a, f_2(x, y))$

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
... and then came light!
The Resolution Principle
Resolution, at last
Unification
Another Example
The upshot
Literal Unification
▷ Occurs Check
Most General Unifier
Unification - Example
Propositional Resolution
The algorithm
Tautology Elimination
First-Order Resolution
Resolution Method
Soundness and Completeness
Subsumption
Historical References

- A substitution $[t\backslash x]$ satisfies the occurs check iff x does not occur in t.
- A disagreement pair $D = (x, t)$ satisfies the occurs check if the variable x does not occur in t.

Examples:

$$\begin{array}{rcl} \varphi & = & P(a, x, f(g(y))) \\ \psi & = & P(z, h(z, w), f(w)) \end{array}$$

$D = (a, z)$ satisfies the occurs check.

$$\begin{array}{rcl} \varphi & = & P(z, h(z, w), f(w)) \\ \psi & = & P(f(z), h(z, w), f(w)) \end{array}$$

$D = (z, f(z))$ does not satisfy the occurs check.

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
... and then came light!
The Resolution Principle
Resolution, at last
Unification
Another Example
The upshot
Literal Unification
Occurs Check
▷ Most General Unifier
Unification - Example
Propositional Resolution
The algorithm
Tautology Elimination
First-Order Resolution
Resolution Method
Soundness and Completeness
Subsumption
Historical References

Most General Unifier

Input: a pair of expressions χ_1 and χ_2.

Output: m.u.g. σ, if χ_1 and χ_2 are unifiable; "no", otherwise.

$\sigma \leftarrow \emptyset$; $W \leftarrow \{\chi_1, \chi_2\}$; $D \leftarrow$ is the disagreement pair of W;
while $|W| > 1$ and D satisfies the occurs check:
  select a variable x and a term t in D (x does not occur in t)
  $\sigma \leftarrow \sigma \circ \{t \backslash x\}$
  $W = \{\chi_1, \chi_2\} \leftarrow \{\chi_1[t \backslash x], \chi_2[t \backslash x]\}$
  $D \leftarrow$ disagreement pair of W
end_while
if $|W| = 1$
then return σ
else return "no"

Unification - Example

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
... and then came light!
The Resolution Principle
Resolution, at last
Unification
Another Example
The upshot
Literal Unification
Occurs Check
Most General Unifier
▷ Unification - Example
Propositional Resolution
The algorithm
Tautology Elimination
First-Order Resolution
Resolution Method
Soundness and Completeness
Subsumption
Historical References

$\chi_1 = P(a, x, f(y))$ e $\chi_2 = P(u, v, w)$

Unification - Example

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
... and then came light!
The Resolution Principle
Resolution, at last
Unification
Another Example
The upshot
Literal Unification
Occurs Check
Most General Unifier
▷ Unification - Example
Propositional Resolution
The algorithm
Tautology Elimination
First-Order Resolution
Resolution Method
Soundness and Completeness
Subsumption
Historical References

$\chi_1 = P(a, x, f(y))$ e $\chi_2 = P(u, v, w)$

$\sigma \leftarrow \emptyset$; $W \leftarrow \{P(a, x, f(y)), P(u, v, w)\}$; $D \leftarrow (a, u)$

Unification - Example

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
... and then came light!
The Resolution Principle
Resolution, at last
Unification
Another Example
The upshot
Literal Unification
Occurs Check
Most General Unifier
▷ Unification - Example
Propositional Resolution
The algorithm
Tautology Elimination
First-Order Resolution
Resolution Method
Soundness and Completeness
Subsumption
Historical References

$\chi_1 = P(a, x, f(y))$ e $\chi_2 = P(u, v, w)$

$\sigma \leftarrow \emptyset$; $W \leftarrow \{P(a, x, f(y)), P(u, v, w)\}$; $D \leftarrow (a, u)$

$|W| > 1$ and D satisfies the occurs check

Unification - Example

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
... and then came light!
The Resolution Principle
Resolution, at last
Unification
Another Example
The upshot
Literal Unification
Occurs Check
Most General Unifier
▷ Unification - Example
Propositional Resolution
The algorithm
Tautology Elimination
First-Order Resolution
Resolution Method
Soundness and Completeness
Subsumption
Historical References

$\chi_1 = P(a, x, f(y))$ e $\chi_2 = P(u, v, w)$

$\sigma \leftarrow \emptyset$; $W \leftarrow \{P(a, x, f(y)), P(u, v, w)\}$; $D \leftarrow (a, u)$

$|W| > 1$ and D satisfies the occurs check

$\sigma \leftarrow \emptyset \circ \{a \backslash u\} = \{a \backslash u\}$

Unification - Example

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
... and then came light!
The Resolution Principle
Resolution, at last
Unification
Another Example
The upshot
Literal Unification
Occurs Check
Most General Unifier
▷ Unification - Example
Propositional Resolution
The algorithm
Tautology Elimination
First-Order Resolution
Resolution Method
Soundness and Completeness
Subsumption
Historical References

$\chi_1 = P(a, x, f(y))$ e $\chi_2 = P(u, v, w)$

$\sigma \leftarrow \emptyset$; $W \leftarrow \{P(a, x, f(y)), P(u, v, w)\}$; $D \leftarrow (a, u)$

$|W| > 1$ and D satisfies the occurs check

$\sigma \leftarrow \emptyset \circ \{a \backslash u\} = \{a \backslash u\}$

$W \leftarrow \{P(a, x, f(y))[a \backslash u], P(u, v, w)[a \backslash u]\}$

Unification - Example

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
... and then came light!
The Resolution Principle
Resolution, at last
Unification
Another Example
The upshot
Literal Unification
Occurs Check
Most General Unifier
▷ Unification - Example
Propositional Resolution
The algorithm
Tautology Elimination
First-Order Resolution
Resolution Method
Soundness and Completeness
Subsumption
Historical References

$\chi_1 = P(a, x, f(y))$ e $\chi_2 = P(u, v, w)$

$\sigma \leftarrow \emptyset$; $W \leftarrow \{P(a, x, f(y)), P(u, v, w)\}$; $D \leftarrow (a, u)$

- $|W| > 1$ and D satisfies the occurs check
- $\sigma \leftarrow \emptyset \circ \{a \backslash u\} = \{a \backslash u\}$
- $W \leftarrow \{P(a, x, f(y))[a \backslash u], P(u, v, w)[a \backslash u]\}$
- $W = \{P(a, x, f(y)), P(a, v, w)\}$, $D \leftarrow (x, v)$

Unification - Example

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
... and then came light!
The Resolution Principle
Resolution, at last
Unification
Another Example
The upshot
Literal Unification
Occurs Check
Most General Unifier
▷ Unification - Example
Propositional Resolution
The algorithm
Tautology Elimination
First-Order Resolution
Resolution Method
Soundness and Completeness
Subsumption
Historical References

$\chi_1 = P(a, x, f(y))$ e $\chi_2 = P(u, v, w)$

$\sigma \leftarrow \emptyset$; $W \leftarrow \{P(a, x, f(y)), P(u, v, w)\}$; $D \leftarrow (a, u)$

$|W| > 1$ and D satisfies the occurs check

$\sigma \leftarrow \emptyset \circ \{a \backslash u\} = \{a \backslash u\}$

$W \leftarrow \{P(a, x, f(y))[a \backslash u], P(u, v, w)[a \backslash u]\}$

$W = \{P(a, x, f(y)), P(a, v, w)\}$, $D \leftarrow (x, v)$

$|W| > 1$ and D satisfies the occurs check

Unification - Example

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
... and then came light!
The Resolution Principle
Resolution, at last
Unification
Another Example
The upshot
Literal Unification
Occurs Check
Most General Unifier
▷ Unification - Example
Propositional Resolution
The algorithm
Tautology Elimination
First-Order Resolution
Resolution Method
Soundness and Completeness
Subsumption
Historical References

$\chi_1 = P(a, x, f(y))$ e $\chi_2 = P(u, v, w)$

$\sigma \leftarrow \emptyset$; $W \leftarrow \{P(a, x, f(y)), P(u, v, w)\}$; $D \leftarrow (a, u)$

$|W| > 1$ and D satisfies the occurs check

$\sigma \leftarrow \emptyset \circ \{a \backslash u\} = \{a \backslash u\}$

$W \leftarrow \{P(a, x, f(y))[a \backslash u], P(u, v, w)[a \backslash u]\}$

$W = \{P(a, x, f(y)), P(a, v, w)\}$, $D \leftarrow (x, v)$

$|W| > 1$ and D satisfies the occurs check

$\sigma \leftarrow \{a \backslash u\} \circ \{v \backslash x\} = \{a \backslash u, v \backslash x\}$

Unification - Example

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
... and then came light!
The Resolution Principle
Resolution, at last
Unification
Another Example
The upshot
Literal Unification
Occurs Check
Most General Unifier
▷ Unification - Example
Propositional Resolution
The algorithm
Tautology Elimination
First-Order Resolution
Resolution Method
Soundness and Completeness
Subsumption
Historical References

$\chi_1 = P(a, x, f(y))$ e $\chi_2 = P(u, v, w)$

$\sigma \leftarrow \emptyset$; $W \leftarrow \{P(a, x, f(y)), P(u, v, w)\}$; $D \leftarrow (a, u)$

$|W| > 1$ and D satisfies the occurs check

$\sigma \leftarrow \emptyset \circ \{a \backslash u\} = \{a \backslash u\}$

$W \leftarrow \{P(a, x, f(y))[a \backslash u], P(u, v, w)[a \backslash u]\}$

$W = \{P(a, x, f(y)), P(a, v, w)\}$, $D \leftarrow (x, v)$

$|W| > 1$ and D satisfies the occurs check

$\sigma \leftarrow \{a \backslash u\} \circ \{v \backslash x\} = \{a \backslash u, v \backslash x\}$

$W \leftarrow \{P(a, x, f(y))[v \backslash x], P(a, v, w)[v \backslash x]\}$

Unification - Example

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
... and then came light!
The Resolution Principle
Resolution, at last
Unification
Another Example
The upshot
Literal Unification
Occurs Check
Most General Unifier
▷ Unification - Example
Propositional Resolution
The algorithm
Tautology Elimination
First-Order Resolution
Resolution Method
Soundness and Completeness
Subsumption
Historical References

$\chi_1 = P(a, x, f(y))$ e $\chi_2 = P(u, v, w)$

$\sigma \leftarrow \emptyset$; $W \leftarrow \{P(a, x, f(y)), P(u, v, w)\}$; $D \leftarrow (a, u)$

$|W| > 1$ and D satisfies the occurs check

$\sigma \leftarrow \emptyset \circ \{a \backslash u\} = \{a \backslash u\}$

$W \leftarrow \{P(a, x, f(y))[a \backslash u], P(u, v, w)[a \backslash u]\}$

$W = \{P(a, x, f(y)), P(a, v, w)\}$, $D \leftarrow (x, v)$

$|W| > 1$ and D satisfies the occurs check

$\sigma \leftarrow \{a \backslash u\} \circ \{v \backslash x\} = \{a \backslash u, v \backslash x\}$

$W \leftarrow \{P(a, x, f(y))[v \backslash x], P(a, v, w)[v \backslash x]\}$

$W = \{P(a, v, f(y)), P(a, v, w)\}$, $D \leftarrow (f(y), w)$

Unification - Example

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
... and then came light!
The Resolution Principle
Resolution, at last
Unification
Another Example
The upshot
Literal Unification
Occurs Check
Most General Unifier
▷ Unification - Example
Propositional Resolution
The algorithm
Tautology Elimination
First-Order Resolution
Resolution Method
Soundness and Completeness
Subsumption
Historical References

$\chi_1 = P(a, x, f(y))$ e $\chi_2 = P(u, v, w)$

$\sigma \leftarrow \emptyset$; $W \leftarrow \{P(a, x, f(y)), P(u, v, w)\}$; $D \leftarrow (a, u)$

$|W| > 1$ and D satisfies the occurs check

$\sigma \leftarrow \emptyset \circ \{a \backslash u\} = \{a \backslash u\}$

$W \leftarrow \{P(a, x, f(y))[a \backslash u], P(u, v, w)[a \backslash u]\}$

$W = \{P(a, x, f(y)), P(a, v, w)\}$, $D \leftarrow (x, v)$

$|W| > 1$ and D satisfies the occurs check

$\sigma \leftarrow \{a \backslash u\} \circ \{v \backslash x\} = \{a \backslash u, v \backslash x\}$

$W \leftarrow \{P(a, x, f(y))[v \backslash x], P(a, v, w)[v \backslash x]\}$

$W = \{P(a, v, f(y)), P(a, v, w)\}$, $D \leftarrow (f(y), w)$

$|W| > 1$ and D satisfies the occurs check

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
... and then came light!
The Resolution Principle
Resolution, at last
Unification
Another Example
The upshot
Literal Unification
Occurs Check
Most General Unifier
▷ Unification - Example
Propositional Resolution
The algorithm
Tautology Elimination
First-Order Resolution
Resolution Method
Soundness and Completeness
Subsumption
Historical References

Unification - Example

$\chi_1 = P(a, x, f(y))$ e $\chi_2 = P(u, v, w)$

$\sigma \leftarrow \emptyset$; $W \leftarrow \{P(a, x, f(y)), P(u, v, w)\}$; $D \leftarrow (a, u)$

- $|W| > 1$ and D satisfies the occurs check
- $\sigma \leftarrow \emptyset \circ \{a \backslash u\} = \{a \backslash u\}$
- $W \leftarrow \{P(a, x, f(y))[a \backslash u], P(u, v, w)[a \backslash u]\}$
- $W = \{P(a, x, f(y)), P(a, v, w)\}$, $D \leftarrow (x, v)$
  - $|W| > 1$ and D satisfies the occurs check
  - $\sigma \leftarrow \{a \backslash u\} \circ \{v \backslash x\} = \{a \backslash u, v \backslash x\}$
  - $W \leftarrow \{P(a, x, f(y))[v \backslash x], P(a, v, w)[v \backslash x]\}$
  - $W = \{P(a, v, f(y)), P(a, v, w)\}$, $D \leftarrow (f(y), w)$
    - $|W| > 1$ and D satisfies the occurs check
    - $\sigma \leftarrow \{a \backslash u, v \backslash x\} \circ \{f(y) \backslash w\} = \{a \backslash u, v \backslash x, f(y) \backslash w\}$

Unification - Example

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
... and then came light!
The Resolution Principle
Resolution, at last
Unification
Another Example
The upshot
Literal Unification
Occurs Check
Most General Unifier
▷ Unification - Example
Propositional Resolution
The algorithm
Tautology Elimination
First-Order Resolution
Resolution Method
Soundness and Completeness
Subsumption
Historical References

$\chi_1 = P(a, x, f(y))$ e $\chi_2 = P(u, v, w)$

$\sigma \leftarrow \emptyset$; $W \leftarrow \{P(a, x, f(y)), P(u, v, w)\}$; $D \leftarrow (a, u)$

$|W| > 1$ and D satisfies the occurs check

$\sigma \leftarrow \emptyset \circ \{a \backslash u\} = \{a \backslash u\}$

$W \leftarrow \{P(a, x, f(y))[a \backslash u], P(u, v, w)[a \backslash u]\}$

$W = \{P(a, x, f(y)), P(a, v, w)\}$, $D \leftarrow (x, v)$

$|W| > 1$ and D satisfies the occurs check

$\sigma \leftarrow \{a \backslash u\} \circ \{v \backslash x\} = \{a \backslash u, v \backslash x\}$

$W \leftarrow \{P(a, x, f(y))[v \backslash x], P(a, v, w)[v \backslash x]\}$

$W = \{P(a, v, f(y)), P(a, v, w)\}$, $D \leftarrow (f(y), w)$

$|W| > 1$ and D satisfies the occurs check

$\sigma \leftarrow \{a \backslash u, v \backslash x\} \circ \{f(y) \backslash w\} = \{a \backslash u, v \backslash x, f(y) \backslash w\}$

$W \leftarrow \{P(a, v, f(y))[f(y) \backslash w], P(a, v, w)[f(y) \backslash w]\}$

Unification - Example

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
... and then came light!
The Resolution Principle
Resolution, at last
Unification
Another Example
The upshot
Literal Unification
Occurs Check
Most General Unifier
▷ Unification - Example
Propositional Resolution
The algorithm
Tautology Elimination
First-Order Resolution
Resolution Method
Soundness and Completeness
Subsumption
Historical References

$\chi_1 = P(a, x, f(y))$ e $\chi_2 = P(u, v, w)$

$\sigma \leftarrow \emptyset$; $W \leftarrow \{P(a, x, f(y)), P(u, v, w)\}$; $D \leftarrow (a, u)$

$|W| > 1$ and D satisfies the occurs check

$\sigma \leftarrow \emptyset \circ \{a \backslash u\} = \{a \backslash u\}$

$W \leftarrow \{P(a, x, f(y))[a \backslash u], P(u, v, w)[a \backslash u]\}$

$W = \{P(a, x, f(y)), P(a, v, w)\}$, $D \leftarrow (x, v)$

$|W| > 1$ and D satisfies the occurs check

$\sigma \leftarrow \{a \backslash u\} \circ \{v \backslash x\} = \{a \backslash u, v \backslash x\}$

$W \leftarrow \{P(a, x, f(y))[v \backslash x], P(a, v, w)[v \backslash x]\}$

$W = \{P(a, v, f(y)), P(a, v, w)\}$, $D \leftarrow (f(y), w)$

$|W| > 1$ and D satisfies the occurs check

$\sigma \leftarrow \{a \backslash u, v \backslash x\} \circ \{f(y) \backslash w\} = \{a \backslash u, v \backslash x, f(y) \backslash w\}$

$W \leftarrow \{P(a, v, f(y))[f(y) \backslash w], P(a, v, w)[f(y) \backslash w]\}$

$W = \{P(a, v, f(y))\}$; D is undefined;

Unification - Example

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
... and then came light!
The Resolution Principle
Resolution, at last
Unification
Another Example
The upshot
Literal Unification
Occurs Check
Most General Unifier
▷ Unification - Example
Propositional Resolution
The algorithm
Tautology Elimination
First-Order Resolution
Resolution Method
Soundness and Completeness
Subsumption
Historical References

$\chi_1 = P(a, x, f(y))$ e $\chi_2 = P(u, v, w)$

$\sigma \leftarrow \emptyset$; $W \leftarrow \{P(a, x, f(y)), P(u, v, w)\}$; $D \leftarrow (a, u)$

$|W| > 1$ and D satisfies the occurs check

$\sigma \leftarrow \emptyset \circ \{a \backslash u\} = \{a \backslash u\}$

$W \leftarrow \{P(a, x, f(y))[a \backslash u], P(u, v, w)[a \backslash u]\}$

$W = \{P(a, x, f(y)), P(a, v, w)\}$, $D \leftarrow (x, v)$

$|W| > 1$ and D satisfies the occurs check

$\sigma \leftarrow \{a \backslash u\} \circ \{v \backslash x\} = \{a \backslash u, v \backslash x\}$

$W \leftarrow \{P(a, x, f(y))[v \backslash x], P(a, v, w)[v \backslash x]\}$

$W = \{P(a, v, f(y)), P(a, v, w)\}$, $D \leftarrow (f(y), w)$

$|W| > 1$ and D satisfies the occurs check

$\sigma \leftarrow \{a \backslash u, v \backslash x\} \circ \{f(y) \backslash w\} = \{a \backslash u, v \backslash x, f(y) \backslash w\}$

$W \leftarrow \{P(a, v, f(y))[f(y) \backslash w], P(a, v, w)[f(y) \backslash w]\}$

$W = \{P(a, v, f(y))\}$; D is undefined;

$|W| = 1$, returns $\sigma = \{a \backslash u, v \backslash x, f(y) \backslash w\}$

Propositional Resolution

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
... and then came light!
The Resolution Principle
Resolution, at last
Unification
Another Example
The upshot
Literal Unification
Occurs Check
Most General Unifier
Unification - Example
▷ Propositional Resolution
The algorithm
Tautology Elimination
First-Order Resolution
Resolution Method
Soundness and Completeness
Subsumption
Historical References

The method proposed by Robinson in [Rob65b].

- □ There are no axioms.
- □ There is only one inference rule:

$$[\text{RESOLUTION}] \quad \frac{\begin{array}{ccc}(\varphi & \vee & l) \\ (\psi & \vee & \neg l)\end{array}}{\begin{array}{ccc}(\varphi & \vee & \psi)\end{array}}$$

- □ Premises are called parent clauses (or the resolvends).
- □ The conclusion is called the resolvent.
- □ The literals l and $\neg l$ are known as complementary literals.
- □ The parent clauses are *resolved* on the the complementary literals, generating the resolvent.

The algorithm

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
... and then came light!
The Resolution Principle
Resolution, at last
Unification
Another Example
The upshot
Literal Unification
Occurs Check
Most General Unifier
Unification - Example
Propositional Resolution
▷ The algorithm
Tautology Elimination
First-Order Resolution
Resolution Method
Soundness and Completeness
Subsumption
Historical References

Let φ be a propositional formula and Γ_0 be the set of clauses resulting from the transformation of φ into CNF.

do set $\Gamma = \Gamma_0$

select $l, c_1 \in \Gamma, c_2 \in \Gamma$ such that $\left\{ \begin{array}{l} l \in c_1, \neg l \in c_2, \\ c_1 \text{ and } c_2 \text{ have} \\ \text{not been resolved} \end{array} \right\}$

compute the resolvent r

replace Γ_0 by $\Gamma_0 \cup \{r\}$

until $\Box \in \Gamma_0$ or $\Gamma_0 = \Gamma$

Note that the algorithm is non-deterministic.

Tautology Elimination

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
... and then came light!
The Resolution Principle
Resolution, at last
Unification
Another Example
The upshot
Literal Unification
Occurs Check
Most General Unifier
Unification - Example
Propositional Resolution
The algorithm
▷ Tautology Elimination
First-Order Resolution
Resolution Method
Soundness and Completeness
Subsumption
Historical References

Let $\Gamma_i = \Gamma_0$
repeat
    Select c_1 e $c_2 \in \Gamma_i$ such that $l \in c_1, \neg l \in c_2$, where l is a literal and c_1 and c_2 have not been resolved
    Compute the resolvent r
    if not((r a tautology) or ($r \in \Gamma_i$)) **then**
        Do $\Gamma_{i+1} = \Gamma_i \cup \{r\}$
    end if
until $\square \in \Gamma_0$ or $\Gamma_{i+1} = \Gamma_i$

First-Order Resolution

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
... and then came light!
The Resolution Principle
Resolution, at last
Unification
Another Example
The upshot
Literal Unification
Occurs Check
Most General Unifier
Unification - Example
Propositional Resolution
The algorithm
Tautology Elimination
▷ First-Order Resolution
Resolution Method
Soundness and Completeness
Subsumption
Historical References

There are two inference rules:

$$\frac{\begin{array}{ccc} D & \vee & L_1 \\ D' & \vee & \neg L_2 \end{array}}{(D \vee D')\sigma}$$

where σ is the most general unifier for L_1 and L_2.

And we also need factoring:

Let $C \stackrel{\text{def}}{=} L_1 \vee \ldots \vee L_n$ be a clause and σ the most general unifier for a subset of C. Then, $C\sigma$ is the *factor* of C and is added to the set of clauses.

Resolution Method

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
... and then came light!
The Resolution Principle
Resolution, at last
Unification
Another Example
The upshot
Literal Unification
Occurs Check
Most General Unifier
Unification - Example
Propositional Resolution
The algorithm
Tautology Elimination
First-Order Resolution
▷ Resolution Method
Soundness and Completeness
Subsumption
Historical References

1. Let Γ_i be a set of clauses;
2. Choose clauses C_1 and C_2 with complementary literals in Γ_i;
3. Compute the resolvent R from C_1 and C_2, applying factoring;
4. If R is the empty clause, then stop and return "unsatisfiable";
5. Otherwise, $N_{i+1} \leftarrow N_i \cup \{R\}$;
6. If $N_{i+1} = N_i$, then stop and return "satisfiable";
7. Otherwise, go back to step 2.

Soundness and Completeness

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
... and then came light!
The Resolution Principle
Resolution, at last
Unification
Another Example
The upshot
Literal Unification
Occurs Check
Most General Unifier
Unification - Example
Propositional Resolution
The algorithm
Tautology Elimination
First-Order Resolution
Resolution Method
▷ Soundness and Completeness
Subsumption
Historical References

1. Resolution for first-order is strongly sound:

$$\text{if } \Gamma \vdash^{Res}_{\mathcal{L}_{fo}} \varphi, \text{ then } \Gamma \models_{\mathcal{L}_{fo}} \varphi$$

2. Resolution for first-order is refutationally complete:

$$\text{if } \Gamma \models_{\mathcal{L}_{fo}} \bot, \text{ then } \Gamma \vdash^{Res}_{\mathcal{L}_{fo}} \bot$$

Soundness and Completeness

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
... and then came light!
The Resolution Principle
Resolution, at last
Unification
Another Example
The upshot
Literal Unification
Occurs Check
Most General Unifier
Unification - Example
Propositional Resolution
The algorithm
Tautology Elimination
First-Order Resolution
Resolution Method
▷ Soundness and Completeness
Subsumption
Historical References

1. Resolution for first-order is strongly sound:

$$\text{if } \Gamma \vdash^{Res}_{\mathcal{L}_{fo}} \varphi\text{, then } \Gamma \models_{\mathcal{L}_{fo}} \varphi$$

2. Resolution for first-order is refutationally complete:

$$\text{if } \Gamma \models_{\mathcal{L}_{fo}} \bot\text{, then } \Gamma \vdash^{Res}_{\mathcal{L}_{fo}} \bot$$

Binary resolution is also consequence complete:

Lemma 1 ([Lee67]). Let C be a set of propositional clauses. If a clause D is a consequence of C, then there is a clause D' which is derived by binary propositional resolution from C and D' subsumes D.

Subsumption

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
... and then came light!
The Resolution Principle
Resolution, at last
Unification
Another Example
The upshot
Literal Unification
Occurs Check
Most General Unifier
Unification - Example
Propositional Resolution
The algorithm
Tautology Elimination
First-Order Resolution
Resolution Method
Soundness and Completeness
▷ Subsumption
Historical References

Lemma 2. Let Γ be a set of clauses such that clauses $C, C' \in \Gamma$ and $C \models C'$. Γ is unsatisfiable if, and only if, $\Gamma \setminus \{C'\}$ is unsatisfiable.

Obs.: If $C \models C'$, we say that C *subsumes* C'.

Historical References

Logic
The Early Days
DPLL
▷ Resolution
John Alan Robinson
Argonne
1962-1963
... and then came light!
The Resolution Principle
Resolution, at last
Unification
Another Example
The upshot
Literal Unification
Occurs Check
Most General Unifier
Unification - Example
Propositional Resolution
The algorithm
Tautology Elimination
First-Order Resolution
Resolution Method
Soundness and Completeness
Subsumption
▷ Historical References

The following is a list of historical accounts, some of which are given by people working in the early years of Automated Reasoning:

[Dav83], [Lov84], [BL84], [Bib07], and [Dav01] .

Logic
The Early Days
DPLL
Resolution
▷ References
References
References
References
References
References
References
References

References

References

[Bib07] Wolfgang Bibel. Early history and perspectives of automated deduction. In Joachim Hertzberg, Michael Beetz, and Roman Englert, editors, *KI 2007: Advances in Artificial Intelligence, 30th Annual German Conference on AI, KI 2007, Osnabrück, Germany, September 10-13, 2007, Proceedings*, volume 4667 of *Lecture Notes in Computer Science*, pages 2–18. Springer, 2007.

[BL84] Woodrow W. Bledsoe and Donald Loveland, editors. *Automated Theorem Proving: After 25 Years*, volume 19 of *Contemporary Mathematics Series*. American Math Society, 1984.

[Bla37] Archie Blake. *Canonical Expressions in Boolean Algebra*. PhD thesis, University of Chicago, Illinois, 1937.

[Dav57] M. Davis. A computer program for Presburger's algorithm. In Abraham Robinson, editor, *Proving Theorems, (as Done by Man, Logician, or Machine)*, pages 215–233, Cornell University, Ithaca, New York, 1957. Communications Research Division, Institute for Defense Analysis. Summaries of Talks Presented at the 1957 Summer Institute for Symbolic Logic. Second edition; publication date is 1960.

References

[Dav83] Martin Davis. The prehistory and early history of automated deduction. In Jörg Siekmann and Graham Wrightson, editors, *Automation of Reasoning: Classical Papers in Computational Logic 1957–1966*, volume 1, pages 1–28. Springer-Verlag, 1983.

[Dav01] Martin Davis. The early history of automated deduction. In A. Robinson and A. Voronkov, editors, *Handbook of Automated Reasoning*, volume I, chapter 1, pages 3–15. Elsevier Science, 2001.

[DFS59] Bradford Dunham, Richard Fridshal, and Gilbert Leinbaugh Sward. A non-heuristic program for proving elementary logical theorems. In *Proceedings of an International Conference on Information Processing*, pages 282–284, Paris, 1959. UNESCO.

[DLL62] Martin Davis, George Logemann, and Donald W. Loveland. A machine program for theorem-proving. *Communications of the ACM*, 5(7):394–397, 1962.

References

[DN63] Bradford Dunham and J.H. North. Theorem testing by computer. In *Proceedings of the Symposium on Mathematical Theory of Automata, Microwave Research Institute Symposium Series*, volume 12, pages 173–177. Polytechnic Institute of Brooklyn, New York, 1963.

[DP60] Martin Davis and Hilary Putnam. A computing procedure for quantification theory. *Journal of the ACM*, 7(3):201–215, 1960.

[FR74] Michael J. Fischer and Michael O. Rabin. Super-exponential complexity of presburger arithmetic. In R. M. Karp, editor, *Complexity of computation*, pages 27–41, Providence, RI, 1974. American Mathematical Society.

[Gel59] Herbert Gelernter. Realization of a geometry-theorem proving machine. In *Proceedings of an International Conference on Information Processing*, pages 273–282, Paris, 1959. UNESCO House.

[Gil60] Paul C. Gilmore. A proof method for quantification theory: Its justification and realization. *IBM J. Research and Development*, 4:28–35, January, 1960.

References

[Gil70] Paul C. Gilmore. An examination of the geometry theorem machine. *Artificial Intelligence*, 1(3–4):171–187, 1970.

[KK71] Robert Kowalski and Donald Kuehner. Linear resolution with selection function. *Artificial Intelligence*, 2(3–4):227–260, 1971.

[Lee67] Richard Chia-Tung Lee. *A completeness theorem and computer program for finding theorems derivable from given axioms*. PhD thesis, Berkeley, 1967.

[Lov70] Donald W. Loveland. A Linear Format for Resolution. In *Symposium on Automatic Demonstration*, volume 125 of *Lecture Notes in Mathematics*, pages 147–162, Berlin, 1970. Springer-Verlag.

[Lov84] Donald W. Loveland. Automated theorem proving: A quarter century review. *Contemporary Mathematics*, 29:1–45, 1984.

[NS56] Allen Newell and Herbert A. Simon. The logic theory machine: A complex information processing system. *IRE Transactions of information theory*, 2-3:61–79, 1956.

References

[NS57] Allen Newell and John Clifford Shaw. Programming the logic theory machine. In *Proceedings of the 1957 Western Joint Computer Conference*, pages 230–240. IRE, 1957.

[NSS57] Allen Newell, John Clifford Shaw, and Herbert A. Simon. Empirical explorations with the logic theory machine. In *Western Joint Computer Conference*, volume 15, pages 218–239, 1957. Also in Feigenbaum and Feldman, 1963.

[Opp78] Derek C. Oppen. A $2^{2^{2pn}}$ upper bound on the complexity of Presburger arithmetic. *Journal of Computer and System Sciences*, 16(3):323–332, June 1978.

[PPV60] Dag Prawitz, Håkan Prawitz, and Neri Voghera. A mechanical proof procedure and its realization in an electronic computer. *Journal of the ACM*, 7(2):102–128, 1960.

References

[Pre29] Mojżesz Presburger. Uber die vollstandigkeit eines gewissen systems der arithmetik ganzer zahlen, in welchem die addition als einzige operation hervorstritt. *Sprawozdanie z I Kongresu Matematikow Krajow Slowcanskich Warszawa*, pages 92–101, 1929.

[Rob63] John A. Robinson. Theorem-proving on the computer. *Journal of the ACM*, 10(2):163–174, April 1963.

[Rob65a] John A. Robinson. Automatic Deduction with Hyper-resolution. *International Journal of Computer Mathematics*, 1:227–234, 1965.

[Rob65b] John A. Robinson. A Machine–Oriented Logic Based on the Resolution Principle. *Journal of the ACM*, 12(1):23–41, January 1965.

[vOQ55] Willard van Orman Quine. A way to simplify truth functions. *American Mathematical Monthly*, 62:627–631, 1955.

References

[Wan60a] H. Wang. Toward mechanical mathematics. *IBM Journal of Research and Development*, 4(1):2–22, Jan 1960.

[Wan60b] Hao Wang. Proving theorems by pattern recognition i. *Commun. ACM*, 3(4):220–234, April 1960.

[Wan61] H. Wang. Proving theorems by pattern recognition – ii. *The Bell System Technical Journal*, 40(1):1–41, Jan 1961.

[WCR64] Lawrence Wos, Daniel F. Carson, and George A. Robinson. The Unit Preference Strategy in Theorem Proving. In *Proceedings of AFIPS Fall Joint Computer Conference*, pages 615–621. Thompson Book Company, 1964.

[WRC65] Lawrence Wos, George Robinson, and Daniel Carson. Efficiency and Completeness of the Set of Support Strategy in Theorem Proving. *Journal of the ACM*, 12:536–541, October 1965.