Formal Specification and Verification

8.11.2016

Viorica Sofronie-Stokkermans

e-mail: sofronie@uni-koblenz.de

Mathematical foundations

Formal logic:

- Syntax: a formal language (formula expressing facts)
- Semantics: to define the meaning of the language, that is which facts are valid)
- Deductive system: made of axioms and inference rules to formaly derive theorems, that is facts that are provable

Last time

Propositional classical logic

- Syntax
- Semantics

  Models, Validity, and Satisfiability

  Entailment and Equivalence
- Checking Unsatisfiability

  Truth tables

  "Rewriting" using equivalences

  Proof systems: clausal/non-clausal

Today

Propositional classical logic

Proof systems: clausal/non-clausal

- non-clausal: Hilbert calculus

  sequent calculus

- clausal: Resolution; DPLL (translation to CNF needed)

- Binary Decision Diagrams

Today

Propositional classical logic

Proof systems: clausal/non-clausal

- non-clausal: Hilbert calculus

  sequent calculus

- clausal: Resolution; DPLL (translation to CNF needed)

- Binary Decision Diagrams

A deductive system for Propositional logic

Variant of the system of Hilbert-Ackermann

(Signature: $\vee, \neg$; $\quad x \rightarrow y \equiv_{\text{Def}} \neg x \vee y$)

Axiom Schemata (to be instantiated for all possible formulae)

(1) $(p \vee p) \rightarrow p$

(2) $p \rightarrow (q \vee p)$

(3) $(p \vee q) \rightarrow (q \vee p)$

(4) $(p \rightarrow q) \rightarrow (r \vee p \rightarrow r \vee q)$

Inference rules

Modus Ponens: $\dfrac{p, \quad p \rightarrow q}{q}$

Example of proof

Prove $\phi \vee \neg\phi$

1. $((\phi \vee \phi) \rightarrow \phi) \rightarrow (\neg\phi \vee (\phi \vee \phi) \rightarrow \neg\phi \vee \phi)$ [Instance of (4)]

2. $\phi \vee \phi \rightarrow \phi$ [Instance of (1)]

3. $\neg\phi \vee (\phi \vee \phi) \rightarrow (\neg\phi \vee \phi)$ [1., 2., and MP]

3'. $= (\phi \rightarrow (\phi \vee \phi)) \rightarrow (\neg\phi \vee \phi)$ [3 and definition of $\rightarrow$]

4. $\phi \rightarrow \phi \vee \phi$ [Instance of (2)]

5. $\neg\phi \vee \phi$ [3., 4. and MP]

6. $(\neg\phi \vee \phi) \rightarrow (\phi \vee \neg\phi)$ [Instance of (3)]

7. $\phi \vee \neg\phi)$ [5., 6. and MP]

Soundness

Γ is called sound $:\Leftrightarrow$

$$\frac{F_1 \;\; \ldots \;\; F_n}{F} \in \Gamma \quad \Rightarrow \quad F_1, \ldots, F_n \models F$$

Γ sound iff If $N \vdash_\Gamma F$ then $N \models F$.

Theorem. The Hilbert deductive system is sound.

Proof: The proof for propositional logic is by induction on the length of the formal proof of F from N.

Proof of length 0: show that all axioms are valid

Induction step $n \mapsto n + 1$: uses the definition of a proof.
It is sufficient to show that $(\phi \wedge (\phi \rightarrow \phi')) \models \phi'$.

Completeness

Γ is called complete $:\Leftrightarrow$

$$N \models F \quad \Rightarrow \quad N \vdash_\Gamma F$$

Theorem. The Hilbert deductive system is complete.

Completeness: Proof Idea

Entailment vs. Validity: $N, F \models G$ iff $N \models F \rightarrow G$.

Deduction Theorem: $N, F \vdash G$ iff $N \vdash F \rightarrow G$.

Definition: A set N of formulae is inconsistent if there is a formula F such that $N \models F$ and $N \models \neg F$.

$N \models F$ iff $N \cup \{\neg F\}$ unsatisfiable

$N \vdash F$ iff $N \cup \{\neg F\}$ inconsistent

Proof idea

To show: $N \models F \Rightarrow N \vdash F$

equivalent to: $N \cup \{\neg F\}$ unsatisfiable $\Rightarrow N \cup \{\neg F\}$ inconsistent.

equivalent to: $N \cup \{\neg F\}$ consistent $\Rightarrow N \cup \{\neg F\}$ satisfiable

Completeness: Proof

We show: For every set N of formulae, if N is consistent then N is satisfiable.

Proof: Let $F_1, \ldots, F_n, \ldots$ an enumeration of all propositional logic formulae over Π.

Given N consistent, define a sequence of sets of formulae $N_0, N_1, N_2 \ldots$ by:

$$N_0 = N$$

$$N_{n+1} = \begin{cases} N_n \cup \{F_n\} & \text{if } N_n \cup \{F_n\} \text{ consistent} \\ N_n \cup \{\neg F_n\} & \text{if } N_n \cup \{\neg F_n\} \text{ consistent} \end{cases}$$

$N_0 \subseteq N_1 \subseteq N_2 \subseteq \cdots \subseteq N_n \subseteq \ldots$ and all these sets are consistent.

Let $N^* = \bigcup_{n \in \mathbb{N}} N_i$. N^* is consistent. We define a valuation $\mathcal{A}$ with

$$\mathcal{A}(P) = \begin{cases} 1 & \text{if } P \in N^* \\ 0 & \text{if } \neg P \in N^* \end{cases}$$

Then we can show that:

$$\mathcal{A}(F) = \begin{cases} 1 & \text{if } F \in N^* \\ 0 & \text{if } \neg F \in N^* \end{cases}$$

Hence, $\mathcal{A} \models N$

Overview

Propositional classical logic

Proof systems: clausal/non-clausal

- non-clausal: Hilbert calculus

  sequent calculus

- clausal: Resolution; DPLL (translation to CNF needed)

- Binary Decision Diagrams

Sequent calculus for propositional logic

Sequent Calculus based on notion of sequent

$$\underbrace{\psi_1, \ldots, \psi_m}_{\text{Antecedent}} \Rightarrow \underbrace{\phi_1, \ldots, \phi_n}_{\text{Succedent}}$$

Has same semantics as

$$\models \psi_1 \wedge \cdots \wedge \psi_m \rightarrow (\phi_1 \vee \cdots \vee \phi_n)$$

$$\{\psi_1, \ldots, \psi_m\} \models \phi_1 \vee \cdots \vee \phi_n$$

Notation for Sequents

$$\underbrace{\psi_1, \ldots, \psi_m}_{\text{Antecedent}} \Rightarrow \underbrace{\phi_1, \ldots, \phi_n}_{\text{Succedent}}$$

Consider antecedent/succedent as sets of formulae (may be empty)

Notation for Sequents

$$\underbrace{\psi_1, \ldots, \psi_m}_{\text{Antecedent}} \Rightarrow \underbrace{\phi_1, \ldots, \phi_n}_{\text{Succedent}}$$

Consider antecedent/succedent as sets of formulae (may be empty)

Conventions:

- empty antecedent = empty conjunction = $\top$
- empty succedent = empty disjunction = $\bot$

Notation for Sequents

$$\underbrace{\psi_1, \ldots, \psi_m}_{\text{Antecedent}} \Rightarrow \underbrace{\phi_1, \ldots, \phi_n}_{\text{Succedent}}$$

Consider antecedent/succedent as sets of formulae (may be empty)

Conventions:

- empty antecedent = empty conjunction = $\top$
- empty succedent = empty disjunction = $\bot$

Alternative notation:

$\psi_1, \ldots, \psi_m \vdash \phi_1, \ldots, \phi_n$

Not used here because of the risk of potential confusion with the provability relation

Notation for Sequents

$$\underbrace{\psi_1, \ldots, \psi_m}_{\text{Antecedent}} \Rightarrow \underbrace{\phi_1, \ldots, \phi_n}_{\text{Succedent}}$$

Consider antecedent/succedent as sets of formulas, may be empty

Schema Variables:

$\phi, \psi, \ldots$ match formulas, $\Gamma, \Delta, \ldots$ match sets of formulas

Characterize infinitely many sequents with a single schematic sequent:

Example: $\Gamma \Rightarrow \Delta, \phi \wedge \psi$
Matches any sequent with occurrence of conjunction in succedent
We call $\phi \wedge \psi$ main formula and Γ, Δ side formulae of sequent.

Sequent Calculus Rules of Propositional Logic

Write syntactic transformation schema for sequents that reflects semantics of connectives as closely as possible

$$\text{Rule Name}\quad \frac{\overbrace{\Gamma_1 \Rightarrow \Delta_1 \ldots\ \Gamma_n \Rightarrow \Delta_n}^{\text{premises}}}{\underbrace{\Gamma \Rightarrow \Delta}_{\text{conclusion}}}\ .$$

Sequent Calculus Rules of Propositional Logic

Write syntactic transformation schema for sequents that reflects semantics of connectives as closely as possible

$$\text{Rule Name} \quad \frac{\overbrace{\Gamma_1 \Rightarrow \Delta_1 \ldots \ \Gamma_n \Rightarrow \Delta_n}^{\text{premises}}}{\underbrace{\Gamma \Rightarrow \Delta}_{\text{conclusion}}} \ .$$

Example:

$$\text{andRight} \quad \frac{\Gamma \Rightarrow \phi, \Delta \quad \Gamma \Rightarrow \psi, \Delta}{\Gamma \Rightarrow \phi \wedge \psi, \Delta} \ .$$

Sequent Calculus Rules of Propositional Logic

Write syntactic transformation schema for sequents that reflects semantics of connectives as closely as possible

$$\text{Rule Name} \quad \frac{\overbrace{\Gamma_1 \Rightarrow \Delta_1 \ldots \ \Gamma_n \Rightarrow \Delta_n}^{\text{premises}}}{\underbrace{\Gamma \Rightarrow \Delta}_{\text{conclusion}}} \ .$$

Example:

$$\text{andRight} \quad \frac{\Gamma \Rightarrow \phi, \Delta \quad \Gamma \Rightarrow \psi, \Delta}{\Gamma \Rightarrow \phi \wedge \psi, \Delta} \ .$$

Informal meaning:

In order to prove that Γ entails $(\phi \wedge \psi) \vee \Delta$ we need to prove that:
- Γ entails $\phi \vee \Delta$ and
- Γ entails $\psi \vee \Delta$

Sequent Calculus Rules of Propositional Logic

Write syntactic transformation schema for sequents that reflects semantics of connectives as closely as possible

$$\text{Rule Name} \quad \frac{\overbrace{\Gamma_1 \Rightarrow \Delta_1 \ldots \; \Gamma_n \Rightarrow \Delta_n}^{\text{premises}}}{\underbrace{\Gamma \Rightarrow \Delta}_{\text{conclusion}}} \; .$$

Example:

$$\text{andRight} \quad \frac{\Gamma \Rightarrow \phi, \Delta \quad \Gamma \Rightarrow \psi, \Delta}{\Gamma \Rightarrow \phi \wedge \psi, \Delta} \; .$$

Sound rule (essential): If $\models (\Gamma_1 \rightarrow \Delta_1)$ and $\ldots$ $\models (\Gamma_n \rightarrow \Delta_n)$ then $\models (\Gamma \rightarrow \Delta)$

Sequent Calculus Rules of Propositional Logic

Write syntactic transformation schema for sequents that reflects semantics of connectives as closely as possible

$$\text{Rule Name} \quad \frac{\overbrace{\Gamma_1 \Rightarrow \Delta_1 \ldots \; \Gamma_n \Rightarrow \Delta_n}^{\text{premises}}}{\underbrace{\Gamma \Rightarrow \Delta}_{\text{conclusion}}} \; .$$

Example:

$$\text{andRight} \quad \frac{\Gamma \Rightarrow \phi, \Delta \quad \Gamma \Rightarrow \psi, \Delta}{\Gamma \Rightarrow \phi \wedge \psi, \Delta} \; .$$

Sound rule (essential): If $\models (\Gamma_1 \rightarrow \Delta_1)$ and ... and $\models (\Gamma_n \rightarrow \Delta_n)$ then $\models (\Gamma \rightarrow \Delta)$

Complete rule (desirable): If $\models (\Gamma \rightarrow \Delta)$ then $\models (\Gamma_1 \rightarrow \Delta_1), \ldots \models (\Gamma_n \rightarrow \Delta_n)$

Rules of Propositional Sequent Calculus

main	left side (antecedent)	right side (succedent)
not	$\dfrac{\Gamma \Rightarrow \phi, \Delta}{\Gamma, \neg\phi \Rightarrow \Delta}$	$\dfrac{\Gamma, \phi \Rightarrow \Delta}{\Gamma \Rightarrow \neg\phi, \Delta}$

Rules of Propositional Sequent Calculus

main	left side (antecedent)	right side (succedent)
not	$\dfrac{\Gamma \Rightarrow \phi, \Delta}{\Gamma, \neg\phi \Rightarrow \Delta}$	$\dfrac{\Gamma, \phi \Rightarrow \Delta}{\Gamma \Rightarrow \neg\phi, \Delta}$
and	$\dfrac{\Gamma, \phi, \psi \Rightarrow \Delta}{\Gamma, \phi \wedge \psi \Rightarrow \Delta}$	$\dfrac{\Gamma \Rightarrow \phi, \Delta \quad \Gamma \Rightarrow \psi, \Delta}{\Gamma \Rightarrow \phi \wedge \psi, \Delta}$

Rules of Propositional Sequent Calculus

main	left side (antecedent)	right side (succedent)
not	$\dfrac{\Gamma \Rightarrow \phi, \Delta}{\Gamma, \neg\phi \Rightarrow \Delta}$	$\dfrac{\Gamma, \phi \Rightarrow \Delta}{\Gamma \Rightarrow \neg\phi, \Delta}$
and	$\dfrac{\Gamma, \phi, \psi \Rightarrow \Delta}{\Gamma, \phi \wedge \psi \Rightarrow \Delta}$	$\dfrac{\Gamma \Rightarrow \phi, \Delta \quad \Gamma \Rightarrow \psi, \Delta}{\Gamma \Rightarrow \phi \wedge \psi, \Delta}$
or	$\dfrac{\Gamma, \phi \Rightarrow \Delta \quad \Gamma, \psi \Rightarrow \Delta}{\Gamma, \phi \vee \psi \Rightarrow \Delta}$	$\dfrac{\Gamma \Rightarrow \phi, \psi, \Delta}{\Gamma \Rightarrow \phi \vee \psi, \Delta}$

Rules of Propositional Sequent Calculus

main	left side (antecedent)	right side (succedent)
not	$\dfrac{\Gamma \Rightarrow \phi, \Delta}{\Gamma, \neg\phi \Rightarrow \Delta}$	$\dfrac{\Gamma, \phi \Rightarrow \Delta}{\Gamma \Rightarrow \neg\phi, \Delta}$
and	$\dfrac{\Gamma, \phi, \psi \Rightarrow \Delta}{\Gamma, \phi \wedge \psi \Rightarrow \Delta}$	$\dfrac{\Gamma \Rightarrow \phi, \Delta \quad \Gamma \Rightarrow \psi, \Delta}{\Gamma \Rightarrow \phi \wedge \psi, \Delta}$
or	$\dfrac{\Gamma, \phi \Rightarrow \Delta \quad \Gamma, \psi \Rightarrow \Delta}{\Gamma, \phi \vee \psi \Rightarrow \Delta}$	$\dfrac{\Gamma \Rightarrow \phi, \psi, \Delta}{\Gamma \Rightarrow \phi \vee \psi, \Delta}$
imp	$\dfrac{\Gamma \Rightarrow \phi, \Delta \quad \Gamma, \psi \Rightarrow \Delta}{\Gamma, \phi \rightarrow \psi \Rightarrow \Delta}$	$\dfrac{\Gamma, \phi \Rightarrow \psi, \Delta}{\Gamma \Rightarrow \phi \rightarrow \psi, \Delta}$

Rules of Propositional Sequent Calculus

main	left side (antecedent)	right side (succedent)
not	$\dfrac{\Gamma \Rightarrow \phi, \Delta}{\Gamma, \neg\phi \Rightarrow \Delta}$	$\dfrac{\Gamma, \phi \Rightarrow \Delta}{\Gamma \Rightarrow \neg\phi, \Delta}$
and	$\dfrac{\Gamma, \phi, \psi \Rightarrow \Delta}{\Gamma, \phi \wedge \psi \Rightarrow \Delta}$	$\dfrac{\Gamma \Rightarrow \phi, \Delta \quad \Gamma \Rightarrow \psi, \Delta}{\Gamma \Rightarrow \phi \wedge \psi, \Delta}$
or	$\dfrac{\Gamma, \phi \Rightarrow \Delta \quad \Gamma, \psi \Rightarrow \Delta}{\Gamma, \phi \vee \psi \Rightarrow \Delta}$	$\dfrac{\Gamma \Rightarrow \phi, \psi, \Delta}{\Gamma \Rightarrow \phi \vee \psi, \Delta}$
imp	$\dfrac{\Gamma \Rightarrow \phi, \Delta \quad \Gamma, \psi \Rightarrow \Delta}{\Gamma, \phi \rightarrow \psi \Rightarrow \Delta}$	$\dfrac{\Gamma, \phi \Rightarrow \psi, \Delta}{\Gamma \Rightarrow \phi \rightarrow \psi, \Delta}$

close $\dfrac{}{\Gamma, \phi \Rightarrow \phi, \Delta}$ true $\dfrac{}{\Gamma \Rightarrow \mathsf{true}, \Delta}$ false $\dfrac{}{\Gamma, \mathsf{false} \Rightarrow \Delta}$

Justification of Rules

Compute rules by applying semantic definitions

Justification of Rules

Compute rules by applying semantic definitions

orRight $\frac{\Gamma \Rightarrow \phi, \psi, \Delta}{\Gamma \Rightarrow \phi \vee \psi, \Delta}$

Follows directly from semantics of sequents

Justification of Rules

Compute rules by applying semantic definitions

orRight $\frac{\Gamma \Rightarrow \phi, \psi, \Delta}{\Gamma \Rightarrow \phi \vee \psi, \Delta}$

Follows directly from semantics of sequents

andRight $\frac{\Gamma \Rightarrow \phi, \Delta \quad \Gamma \Rightarrow \psi, \Delta}{\Gamma \Rightarrow \phi \wedge \psi, \Delta}$

$\models \Gamma \rightarrow (\phi \wedge \psi) \vee \Delta$ iff ($\models \Gamma \rightarrow \phi \vee \Delta$ and $\models \Gamma \rightarrow \psi \vee \Delta$)

Sequent Calculus Proofs

Goal to prove: $\mathcal{G} = (\psi_1, \ldots, \psi_m \Rightarrow \phi_1, \ldots, \phi_n)$

Sequent Calculus Proofs

Goal to prove: $\mathcal{G} = (\psi_1, \ldots, \psi_m \Rightarrow \phi_1, \ldots, \phi_n)$

- find rule R whose conclusion matches $\mathcal{G}$
- instantiate R such that conclusion identical to $\mathcal{G}$
- recursively find proofs for resulting premisses $\mathcal{G}_1, \ldots, \mathcal{G}_r$
- tree structure with goal as root
- close proof branch when rule without premises encountered

A Simple Proof

$$\Rightarrow (p \wedge (p \rightarrow q)) \rightarrow q)$$

A Simple Proof

$$\dfrac{p \wedge (p \rightarrow q) \Rightarrow q}{\Rightarrow (p \wedge (p \rightarrow q)) \rightarrow q} \qquad \text{(imp), right}$$

A Simple Proof

$$\cfrac{\cfrac{p, (p \rightarrow q) \Rightarrow q}{p \wedge (p \rightarrow q) \Rightarrow q}}{\Rightarrow (p \wedge (p \rightarrow q)) \rightarrow q}$$

(and), left

(imp), right

A Simple Proof

$$\cfrac{\cfrac{\cfrac{p \Rightarrow q, p \qquad p, q \Rightarrow q}{p, (p \rightarrow q) \Rightarrow q}}{p \wedge (p \rightarrow q) \Rightarrow q}}{\Rightarrow (p \wedge (p \rightarrow q)) \rightarrow q)}$$

(imp), left

(and), left

(imp), right

A Simple Proof

$$\dfrac{\dfrac{\dfrac{\dfrac{}{p \Rightarrow q, p} \qquad \dfrac{}{p, q \Rightarrow q}}{p, (p \to q) \Rightarrow q}}{p \wedge (p \to q) \Rightarrow q}}{\Rightarrow (p \wedge (p \to q)) \to q}$$

close, close

(imp), left

(and), left

(imp), right

A Simple Proof

$$\dfrac{\dfrac{\dfrac{\dfrac{\text{close } *}{p \Rightarrow q, p} \qquad \dfrac{\text{close } *}{p, q \Rightarrow q}}{p, (p \rightarrow q) \Rightarrow q}}{p \wedge (p \rightarrow q) \Rightarrow q}}{\Rightarrow (p \wedge (p \rightarrow q)) \rightarrow q)}$$

A proof is closed iff all its branches are closed

Soundness, Completeness, Termination

Soundness and completeness can be proved for every rule:

Sound: If $\models (\Gamma_1 \to \Delta_1)$ and ... and $\models (\Gamma_n \to \Delta_n)$ then $\models (\Gamma \to \Delta)$

Complete: If $\models (\Gamma{\to}\Delta)$ then $\models (\Gamma_1{\to}\Delta_1), \ldots \models (\Gamma_n{\to}\Delta_n)$

Soundness, Completeness

Soundness and completeness can be proved for every rule:

Sound: If $\models (\Gamma_1 \rightarrow \Delta_1)$ and ... and $\models (\Gamma_n \rightarrow \Delta_n)$ then $\models (\Gamma \rightarrow \Delta)$

Complete: If $\models (\Gamma \rightarrow \Delta)$ then $\models (\Gamma_1 \rightarrow \Delta_1), \ldots \models (\Gamma_n \rightarrow \Delta_n)$

Consequence: The following are equivalent:

(1) $\Gamma \models \Delta$

(2) there exists a proof in the sequent calculus for $\Gamma \Rightarrow \Delta$.

Overview

Propositional classical logic

Proof systems: clausal/non-clausal

- non-clausal: Hilbert calculus

  sequent calculus

- clausal: Resolution; DPLL (translation to CNF needed)
- Binary Decision Diagrams

The Propositional Resolution Calculus

Resolution inference rule:

$$\frac{C \vee A \qquad \neg A \vee D}{C \vee D}$$

Terminology: $C \vee D$: resolvent; A: resolved atom

(Positive) factorisation inference rule:

$$\frac{C \vee A \vee A}{C \vee A}$$

The Resolution Calculus *Res*

These are schematic inference rules; for each substitution of the schematic variables C, D, and A, respectively, by propositional clauses and atoms we obtain an inference rule.

As "$\vee$" is considered associative and commutative, we assume that A and $\neg A$ can occur anywhere in their respective clauses.

Sample Refutation

1.	$\neg P \vee \neg P \vee Q$	(given)
2.	$P \vee Q$	(given)
3.	$\neg R \vee \neg Q$	(given)
4.	R	(given)
5.	$\neg P \vee Q \vee Q$	(Res. 2. into 1.)
6.	$\neg P \vee Q$	(Fact. 5.)
7.	$Q \vee Q$	(Res. 2. into 6.)
8.	Q	(Fact. 7.)
9.	$\neg R$	(Res. 8. into 3.)
10.	$\perp$	(Res. 4. into 9.)

Resolution with Implicit Factorization *RIF*

$$\frac{C \vee A \vee \ldots \vee A \qquad \neg A \vee D}{C \vee D}$$

1.	$\neg P \vee \neg P \vee Q$	(given)
2.	$P \vee Q$	(given)
3.	$\neg R \vee \neg Q$	(given)
4.	R	(given)
5.	$\neg P \vee Q \vee Q$	(Res. 2. into 1.)
6.	$Q \vee Q \vee Q$	(Res. 2. into 5.)
7.	$\neg R$	(Res. 6. into 3.)
8.	$\perp$	(Res. 4. into 7.)

Soundness and Completeness

Theorem 1.6. Propositional resolution is sound.

for both the resolution rule and the positive factorization rule

the conclusion of the inference is entailed by the premises.

Theorem 1.7. Propositional resolution is refutationally complete.

If $N \models \bot$ we can deduce $\bot$ starting from N and using

the inference rules of the propositional resolution calculus.

The DPLL Procedure

Goal:

Given a propositional formula in CNF (or alternatively, a finite set N of clauses), check whether it is satisfiable (and optionally: output *one* solution, if it is satisfiable).

Satisfiability of Clause Sets

$\mathcal{A} \models N$ if and only if $\mathcal{A} \models C$ for all clauses C in N.

$\mathcal{A} \models C$ if and only if $\mathcal{A} \models L$ for some literal $L \in C$.

Partial Valuations

Since we will construct satisfying valuations incrementally,
we consider partial valuations
(that is, partial mappings $\mathcal{A} : \Pi \to \{0, 1\}$).

We start with an empty valuation and try to extend it step by step to all variables occurring in N.

If $\mathcal{A}$ is a partial valuation, then literals and clauses can be true, false, or undefined under $\mathcal{A}$.

A clause is true under $\mathcal{A}$ if one of its literals is true;
it is false (or "conflicting") if all its literals are false;
otherwise it is undefined (or "unresolved").

Unit Clauses

Observation:

Let $\mathcal{A}$ be a partial valuation. If the set N contains a clause C, such that all literals but one in C are false under $\mathcal{A}$, then the following properties are equivalent:

- there is a valuation that is a model of N and extends $\mathcal{A}$.
- there is a valuation that is a model of N and extends $\mathcal{A}$ and makes the remaining literal L of C true.

C is called a unit clause; L is called a unit literal.

Pure Literals

One more observation:

Let $\mathcal{A}$ be a partial valuation and P a variable that is undefined under $\mathcal{A}$. If P occurs only positively (or only negatively) in the unresolved clauses in N, then the following properties are equivalent:

- there is a valuation that is a model of N and extends $\mathcal{A}$.
- there is a valuation that is a model of N and extends $\mathcal{A}$ and assigns true (false) to P.

P is called a pure literal.

Example (Idea)

A succinct formulation:

State: $M||F$,

where:

- M partial assignment (sequence of literals),

  some literals are annotated (L^d: decision literal)

- F clause set.

A succinct formulation

UnitPropagation

$M||F, C \vee L \Rightarrow M, L||F, C \vee L$ if $M \models \neg C$, and L undef. in M

Decide

$M||F \Rightarrow M, L^d||F$ if L or $\neg L$ occurs in F, L undef. in M

Fail

$M||F, C \Rightarrow \text{Fail}$ if $M \models \neg C$, M contains no decision literals

Backjump

$$M, L^d, N||F \Rightarrow M, L'||F \quad \text{if} \begin{cases} \text{there is some clause } C \vee L' \text{ s.t.:} \\ F \models C \vee L', M \models \neg C, \\ L' \text{ undefined in } M \\ L' \text{ or } \neg L' \text{ occurs in } F. \end{cases}$$

Example

Assignment:	Clause set:	
$\emptyset$	$\|\|\neg P_1 \vee P_2, \neg P_3 \vee P_4, \neg P_5 \vee \neg P_6, P_6 \vee \neg P_5 \vee \neg P_2$	$\Rightarrow$ (Decide)
${P_1}^d$	$\|\|\neg P_1 \vee P_2, \neg P_3 \vee P_4, \neg P_5 \vee \neg P_6, P_6 \vee \neg P_5 \vee \neg P_2$	$\Rightarrow$ (UnitProp
${P_1}^d P_2$	$\|\|\neg P_1 \vee P_2, \neg P_3 \vee P_4, \neg P_5 \vee \neg P_6, P_6 \vee \neg P_5 \vee \neg P_2$	$\Rightarrow$ (Decide)
${P_1}^d P_2 {P_3}^d$	$\|\|\neg P_1 \vee P_2, \neg P_3 \vee P_4, \neg P_5 \vee \neg P_6, P_6 \vee \neg P_5 \vee \neg P_2$	$\Rightarrow$ (UnitProp
${P_1}^d P_2 {P_3}^d P_4$	$\|\|\neg P_1 \vee P_2, \neg P_3 \vee P_4, \neg P_5 \vee \neg P_6, P_6 \vee \neg P_5 \vee \neg P_2$	$\Rightarrow$ (Decide)
${P_1}^d P_2 {P_3}^d P_4 {P_5}^d$	$\|\|\neg P_1 \vee P_2, \neg P_3 \vee P_4, \neg P_5 \vee \neg P_6, P_6 \vee \neg P_5 \vee \neg P_2$	$\Rightarrow$ (UnitProp
${P_1}^d P_2 {P_3}^d P_4 {P_5}^d \neg P_6$	$\|\|\neg P_1 \vee P_2, \neg P_3 \vee P_4, \neg P_5 \vee \neg P_6, P_6 \vee \neg P_5 \vee \neg P_2$	$\Rightarrow$ (Backtrac
${P_1}^d P_2 {P_3}^d P_4 \neg P_5$	$\|\|\neg P_1 \vee P_2, \neg P_3 \vee P_4, \neg P_5 \vee \neg P_6, P_6 \vee \neg P_5 \vee \neg P_2$	...