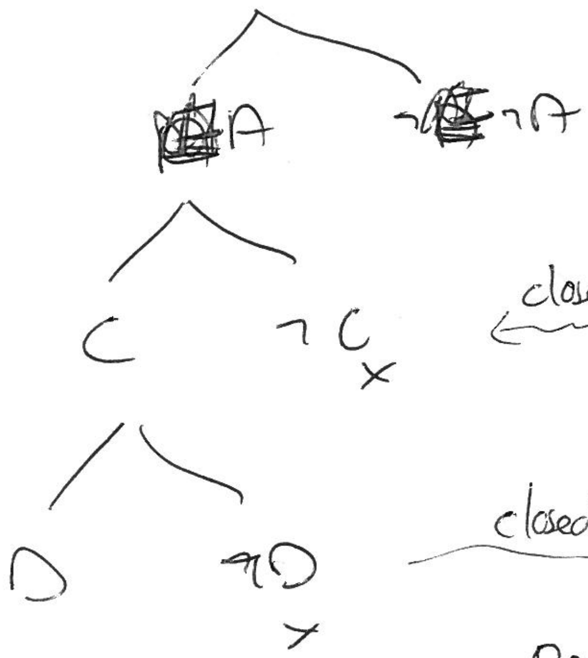
# Lemma Learning

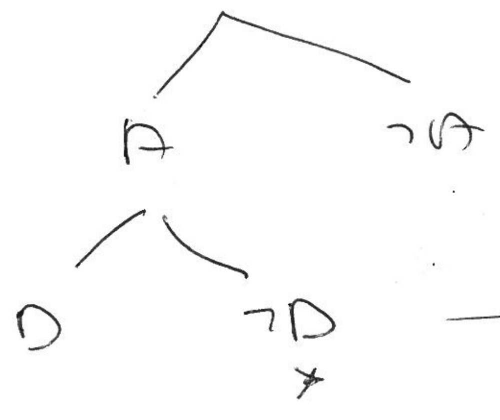



closed by ← $\cancel{\neg A \vee C}$ $C \vee \neg A$

closed by $D \vee \neg C \vee \neg A$

Resolution ("Cut")
$$\frac{D \vee \neg C \vee \neg A \qquad C \vee \neg A}{D \vee \neg A}$$

allows to derive equivalent tree earlier:




⟶ closed by $D \vee \neg A$

Translation of DLs into FO logic

T-Box:

$$\forall x \, (\text{Professor}(x) \wedge \exists y \, (\text{supervises}(x,y) \wedge \text{Student}(y)) \rightarrow \text{BusyPerson}(x))$$

"general"

Now use any First-Order ~~FO~~ logic Theorem Prover.

Problems:

- Efficiency
- It might not terminate at all

Research challenge:

- Find refinements ~~of~~ that avoid problems

# Conclusions

3

"Parameter Space":

- Logic: Propositional – Description Logic – First-Order
- Theories: "all" models – specific models (integers...)
- Services: Theorem Proving (validity) – Disproving (satisfiability) – Abduction

  $$\frac{A \rightarrow B \quad B}{A}$$

Challenge: Build a Theorem Prover that ~~matches~~ supports "best" your application

# Freshness requirement

$$(\exists x\, P(x)) \wedge \neg P(x)$$ (satisfiable)

$$\not\equiv$$

$$\exists x\, (P(x) \wedge \neg P(x))$$ (unsatisfiable)

But

$$(\exists x\, P(x)) \wedge \neg P(x)$$

$$\equiv$$

$$\exists y\, (P y \wedge \neg P x)$$

# Skolemization notes

(1) Not an <u>equivalence</u> Transformation

Take $\mathbb{N} \cup \{ \text{"}f(x)=x\text{"} \} \models \forall x \exists y \; x<y$

but $\mathbb{N} \cup \{ \text{"}f(x)=x\text{"} \} \not\models \forall x \; x<f(x)$

(The converse does hold)

(2) "Miniscoping" improvement

$$\forall x \forall y \exists z \quad P(x,y) \wedge Q(x,z)$$

"official"

$$\forall x \forall y P(x,y) \wedge Q(x, \underline{f(y,z)})$$

$$\equiv$$

$$\forall y \, ( \underline{\cancel{\forall x P(x,y)}} \wedge \exists z \, Q(x,z))$$

$$\equiv$$

$$\forall x \, ((\forall y \, P(x,y)) \wedge (\exists z \, Q(x,z))$$

$$\equiv$$

$$\forall x \exists z \forall y \, (P(x,y) \wedge Q(x,z)$$

$$\rightsquigarrow$$

$$\forall x \forall y \, (P(x,y) \wedge Q(x, \overline{f(x)}))$$

"Better"?

# Duplization of subformulas

$$\varphi: (A_1 \wedge \ldots \wedge A_m) \vee (B_1 \wedge \ldots \wedge B_n)$$

$\Downarrow$ CNF conversion

$$\underbrace{(A_1 \vee B_1) \wedge (A_1 \vee B_2) \wedge \ldots \wedge (A_m \vee B_n)}_{m \times n \text{ clauses}}$$

In general exponential blowup!

Solution: Linear transformation by "naming" subformulas:

~~$A \Leftrightarrow (A$~~

Transform $\varphi$ into:

"fresh" $\rightarrow$ $A \Leftrightarrow (A_1 \wedge \ldots \wedge A_m)$ $\rightarrow$ $m+1$ clauses disj

$\wedge$ $B \Leftrightarrow (B_1 \wedge \ldots \wedge B_n)$ $\rightarrow$ $n+1$ clause disj

$\wedge$ $A \vee B$ 1 clause disj

$\Sigma = m+n+3$ clauses disj.

(a little more complicated for FOL, but idea ~~remains~~ the same)

# Constructing a Herbrand Interpretation $\mathcal{I}$ from an Interpretation (by example)

Recall $U_{\mathbb{N}} = \{0, 1, 2 \ldots\}$

$\leq_{\mathbb{N}} = \{(0,0), (0,1) \ldots (1,1), (1,2) \ldots\}$

$s_{\mathbb{N}} = \lambda x .\, x+1$

$\leadsto$ ~~Herbrand Interpretation for $\Sigma_{Pres}$~~

Take any ground ~~Term~~ ~~Atom~~: $\Sigma_{Pres}$ Atom

e.g. $A = (\, s(0) \leq s(0) + s(0) \,)$

$$\underbrace{s(0)}_{\text{eval}} \leq \underbrace{s(0)+s(0)}_{\text{eval}}$$

$(1 \;,\; 2) \in \leq_{\mathbb{N}} \Rightarrow A \in \mathcal{I}$

~~Therefore~~ set

$A = (\, s(0) \leq 0 \,)$

$(1, \; 0) \notin \leq_{\mathbb{N}} \Rightarrow A \notin \mathcal{I}$

$\leadsto$ ~~I from~~ ~~Construction~~ $\mathcal{I}$ "says the same" as $U_{\mathbb{N}}$ ~~&~~ w.r.t. predicates

# Herbrand's Theorem – How to prove it

- Indirect: $\forall\varphi$ is unsatisfiable

$$\Updownarrow \quad \cancel{\exists\neg\varphi \text{ is}} \models \neg\forall\varphi$$

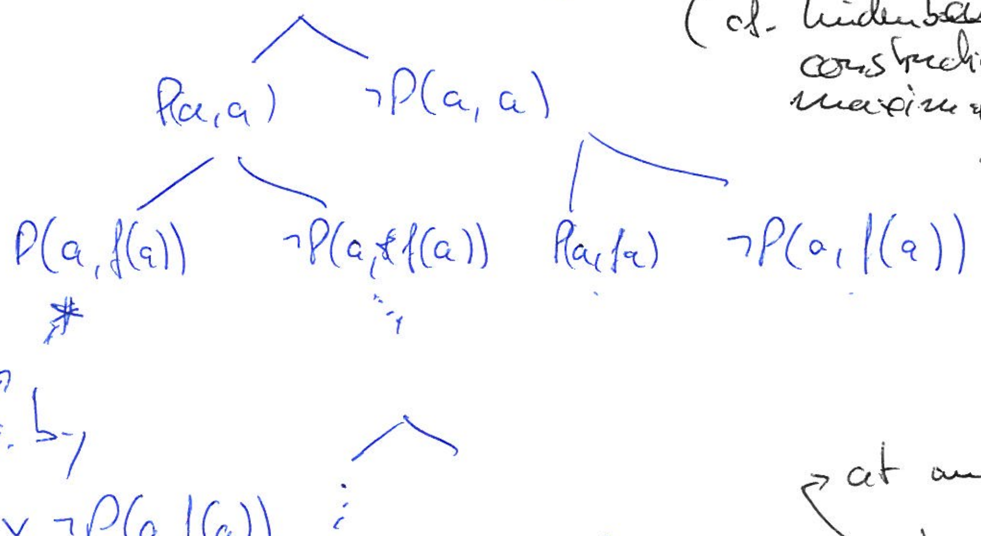
$$\Updownarrow \quad \cancel{\vdash \neg\forall\varphi} \vdash \exists\neg\varphi$$

completeness, soundness of ND

<u>Proof</u> is a finite object – ~~Use~~ use witnesses ~~of~~ terms for $\exists$-variables ~~define the~~ to ~~de fird~~ determine the ground instances of $\varphi$

- Direct (clause logic)
  ~~to~~ show: Every finite set of ground instances of $N$ is satisfiable $\Rightarrow$ $N$ is satisfiable

  Proof idea: enumerate interpretations, e.g. (cf. Lindenbaum construction, maximal consistent sets)




"closed" e.g. by
$\neg P(a,a) \vee \neg P(a,f(a))$

At least one non-closed branch must exist (limit) ~~always~~ at any stage (otherwise some finite set of ground instances is unsatisfiable) which defines a model for $N$

~~Soundness and (Refutational) Completeness~~

Given: ~~N, set of clauses~~

## Soundness of Resolution

To show:

~~For any clause set N:~~

If Prop. Resolution derives the empty clause $\bot$ from $N$ then $N$ is unsatisfiable

Equivalently:

If $N$ is satisfiable then Prop. Resolution does not derive $\bot$

~~It suffices to show that derived clauses are consequences of parent clauses~~

Proof idea:

- Assume $\mathcal{I} \models N$
- Show $\mathcal{I} \models N \cup \{C\}$ for any derived clause $C$

  (It follows $C \neq \bot$)
- Induction on length of derivation

# Refutational (!) completeness of Prop. Resolution

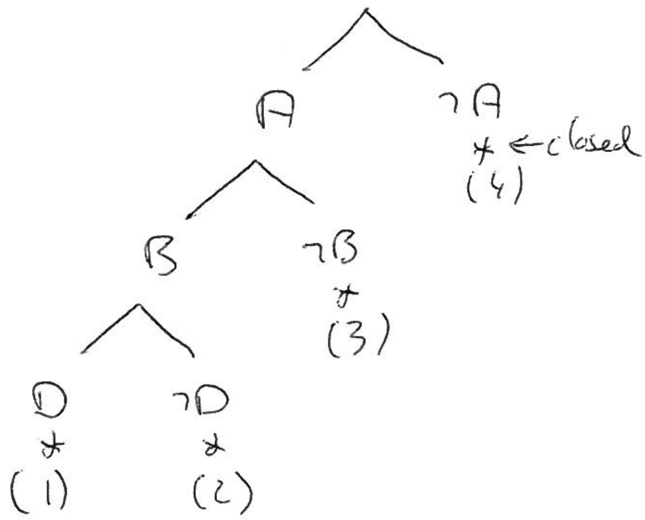
To show: $N$ is unsatisfiable $\Rightarrow$ Resolution derives $\bot$

Proof by "semantic tree method"

Example:

$$N = \{ \underset{(1)}{\neg A \vee \neg B \vee \neg D}, \quad \underset{(2)}{\neg A \vee D}, \quad \underset{(3)}{\neg A \vee B}, \quad \underset{(4)}{A} \}$$

Semantic Tree $T$ for $\{A, B, D\}$:

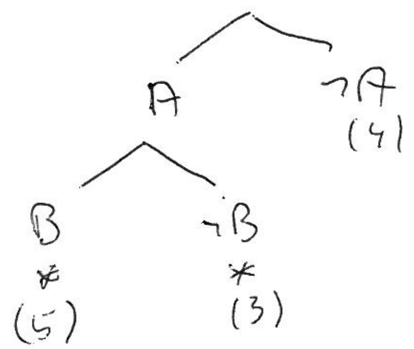



Fact: $N$ is unsatisfiable $\Rightarrow$ $T$ is closed

Read off Resolution inferences

$$\frac{(1) \qquad (2)}{\neg A \vee \neg A \vee \neg B} \quad =: (5) \text{ new clause}$$

New (smaller) semantic tree $T'$ for (1) - (5):




Repeat this until root-only tree derived (gives $\bot$)

# Resolution Strategies

(for better efficiency)

- Ordered Resolution
  - Ordering $A > B > D$
  - Resolution and Factoring on maximal literals only

$$\frac{\underline{\neg A} \vee B \qquad \underline{A} \vee \cancel{B}\, C}{B \vee C} \quad \text{ok}$$

$$\frac{\underline{\neg A} \vee B \qquad \neg B \vee D}{\neg A \vee D} \quad \underline{\text{not}}\ \text{ok}$$

(B not maximal)

  - Completeness: "easy" - use semantic tree according to ordering (smaller literals towards root)

- Hyperresolution

only positive literals

all negative lits resolved away ~~in~~ in one step

$$\frac{\underline{A} \vee B \qquad \underline{C} \vee D \qquad \neg \underline{A} \vee \neg \underline{C} \vee E}{B \vee D \vee E}$$

very effective for Horn clauses (linear)

- Linear Resolution (for Horn Klauses) a la "Prolog"

$?- A, B$ "Query" "Prove A and B" $\neg A \vee \neg B$ (1)

$A \Leftarrow C$ "Program" $A \vee \neg C$ (2)

$C \Leftarrow B$ $C \vee \neg B$ (3)

$B.$ $B$ (4)

$\downarrow$ Backward Chaining

~~$?- C, B$~~

~~$?- B, B$~~

| | | |
|---|---|---|
| $?- A, B$ | $\neg A \vee \neg B$ | (1) |
| $?- C, B$ | $\neg C \vee \neg B$ | + (2) |
| $?- B, B$ | $\neg B \vee \neg B$ | + (3) |
| $?- B$ | $\neg B$ | + (4) |
| $?-.$ | $\perp$ | + (4) |

# Unifiers



$$\begin{tikzcd}
P(f(x), y) \arrow[dr, "\gamma"'] & & P(z, z) \arrow[dl] \\
& P(f(a), f(a)) &
\end{tikzcd}$$

$\gamma$ unifier, but not most general one

$P(f(a), f(a))$ common instance

$$\begin{tikzcd}
P(f(x), y) \arrow[dr, "\sigma"] & & P(z, z) \arrow[dl] \\
& P(f(x), f(x)) \arrow[d, "\delta"] & \\
& P(f(a), f(a)) &
\end{tikzcd}$$

$\gamma = \sigma \circ \delta$

# ~~Lifting~~ Completeness of FO-Resolution

13

$$\begin{tikzcd}
N \text{ unsatisfiable} \arrow[rr] \arrow[d, squiggly, "\text{Herbrand Theorem}"] & & N \vdash \bot \\
N' \text{ unsatisfiable} \arrow[rr, "~~\text{Bo}~~"', "\text{Completeness Prop. Resolution}"'] & & N' \vdash \bot \arrow[u, squiggly, "\text{Lifting}"']
\end{tikzcd}$$

finite set of ground instances

## Lifting

$$\begin{array}{lccccccc}
 & \overbrace{\phantom{C_1 \ldots C_k}}^{\in N} & & \xrightarrow[\text{By FO-Resolution}]{③} & & & & \\
\text{"Lifted"} & C_1 \ldots C_k & & C_{k+1}, & C_{k+2}, & \ldots & C_n\ (=\bot) \\
 & \downarrow \sigma \quad ① \downarrow & & \downarrow \sigma & \downarrow & & \\
\text{Given:} & \underbrace{C'_1 \ldots C'_k}_{=N'}, & & C'_{k+1}, & C'_{k+2} & \ldots & C'_n\ (=\bot) \\
 & & & \xrightarrow[\text{By Prop. Res}]{②} & & & &
\end{array}$$

~~bra~~

~~The~~ <u>To show:</u> Prop. Resolution steps ② ~~simulated by~~ are instances of FO-Resolution steps ③ with more general clauses

# Equality (I)

"=" – Predicate with special meaning

E.g. $P(a) \wedge a=b \wedge c=b \wedge \neg P(c)$

is E-unsatisfiable

How to prove that?

1. Add "Equality axioms" (~~all~~ each universally quantified)

Equiv. relation:

$$\begin{cases} x=x & \text{(Ref)} \\ x=y \rightarrow y=x & \text{(Sym)} \\ x=y \wedge y=z \rightarrow x=z & \text{(Trans)} \end{cases}$$

~~Cong~~ Substitutivity:

$x=y \rightarrow f(\ldots x \ldots) = f(\ldots y \ldots)$ for each function symbol

e.g. $x=y \rightarrow f(z,x) = f(z,y)$

$x=y \wedge P(\ldots x \ldots) \rightarrow P(\ldots y \ldots)$ for each predicate symbol $P$

**Problem**: Search space explodes

# Equality (2)

2. ~~Paramodulation~~ Special inference rules

- Paramodulation (much better) inference

$$\frac{s = t \vee C \qquad L[s'] \vee D}{(L\sigma[t\sigma] \vee C\sigma \vee D\sigma)}$$

where $\sigma = mgu(s, s')$

Example:

$$\frac{\underline{f(x,a)} = x \vee P(x) \qquad Q(\underline{f(b,y)}) \vee R(y)}{Q(b) \vee R(a)}$$

$\sigma = \{x \to b, y \to a\}$

Important Improvements

- $s'$ is not a variable
- $s\sigma \not\leq t\sigma$ for some well-founded term-ordering $\leq$

- add clause $x = x$ (harmless)

# Set of Support

14a

| SOS | USABLE | given clause |
|---|---|---|
| $\underline{\neg A}$<br>$A \vee B$<br>$\neg B \vee C$ | $\emptyset$ | |
| $\underline{A \vee B}$<br>$\neg B \vee C$ | $\neg A$ | |
| $B$<br>$\underline{\neg B \vee C}$ | $\neg A$<br>$A \vee B$ | |
| $B$<br>$A \vee C$ | $\neg A$<br>$A \vee B$<br>$\neg B \vee C$ | |

## Retention Test:

Most important: "subsumption"

Def.: Clause $C$ <u>subsumes</u> $D$

iff $C\sigma \subseteq D$

E.g. $Px \vee Q\cancel{y}$ subsumes

$Pa \vee Qb \vee Rc$

subsumed clauses can (usually) be deleted

# Inst-Gen Inference Rule



on slide:
$$\frac{\neg \underline{P(a)} \qquad \underline{P(x)} \vee Q(x)}{\neg P(a) \qquad P(a) \vee Q(a)} \; mgu\{x \to a\}$$

general
$$\frac{C \vee A \qquad D \vee \neg B}{(C \vee A)\sigma \quad (D \vee B)\sigma} \quad \sigma = mgu(A, B)$$

Compare with Resolution!

Differences?

# Nelson-Oppen congruence closure



To decide E-satisfiability of a set of ground unit (dis.) equations

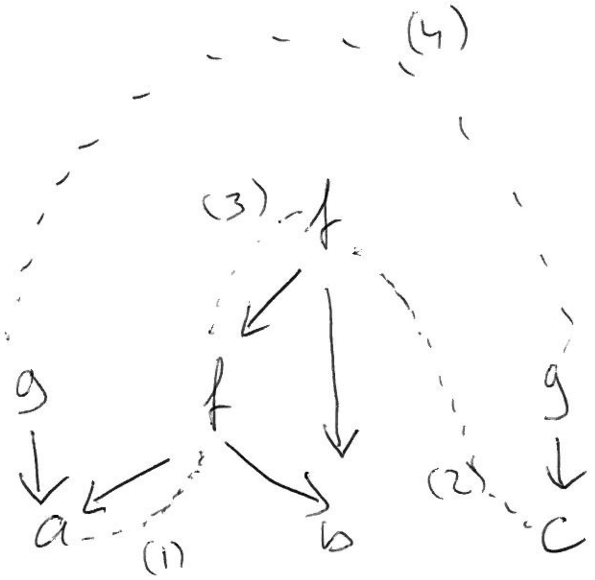
Example

(1) $f(a,b) = a$

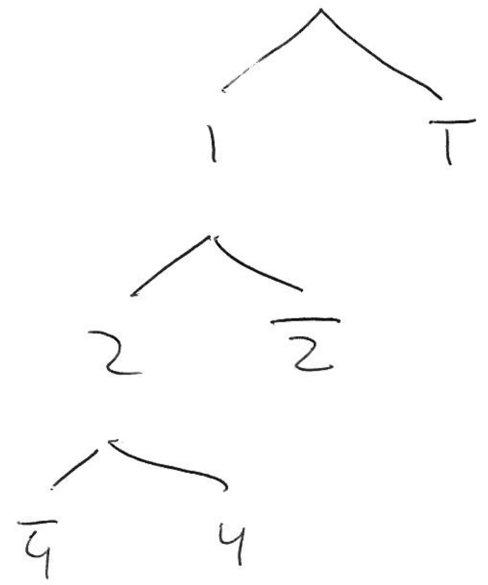
(2) $f(f(a,b),b) = c$

(3) $g(a) \neq g(c)$

NO avoids building the full (infinite!) congruence closure:




- Build term graph of input terms "⟶"
- Add congruences for input equations ". . . ." (1) (2)
- Close under congruence: Add . . . . Between terms with ~~equal sub~~ congruent subterms (3) (4)
- Check if s and t are congruent for some $s \neq t$ (yes)




model $\{1, 2, \bar{4}\}$