CS257: Introduction to Automated Reasoning

Proof Systems

Stanford University

CENTAUR

Agenda

- Abstract Proof Systems
- Satisfiability Proof Systems
- Soundness, Completeness, Termination, and Progressiveness
- A Decision Procedure for Propositional Logic
- Strategies

Next lecture: Normal forms, Solving SAT

Proofs

What is a proof?

- A sequence of steps leading from some assumptions to some conclusions
- Each step should be convincing and should be drawn from a set of accepted **proof rules**

Proof theory is a branch of mathematical logic in which proofs themselves are formal objects we can prove things about

In automated reasoning, representing algorithms as proof systems has several advantages

- Modular and composable
- Easier to prove things about the algorithms
- Can choose which implementation details to highlight and which to leave out

Abstract Proof Systems

An **abstract proof system** is a tuple $\mathbb{P} = \langle \mathbb{P}^S, \mathbb{P}^R \rangle$ where $\mathbb{P}^S$ is a set of proof states and $\mathbb{P}^R$ is a set of proof rules. Each proof rule is a partial function from proof states to sets of proof states.

Proof state

- Represents what is known and assumed at each stage of the proof
- Example of a proof state: a set of propositional formulas

Proof Rules

Proof rule

- Takes an input proof state
- Is only applicable if input proof state satisfies some **premises**
- Returns one or more proof states, the **conclusions**, said to be **derived** from the input state

Notation for proof rules:

$$\text{R} \quad \frac{P_1 \quad P_2 \quad \cdots \quad P_m}{C_1 \quad C_2 \quad \cdots \quad C_n}$$

- R is the rule name (for rerferende)
- Each P_i is a premise
- Each C_i is a conclusion

A Proof System for Propositional Logic

Let $\mathbb{P}_{PL} = \langle \mathbb{P}_{PL}^S, \mathbb{P}_{PL}^P \rangle$ be a proof system for propositional logic

- A proof state $\mathbb{S} \in \mathbb{P}_{PL}^S$ is a set of well-formed propositional logic formulas
- Suppose $\mathbb{P}_{PL}^P$ contains the **modus ponens** rule (MP for short)
  - Let $\mathcal{L}$ be the set of propositional literals (i.e., variables or their negations)
  - We use $\mathbb{S}$ to represent the state the rule is being applied to
  - We can write MP as follows:

$$\text{MP} \quad \frac{p, q \in \mathcal{L} \quad p \in \mathbb{S} \quad p \rightarrow q \in \mathbb{S} \quad q \notin \mathbb{S}}{\mathbb{S} \cup \{q\}}$$

Technically, MP is a proof rule **schema**

- p and q are **parameters**, and each possible instantiation is a separate proof rule
- For convenience, we will refer to both proof rules and proof rule schemas as "proof rules"

Example

$$\text{MP} \quad \frac{p, q \in \mathcal{L} \quad p \in \mathbb{S} \quad p \to q \in \mathbb{S} \quad q \notin \mathbb{S}}{\mathbb{S} \cup \{q\}}$$

Suppose a, b, c, d are propositional variables

What is the result of applying MP to the following proof states?

- $\{a, a \to b\}$ $\qquad$ $\{a, a \to b, b\}$
- $\{a \lor \neg c, \neg d, \neg d \to b\}$ $\qquad$ $\{a \lor \neg c, \neg d, \neg d \to b, b\}$
- $\{c, d, c \to d\}$ $\qquad$ Does not apply

A Proof System for Propositional Logic

Let $\mathcal{V}$ be the set of all propositional variables

Let us consider another rule:

$$\text{Split} \quad \frac{p \in \mathcal{V} \quad p \text{ occurs in some formula in } \mathbb{S} \quad p \notin \mathbb{S} \quad \neg p \notin \mathbb{S}}{\mathbb{S} \cup \{p\} \qquad \mathbb{S} \cup \{\neg p\}}$$

Can we apply Split to $\{a \vee (b \wedge c), \neg d\}$?

- Yes, if we choose p to represent a, b, or c, but not d

Let Splitb be the proof rule obtained by instantiating the parameter p with b

Then, formally:

- $\text{Splitb}(\{a \vee (b \wedge c), \neg d\}) = \{\{a \vee (b \wedge c), \neg d, b\}, \{a \vee (b \wedge c), \neg d, \neg b\}\}$

Derivation Trees

Let $\mathbb{P} = \langle \mathbb{P}^S, \mathbb{P}^R \rangle$ be an abstract proof system

- A proof state $\mathbb{S}$ is **reducible** with respect to $\mathbb{P}$ if at least one of the proof rules in $\mathbb{P}^R$ applies
- A $\mathbb{P}$**-derivation tree** from $\mathbb{S}$ is a finite tree whose nodes are taken from $\mathbb{P}^S$, whose root is $\mathbb{S}$, and with the property that each internal node $\mathbb{S}'$ of the tree is reducible with respect to $\mathbb{P}$, and its children are the conclusions resulting from applying some rule in $\mathbb{P}^R$ to $\mathbb{S}'$
- A derivation tree is **reducible** with respect to $\mathbb{P}$ if at least one of its leaves is reducible with respect to $\mathbb{P}$

Derivation Tree Example

What could a derivation tree from $\{b \to c, \neg b \to c, \neg c\}$ look like?

$$\frac{\{b \to c, \neg b \to c, \neg c\}}{}$$

Derivation Tree Example

What could a derivation tree from $\{b \to c, \neg b \to c, \neg c\}$ look like?

$$\text{Split}\quad \frac{\{b \to c, \neg b \to c, \neg c\}}{\{b \to c, \neg b \to c, \neg c, b\} \quad \{b \to c, \neg b \to c, \neg c, \neg b\}}$$

Derivation Tree Example

What could a derivation tree from $\{b \to c, \neg b \to c, \neg c\}$ look like?

$$\text{Split}\ \dfrac{\{b \to c, \neg b \to c, \neg c\}}{\text{MP}\ \dfrac{\{b \to c, \neg b \to c, \neg c, b\}}{\{b \to c, \neg b \to c, \neg c, b, c\}} \qquad \{b \to c, \neg b \to c, \neg c, \neg b\}}$$

Derivation Tree Example

What could a derivation tree from $\{b \to c, \neg b \to c\}$ look like?

$$\text{Split}\ \dfrac{\{b \to c, \neg b \to c, \neg c\}}{\text{MP}\ \dfrac{\{b \to c, \neg b \to c, \neg c, b\}}{\{b \to c, \neg b \to c, \neg c, b, c\}} \qquad \text{MP}\ \dfrac{\{b \to c, \neg b \to c, \neg c, \neg b\}}{\{b \to c, \neg b \to c, \neg c, \neg b, c\}}}$$

Derivations

Let $\mathbb{P} = \langle \mathbb{P}^S, \mathbb{P}^R \rangle$ be an abstract proof system

- A $\mathbb{P}$**-derivation** from τ is a (possibly infinite) sequence of derivation trees starting with a $\mathbb{P}$-derivation tree τ, where each tree is derived from the previous one by the application of a single rule from $\mathbb{P}^R$ to one of the previous tree's leaves.
- A proof state is $\mathbb{P}$**-saturated** if it is not reducible with respect to $\mathbb{P}$.
- A derivation tree is $\mathbb{P}$**-saturated** if it is not reducible with respect to $\mathbb{P}$.
- A derivation is $\mathbb{P}$**-saturated** if it ends in a $\mathbb{P}$-saturated derivation tree.

Satisfiability Proof Systems

A **satisfiability proof system** is an abstract proof system with the property that its set of proof states includes two distinguished elements, $\textsc{sat}$ and $\textsc{unsat}$.

- Rules which contain $\textsc{unsat}$ as their sole conclusion are called **refuting** rules
- Rules which contain $\textsc{sat}$ as their sole conclusion are called **satisfying** rules
- A **refutation tree** (from $\mathbb{S}$) is a derivation tree (from $\mathbb{S}$), all of whose leaves are $\textsc{unsat}$
- A **satisfied tree** (from $\mathbb{S}$) is a derivation tree (from $\mathbb{S}$), at least one of whose leaves is $\textsc{sat}$
- A **$\mathbb{P}$-refutation** (from $\mathbb{S}$) is a $\mathbb{P}$-derivation (from $\mathbb{S}$) ending with a refutation tree
- A **satisfying** $\mathbb{P}$-derivation is one ending with a satisfied tree.

A Satisfiability Proof System for Propositional Logic

How can we extend $\mathbb{P}_{PL}$ to be a satisfiability proof system?

Simply add $\textsc{sat}$ and $\textsc{unsat}$ to $\mathbb{P}_{PL}^{S}$

Let's also add a refuting rule:

$$\text{Contr} \quad \frac{p \in \mathcal{V} \quad p \in \mathbb{S} \quad \neg p \in \mathbb{S}}{\textsc{unsat}}$$

Derivation Tree Example

With our new rule, is this derivation tree saturated?

$$\text{Split}\ \dfrac{\{b \to c, \neg b \to c, \neg c\}}{\text{MP}\ \dfrac{\{b \to c, \neg b \to c, \neg c, b\}}{\{b \to c, \neg b \to c, \neg c, b, c\}} \qquad \text{MP}\ \dfrac{\{b \to c, \neg b \to c, \neg c, \neg b\}}{\{b \to c, \neg b \to c, \neg c, \neg b, c\}}}$$

Derivation Tree Example

With our new rule, is this derivation tree saturated?

$$\text{Split}\ \dfrac{\{b \rightarrow c, \neg b \rightarrow c, \neg c\}}{\text{MP}\ \dfrac{\{b \rightarrow c, \neg b \rightarrow c, \neg c, b\}}{\text{Contr}\ \dfrac{\{b \rightarrow c, \neg b \rightarrow c, \neg c, b, c\}}{\text{UNSAT}}} \qquad \text{MP}\ \dfrac{\{b \rightarrow c, \neg b \rightarrow c, \neg c, \neg b\}}{\{b \rightarrow c, \neg b \rightarrow c, \neg c, \neg b, c\}}}$$

Derivation Tree Example

With our new rule, is this derivation tree saturated?

$$
\text{Split}\;\dfrac{\{b \to c, \neg b \to c, \neg c\}}{\text{MP}\;\dfrac{\{b \to c, \neg b \to c, \neg c, b\}}{\text{Contr}\;\dfrac{\{b \to c, \neg b \to c, \neg c, b, c\}}{\text{UNSAT}}} \qquad \text{MP}\;\dfrac{\{b \to c, \neg b \to c, \neg c, \neg b\}}{\text{Contr}\;\dfrac{\{b \to c, \neg b \to c, \neg c, \neg b, c\}}{\text{UNSAT}}}}
$$

Soundness

Let $\mathbb{P} = \langle \mathbb{P}^S, \mathbb{P}^R \rangle$ be a satisfiability proof system

- A **satisfiability predicate** is a subset $\mathbb{P}^{Sat} \subseteq \mathbb{P}^S$ such that $\textsc{sat} \in \mathbb{P}^{Sat}$ and $\textsc{unsat} \notin \mathbb{P}^{Sat}$
- $\mathbb{P}^{Sat}$ is also called the set of **satisfiable proof states**
- $\mathbb{P}$ is **refutation sound** with respect to $\mathbb{P}^{Sat}$ if whenever there exists a $\mathbb{P}$-refutation from $\mathbb{S}$, we have $\mathbb{S} \notin \mathbb{P}^{Sat}$
- $\mathbb{P}$ is **solution sound** with respect to $\mathbb{P}^{Sat}$ if whenever there exists a satisfying $\mathbb{P}$-derivation from $\mathbb{S}$, we have $\mathbb{S} \in \mathbb{P}^{Sat}$
- $\mathbb{P}$ is **sound** with respect to $\mathbb{P}^{Sat}$ if it is both refutation sound and solution sound with respect to $\mathbb{P}^{Sat}$

Soundness

Let $\mathbb{P} = \langle \mathbb{P}^S, \mathbb{P}^R \rangle$ be a satisfiability proof system and $\mathbb{P}^{Sat}$ a satisfiability predicate

- A proof rule $p \in \mathbb{P}^R$ is **satisfiability preserving** if, whenever $p(\mathbb{S}) = \{\mathbb{S}_1, \ldots \mathbb{S}_n\}$, we have $\mathbb{S} \in \mathbb{P}^{Sat}$ iff for some $i \in [1, n]$, $\mathbb{S}_i \in \mathbb{P}^{Sat}$

Theorem

$\mathbb{P}$ *is sound if each of its proof rules is satisfiability preserving*

The proof is by induction on the size of derivations (see handout for details)

Soundness Example

Consider again $\mathbb{P}_{PL} = \langle \mathbb{P}^S_{PL}, \mathbb{P}^P_{PL} \rangle$

Let $\mathbb{P}^{Sat}_{PL} = \{\mathbb{S} \in \mathbb{P}^S_{PL} \mid \mathbb{S} \in \mathcal{P}(\mathcal{W}) \text{ and } \mathbb{S} \text{ is propositionally satisfiable } \} \cup \{\textsc{sat}\}$

Are these rules satisfiability preserving?

$$\text{MP} \quad \frac{p, q \in \mathcal{L} \quad p \in \mathbb{S} \quad p \to q \in \mathbb{S} \quad q \notin \mathbb{S}}{\mathbb{S} \cup \{q\}} \qquad \text{Contr} \quad \frac{p \in \mathcal{V} \quad p \in \mathbb{S} \quad \neg p \in \mathbb{S}}{\textsc{unsat}}$$

$$\text{Split} \quad \frac{p \in \mathcal{V} \quad p \text{ occurs in some formula in } \mathbb{S} \quad p \notin \mathbb{S} \quad \neg p \notin \mathbb{S}}{\mathbb{S} \cup \{p\} \quad \mathbb{S} \cup \{\neg p\}}$$

Soundness Example

Is $\mathbb{P}_{PL}$ sound with respect to $\mathbb{P}_{PL}^{Sat}$?

Yes!

Is this rule satisfiability preserving?

$$\text{Add-Var} \quad \frac{p \in \mathcal{V} \quad p \notin \mathbb{S} \quad \neg p \notin \mathbb{S}}{\mathbb{S} \cup \{p\}}$$

Completeness and Termination

Let $\mathbb{P}$ be a satisfiability proof system

- $\mathbb{P}$ is **complete** if for every $\mathbb{S} \in \mathbb{P}^S$, there exists either a satisfying $\mathbb{P}$-derivation or a refutation from $\mathbb{S}$.
- $\mathbb{P}$ is **terminating** if every $\mathbb{P}$-derivation is finite

Completeness and Termination

$$\text{MP} \quad \frac{p, q \in \mathcal{L} \quad p \in \mathbb{S} \quad p \to q \in \mathbb{S} \quad q \notin \mathbb{S}}{\mathbb{S} \cup \{q\}} \qquad \text{Contr} \quad \frac{p \in \mathcal{V} \quad p \in \mathbb{S} \quad \neg p \in \mathbb{S}}{\text{UNSAT}}$$

$$\text{Split} \quad \frac{p \in \mathcal{V} \quad p \text{ occurs in some formula in } \mathbb{S} \quad p \notin \mathbb{S} \quad \neg p \notin \mathbb{S}}{\mathbb{S} \cup \{p\} \quad \mathbb{S} \cup \{\neg p\}}$$

Is $\mathbb{P}_{PL}$ terminating?

Yes!

How would you prove it?

Completeness and Termination

$$\text{MP} \quad \frac{p, q \in \mathcal{L} \quad p \in \mathbb{S} \quad p \rightarrow q \in \mathbb{S} \quad q \notin \mathbb{S}}{\mathbb{S} \cup \{q\}} \qquad \text{Contr} \quad \frac{p \in \mathcal{V} \quad p \in \mathbb{S} \quad \neg p \in \mathbb{S}}{\text{UNSAT}}$$

$$\text{Split} \quad \frac{p \in \mathcal{V} \quad p \text{ occurs in some formula in } \mathbb{S} \quad p \notin \mathbb{S} \quad \neg p \notin \mathbb{S}}{\mathbb{S} \cup \{p\} \quad \mathbb{S} \cup \{\neg p\}}$$

Is $\mathbb{P}_{PL}$ complete?

No!

Can you find a state that is not reducible?

How about $\{b\}$?

Proof Systems and Decision Procedures

If $\mathbb{P}$ is sound with respect to $\mathbb{P}^{Sat}$, complete, and terminating, it induces a **decision procedure** for deciding whether $\mathbb{S} \in \mathbb{P}^{Sat}$

- Simply start with $\mathbb{S}$ and produce any derivation
- It must eventually terminate
- If the final tree is a refutation tree, then $\mathbb{S} \notin \mathbb{P}^{Sat}$
- otherwise, $\mathbb{S} \in \mathbb{P}^{Sat}$

A Decision Procedure for Propositional Logic

Recall that a **variable assignment** v is a mapping from $\mathcal{V}$ to $\{0, 1\}$, and $v \models \mathbb{S}$ means that each formula in $\mathbb{S}$ evaluates to true under the variable assignment v

Let $\mathbb{S}$ be a set of propositional formulas. The **variable assignment** v **induced by** $\mathbb{S}$ is defined as follows:

$$v(p) = \begin{cases} 1 & \text{if } p \in \mathbb{S} \\ 0 & \text{if } \neg p \in \mathbb{S} \\ 0 & \text{otherwise} \end{cases}$$

$\mathbb{S}$ **fully defines** v if v is the variable assignment induced by $\mathbb{S}$ and for each propositional variable p occurring in $\mathbb{S}$, either $p \in \mathbb{S}$ or $\neg p \in \mathbb{S}$.

A Decision Procedure for Propositional Logic

Let $\mathbb{P}_{Enum} = \langle \mathbb{P}^S_{Enum}, \mathbb{P}^P_{Enum} \rangle$, where (as with $\mathbb{P}_{PL}$) $\mathbb{P}^S_{Enum}$ contains all sets of propositional formulas plus the distinguished states SAT and UNSAT

There are three proof rules:

$$\text{Split} \quad \frac{p \in \mathcal{V} \quad p \text{ occurs in some formula in } \mathbb{S} \quad p \notin \mathbb{S} \quad \neg p \notin \mathbb{S}}{\mathbb{S} \cup \{p\} \qquad \mathbb{S} \cup \{\neg p\}}$$

$$\text{Sat} \quad \frac{\mathbb{S} \text{ fully defines } v \quad v \models \mathbb{S}}{\text{SAT}} \qquad \text{Unsat} \quad \frac{\mathbb{S} \text{ fully defines } v \quad v \models \neg\phi \text{ for some } \phi \in \mathbb{S}}{\text{UNSAT}}$$

A Decision Procedure for Propositional Logic

Theorem

Each rule in $\mathbb{P}_{Enum}$ is satisfiability preserving with respect to $\mathbb{P}_{PL}^{Sat}$

Corollary

$\mathbb{P}_{Enum}$ is sound with respect to $\mathbb{P}_{PL}^{Sat}$

Theorem

$\mathbb{P}_{Enum}$ is terminating

Theorem

$\mathbb{P}_{Enum}$ is complete

Therefore, $\mathbb{P}_{Enum}$ can be used as a decision procedure for the SAT problem

Example

Consider the set of propositional formulas $\{a, \neg a \vee b, a \rightarrow \neg b\}$

$$\frac{\{a, \neg a \vee b, a \rightarrow \neg b\}}{}$$

Example

Consider the set of propositional formulas $\{a, \neg a \vee b, a \rightarrow \neg b\}$

$$\text{Split} \quad \frac{\{a, \neg a \vee b, a \rightarrow \neg b\}}{\{a, \neg a \vee b, a \rightarrow \neg b, b\} \quad \{a, \neg a \vee b, a \rightarrow \neg b, \neg b\}}$$

Example

Consider the set of propositional formulas $\{a, \neg a \vee b, a \rightarrow \neg b\}$

$$\text{Split}\quad \frac{\{a, \neg a \vee b, a \rightarrow \neg b\}}{\text{Unsat}\ \dfrac{\{a, \neg a \vee b, a \rightarrow \neg b, b\}}{\mathrm{UNSAT}} \qquad \{a, \neg a \vee b, a \rightarrow \neg b, \neg b\}}$$

Example

Consider the set of propositional formulas $\{a, \neg a \vee b, a \rightarrow \neg b\}$

$$\text{Split}\ \frac{\{a, \neg a \vee b, a \rightarrow \neg b\}}{\text{Unsat}\ \dfrac{\{a, \neg a \vee b, a \rightarrow \neg b, b\}}{\text{UNSAT}} \quad \text{Unsat}\ \dfrac{\{a, \neg a \vee b, a \rightarrow \neg b, \neg b\}}{\text{UNSAT}}}$$

Example

Alternatively, consider the set of propositional formulas $\{a, \neg a \vee \neg b, a \to \neg b\}$

$$\frac{\{a, \neg a \vee \neg b, a \to \neg b\}}{}$$

Example

Alternatively, consider the set of propositional formulas $\{a, \neg a \vee \neg b, a \rightarrow \neg b\}$

$$\text{Split} \quad \frac{\{a, \neg a \vee \neg b, a \rightarrow \neg b\}}{\{a, \neg a \vee \neg b, a \rightarrow \neg b, b\} \qquad \{a, \neg a \vee \neg b, a \rightarrow \neg b, \neg b\}}$$

Example

Alternatively, consider the set of propositional formulas $\{a, \neg a \vee \neg b, a \to \neg b\}$

$$
\text{Split}\ \frac{\{a, \neg a \vee \neg b, a \to \neg b\}}{\text{Unsat}\ \dfrac{\{a, \neg a \vee \neg b, a \to \neg b, b\}}{\text{UNSAT}} \qquad \{a, \neg a \vee \neg b, a \to \neg b, \neg b\}}
$$

Example

Alternatively, consider the set of propositional formulas $\{a, \neg a \vee \neg b, a \rightarrow \neg b\}$

$$\text{Split}\ \frac{\{a, \neg a \vee \neg b, a \rightarrow \neg b\}}{\text{Unsat}\ \dfrac{\{a, \neg a \vee \neg b, a \rightarrow \neg b, b\}}{\text{UNSAT}} \qquad \text{Sat}\ \dfrac{\{a, \neg a \vee \neg b, a \rightarrow \neg b, \neg b\}}{\text{SAT}}}$$

Strategies

Sometimes, a proof system does not have nice properties unless the rules are applied in a specific way

Let $\mathbb{P} = \langle \mathbb{P}^S, \mathbb{P}^R \rangle$ be a proof system

- A $\mathbb{P}$**-strategy** is a partial function that, when defined, takes a derivation tree τ and returns a new derivation tree τ' such that (τ, τ') is a $\mathbb{P}$-derivation
- A $\mathbb{P}$-derivation D **follows** a $\mathbb{P}$-strategy π if each derivation tree (after the first) in D is the result of applying π to the previous derivation tree, and, if D is finite, the final derivation tree is not in the domain of π

Strategy Example

Let $\prec$ be a total order on literals in $\mathcal{L}$ defined as alphabetical by variable name with variables smaller than their negations

Consider the following $\mathbb{P}_{PL}$-strategy π_{PL}:

1. Find the first reducible leaf (in a left-to-right depth-first traversal); if none, then π_{PL} is undefined
2. Apply MP if possible, using the smallest possible literals (according to $\prec$), first for p, then for q
3. Otherwise, if possible, apply Split, instantiating p as small as possible
4. Otherwise, apply Contr

Strategy Example

Let's apply π_{PL} to $\{a \rightarrow \neg b, \neg b \rightarrow \neg a\}$:

$$\frac{\{a \rightarrow \neg b, \neg b \rightarrow \neg a\}}{}$$

Strategy Example

Let's apply π_{PL} to $\{a \rightarrow \neg b, \neg b \rightarrow \neg a\}$:

$$\text{Split} \quad \frac{\{a \rightarrow \neg b, \neg b \rightarrow \neg a\}}{\{a \rightarrow \neg b, \neg b \rightarrow \neg a, a\} \quad \{a \rightarrow \neg b, \neg b \rightarrow \neg a, \neg a\}}$$

Strategy Example

Let's apply π_{PL} to $\{a \to \neg b, \neg b \to \neg a\}$:

$$\text{Split}\ \frac{\{a \to \neg b, \neg b \to \neg a\}}{\text{MP}\ \dfrac{\{a \to \neg b, \neg b \to \neg a, a\}}{\{a \to \neg b, \neg b \to \neg a, a, \neg b\}} \qquad \{a \to \neg b, \neg b \to \neg a, \neg a\}}$$

Strategy Example

Let's apply π_{PL} to $\{a \to \neg b, \neg b \to \neg a\}$:

$$\text{Split}\ \dfrac{\{a \to \neg b, \neg b \to \neg a\}}{\text{MP}\ \dfrac{\{a \to \neg b, \neg b \to \neg a, a\}}{\text{MP}\ \dfrac{\{a \to \neg b, \neg b \to \neg a, a, \neg b\}}{\{a \to \neg b, \neg b \to \neg a, a, \neg b, \neg a\}}} \qquad \{a \to \neg b, \neg b \to \neg a, \neg a\}}$$

Strategy Example

Let's apply π_{PL} to $\{a \to \neg b, \neg b \to \neg a\}$:

$$
\dfrac{\{a \to \neg b, \neg b \to \neg a\}}{\dfrac{\dfrac{\dfrac{\{a \to \neg b, \neg b \to \neg a, a\}}{\{a \to \neg b, \neg b \to \neg a, a, \neg b\}}\text{MP}}{\{a \to \neg b, \neg b \to \neg a, a, \neg b, \neg a\}}\text{MP}}{\text{UNSAT}}\text{Contr} \qquad \{a \to \neg b, \neg b \to \neg a, \neg a\}}\text{Split}
$$

Strategy Example

Let's apply π_{PL} to $\{a \to \neg b, \neg b \to \neg a\}$:

$$\text{Split}\ \dfrac{\{a \to \neg b, \neg b \to \neg a\}}{\text{MP}\ \dfrac{\{a \to \neg b, \neg b \to \neg a, a\}}{\text{MP}\ \dfrac{\{a \to \neg b, \neg b \to \neg a, a, \neg b\}}{\text{Contr}\ \dfrac{\{a \to \neg b, \neg b \to \neg a, a, \neg b, \neg a\}}{\textsc{UNSAT}}}} \qquad \text{Split}\ \dfrac{\{a \to \neg b, \neg b \to \neg a, \neg a\}}{\{a \to \neg b, \neg b \to \neg a, \neg a, b\} \quad \{a \to \neg b, \neg b \to \neg a, \neg a, \neg b\}}}$$

Properties of Strategies

Let $\mathbb{P}^{Sat}$ be a satisfiability predicate for $\mathbb{P}$.

- A $\mathbb{P}$-strategy π is **refutation sound** with respect to $\mathbb{P}^{Sat}$ if whenever there exists a $\mathbb{P}$-refutation from $\mathbb{S}$ following π, we have $\mathbb{S} \in \mathbb{P}^{Sat}$
- A $\mathbb{P}$-strategy π is **solution sound** with respect to $\mathbb{P}^{Sat}$ if whenever there exists a satisfying $\mathbb{P}$-derivation from $\mathbb{S}$ following π, we have $\mathbb{S} \in \mathbb{P}^{Sat}(\mathbb{S})$
- A $\mathbb{P}$-strategy is **sound** with respect to $\mathbb{P}^{Sat}$ if it is both refutation sound and solution sound with respect to $\mathbb{P}^{Sat}$
- A $\mathbb{P}$-strategy π is **terminating** if every $\mathbb{P}$-derivation following π is finite
- A $\mathbb{P}$-strategy π is **progressive** if π is defined for every derivation tree that is not a refutation tree or a satisfied tree.

Properties of Strategies

Let $\mathbb{P}^{Sat}$ be a satisfiability predicate for $\mathbb{P}$.

If $\mathbb{P}$ is sound with respect to $\mathbb{P}^{Sat}$, then every $\mathbb{P}$-strategy is also sound with respect to $\mathbb{P}^{Sat}$

If $\mathbb{P}$ is terminating, then every $\mathbb{P}$-strategy is also terminating

Theorem

$\mathbb{P}$ is complete iff there exists a progressive and terminating strategy for it