RESOLUTION

1. What we'd like are some <u>sound</u> rewrite rules so that if α can be rewritten as β according to those rules (written $\alpha \vdash \beta$), then $\alpha \models \beta$.
Then we can show sentence α is unsatisfiable by showing:

$$\alpha \vdash \alpha_1 \vdash \dots \vdash \alpha_k \vdash \text{False}$$

since that means:

$$\alpha \models \alpha_1 \models \dots \models \alpha_k \models \text{False}$$

and thus:

$$\mathbb{I}(\alpha) \subseteq \mathbb{I}(\alpha_1) \subseteq \dots \subseteq \mathbb{I}(\alpha_k) \subseteq \mathbb{I}(\text{False}) = \emptyset$$

implies $\mathbb{I}(\alpha) = \emptyset$.

---

2. Consider the example CNF sentence α:

$$(\neg P \vee \neg F) \wedge (P \vee B) \wedge F \wedge \neg B$$

As before, we can view each clause as a constraint on any model $m \in \mathbb{I}(\alpha)$:

	m needs to assign
$\neg P \vee \neg F$	$P \to 0$ or $F \to 0$
$P \vee B$	$P \to 1$ or $B \to 1$
F	$F \to 1$
$\neg B$	$B \to 0$

RESOLUTION

③ If we look at the first two clauses:

$\neg P \vee \neg F$ — m needs to assign $P \to 0$ or $F \to 0$

$P \vee B$ — $P \to 1$ or $B \to 1$

P is either going to be 0 or 1... ...if it's 0, then we need $B \to 1$ ...if it's 1, then we need $F \to 0$

We can infer that:

$$\alpha \models \neg F \vee B$$

Note that this also means:

$$\alpha \models (\neg F \vee B) \wedge \alpha$$

because for any sentence γ

$\alpha \models \gamma$ implies $I(\alpha) \subseteq I(\gamma)$

implies $I(\alpha) \subseteq I(\gamma) \cap I(\alpha)$

implies $\alpha \models \gamma \wedge \alpha$

---

④ We can continue this process:

$$(\neg P \vee \neg F) \wedge (P \vee B) \wedge F \wedge \neg B$$

$$\models (\neg P \vee \neg F) \wedge (P \vee B) \wedge F \wedge \neg B \wedge (\neg F \vee B)$$ [by "resolving" $\neg P \vee \neg F$ and $P \vee B$]

$$\models (\neg P \vee \neg F) \wedge (P \vee B) \wedge F \wedge \neg B \wedge (\neg F \vee B) \wedge B$$ [by "resolving" F and $\neg F \vee B$]

$$\models (\neg P \vee \neg F) \wedge (P \vee B) \wedge F \wedge \neg B \wedge (\neg F \vee B) \wedge B \wedge \text{False}$$ [by "resolving" $\neg B$ and B]

$$\models \text{False}$$

And prove that α is unsatisfiable.

RESOLUTION

(5) The general version of this rewrite (called the Resolution Rule) can be defined in the base case as:

$$l_1 \wedge \bar{l}_1 \wedge \beta \vdash \text{False}$$

and in the general case as

$$(l_1 \vee \cdots \vee l_i \vee \cdots \vee l_m) \wedge (l_1' \vee \cdots \vee l_j' \vee \cdots \vee l_n) \wedge \beta$$
$$\vdash (l_1 \vee \cdots \vee l_{i-1} \vee l_{i+1} \vee \cdots \vee l_m \vee l_1' \vee \cdots \vee l_{j-1}' \vee l_{j+1}' \vee \cdots \vee l_n) \wedge \beta$$

for literals $l_1, \ldots, l_m, l_1', \ldots, l_n' \in \text{LITERALS}(\Sigma)$ s.t. $l_i = \overline{l_j'}$ and arbitrary sentence $\beta \in \mathcal{L}(\Sigma)$.

(6) Soundness Thm: If $\alpha \vdash \gamma$, then $\alpha \models \gamma$.

Proof: (base) $\mathrm{II}(l_1 \wedge \bar{l}_i \wedge \beta) = \mathrm{II}(l_i) \cap \mathrm{II}(\bar{l}_1) \cap \mathrm{II}(\beta) = \emptyset = \mathrm{II}(\text{False})$

so $l_1 \wedge \bar{l}_1 \wedge \beta \models \text{False}$

(general) $\mathrm{II}\big((l_1 \vee \cdots \vee l_i \vee \cdots \vee l_m) \wedge (l_i' \vee \cdots \vee l_j' \vee \cdots \vee l_n) \wedge \beta\big)$

$= \mathrm{II}(l_1 \vee \cdots \vee l_i \vee \cdots \vee l_m) \cap \mathrm{II}(l_1' \vee \cdots \vee l_j' \vee \cdots \vee l_n) \cap \mathrm{II}(\beta)$

$= \big(\mathrm{II}(l_i) \cup \mathrm{II}(l_1 \vee \cdots \vee l_{i-1} \vee l_{i+1} \vee \cdots \vee l_m)\big)$

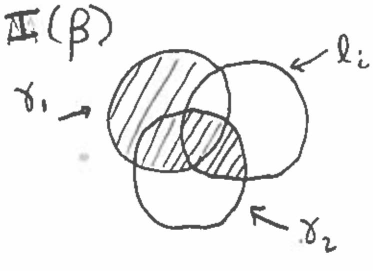
$\cap \big(\mathrm{II}(l_j') \cup \mathrm{II}(l_1' \vee \cdots \vee l_{j-1}' \vee l_{j+1}' \vee \cdots \vee l_n)\big)$

$\cap\, \mathrm{II}(\beta)$

$= \big(\mathrm{II}(l_i) \cup \mathrm{II}(\gamma_1)\big) \cap \big(\mathrm{II}(\bar{l}_i) \cup \mathrm{II}(\gamma_1')\big) \cap \mathrm{II}(\beta)$

$\subseteq \big(\mathrm{II}(\gamma_1) \cup \mathrm{II}(\gamma_1')\big) \cap \mathrm{II}(\beta)$

$= \mathrm{II}\big((\gamma_1 \vee \gamma_1') \wedge \beta\big)$



⑥ So that means, if we can find a sequence of rewrites

$$\alpha \vdash \alpha_1 \vdash \ldots \vdash \alpha_k \vdash \text{False}$$

then

$$\alpha \models \text{False}$$

But is it also true that if we <u>can't</u> find a sequence of rewrites s.t. $\alpha \vdash^* \text{False}$, then we can conclude $\alpha \not\models \text{False}$ (hence α is satisfiable)?

⑦ Surprisingly, yes. To show this, first define the <u>resolution closure</u> of a set of clauses S as the smallest set $RC(S)$ such that:

- $c \in S \Rightarrow c \in RC(S)$
- $c_1, c_2 \in RC(S)$ and $c_1 \wedge c_2 \vdash c \Rightarrow c \in RC(S)$

In other words, $RC(S)$ is the set of clauses you can derive through repeated application of the Resolution Rule.

⑧ For instance,

if $S = \{A \vee B, \neg B \vee \neg C\}$ then $RC(S) = \{A \vee B, \neg B \vee \neg C, A \vee \neg C\}$

if $S = \{A \vee B, B \vee \neg C\}$ then $RC(S) = S$.

RESOLUTION

⑨ Completeness Thm: If CNF sentence $c_1 \wedge \cdots \wedge c_N$ is unsatisfiable, then $\text{False} \in RC(\{c_1, \ldots, c_N\})$

Proof:

(i) We'll show the contrapositive, i.e. if $\text{False} \notin RC(S)$ for $S = \{c_1, \ldots, c_N\}$, then $c_1 \wedge \cdots \wedge c_N$ is satisfiable.

Let $\Sigma = \{\sigma_1, \ldots, \sigma_M\}$

$\boxed{\text{Assume False} \notin RC(S).}$

(ii) Let's construct a conjunction $\ell_1 \wedge \cdots \wedge \ell_M$ of literals s.t. $\ell_1 \wedge \cdots \wedge \ell_M \vDash c$ for every clause $c \in RC(S)$.

σ_m or $\neg\sigma_m$

If we can do that, then

$$\ell_1 \wedge \cdots \wedge \ell_M \vDash c_1 \wedge \cdots \wedge c_N$$

so $I(\ell_1 \wedge \cdots \wedge \ell_M) \subseteq I(c_1 \wedge \cdots \wedge c_N)$, and since $I(\ell_1 \wedge \cdots \wedge \ell_M)$ is nonempty, therefore $I(c_1 \wedge \cdots \wedge c_N)$ is nonempty. $\therefore$ $c_1 \wedge \cdots \wedge c_N$ is satisfiable.

Examples:

(i) Let: $\Sigma = \{A, B, C\}$

$S = \{c_1, c_2\}$

where: c_1 is $\neg B \vee \neg C$

c_2 is $\neg A \vee C$

Thus:

$$RC(S) = \{\neg B \vee \neg C, \neg A \vee C, \neg A \vee \neg B\}$$

(ii) if we construct

$$A \wedge \neg B \wedge C$$

then:

$$A \wedge \neg B \wedge C \vDash \neg B \vee \neg C$$

$$A \wedge \neg B \wedge C \vDash \neg A \vee C$$

$$A \wedge \neg B \wedge C \vDash \neg A \vee \neg B$$

RESOLUTION

(9) (cont.)

(iii) We'll initialize $l_0 = \text{True}$.

For $m = 1$ to M, set $l_m \in \{\sigma_m, \neg\sigma_m\}$ (if possible) such that:

$$l_0 \wedge \cdots \wedge l_m \not\models \neg c \text{ for each clause } c \in RC(S)$$

otherwise set $l_m = \sigma_m$.

(iv) For the sake of contradiction:

Assume m is the first iteration at which $l_0 \wedge \cdots \wedge l_m \models \neg c$ for some clause $c \in RC(S)$.

We know $m > 0$, because $\text{True} \not\models \neg c$ for all c except False, which is not in $RC(S)$ by the premise in (i)

(v) At iteration m, there exist clauses $c_1, c_2 \in RC(S)$ s.t.

$$l_0 \wedge \cdots \wedge l_{m-1} \wedge \sigma_m \models \neg c_1$$
$$l_0 \wedge \cdots \wedge l_{m-1} \wedge \neg\sigma_m \models \neg c_2$$

These clauses must have the forms:

$c_1 : \; c_1' \vee \neg\sigma_m$

$c_2 : \; c_2' \vee \sigma_m$

Otherwise:

$$l_0 \wedge \cdots \wedge l_{m-1} \models \neg c_1$$
$$l_0 \wedge \cdots \wedge l_{m-1} \models \neg c_2$$

which violates the assumption that m is the earliest iteration s.t. $l_0 \wedge \cdots \wedge l_m \models \neg c$ for $c \in RC(S)$

(iii) $l_0 = \text{True}$

$l_1 = A$

$l_2 = \neg B$ (otherwise $A \wedge B \models \neg(\neg A \vee \neg B)$)

$l_3 = C$ (otherwise $A \wedge \neg B \wedge \neg C \models \neg C_2$)

(v) say at iteration 3, we have $l_0 = \text{True}$

$l_1 = A$

$l_2 = B$

then:

$\neg B \vee \neg C$

$\text{True} \wedge A \wedge B \wedge C \models \neg C_1$

$\text{True} \wedge A \wedge B \wedge \neg C \models \neg C_2$

$\neg A \vee C$

9 (cont.)

(vi) $\boxed{\text{Since } c_1, c_2 \in RC(S), \text{ thus } c_1' \vee c_2' \in RC(S), \text{ since } c_1 \text{ and } c_2 \text{ resolve to } c_1' \vee c_2'.}$

(vii) $l_0 \wedge \cdots \wedge l_{m-1} \wedge \sigma_m \models \neg c_1$

so $l_0 \wedge \cdots \wedge l_{m-1} \wedge \sigma_m \models \neg(c_1' \vee \neg\sigma_m)$

so $l_0 \wedge \cdots \wedge l_{m-1} \wedge \sigma_m \models \neg c_1' \wedge \sigma_m$ [DeMorgan]

so $l_0 \wedge \cdots \wedge l_{m-1} \models \neg c_1'$

Analogously, $l_0 \wedge \cdots \wedge l_{m-1} \models \neg c_2'$

Therefore, $l_0 \wedge \cdots \wedge l_{m-1} \models \neg c_1' \wedge \neg c_2'$
$\equiv \neg(c_1' \vee c_2')$ [DeMorgan]

Since $c_1' \vee c_2' \in RC(S)$, we obtain that $l_0 \wedge \cdots \wedge l_{m-1} \models \neg c$ for some $c \in RC(S)$, which means m was NOT the first iteration at which $l_0 \wedge \cdots \wedge l_m \models \neg c$.

— CONTRADICTION!

(viii) Thus, at <u>every</u> iteration m,

$$l_1 \wedge \cdots \wedge l_m \not\models \neg c \quad \forall c \in RC(S)$$

So $l_1 \wedge \cdots \wedge l_M \not\models \neg c \quad \forall c \in RC(S)$

$\boxed{\therefore\ l_1 \wedge \cdots \wedge l_M \models c \quad \forall c \in RC(S) \quad \text{QED}}$

(vi) $\neg B \vee \neg C$ resolves with $\neg A \vee C$ to obtain $\neg A \vee \neg B$, which is in $RC(S)$.

RESOLUTION

(10) So what have we shown so far?

(i) we want to compute whether

$$\alpha \models \beta$$

for any $\alpha, \beta \in \mathcal{L}(\Sigma)$

this is equivalent to showing whether

$$\alpha \vee \neg \beta$$

is satisfiable

???

(ii) we want to compute whether

γ is satisfiable

for any $\gamma \in \mathcal{L}(\Sigma)$ such that γ is in CNF

this can be done using <u>resolution</u> in a finite number of steps

---

(11) We're missing one key piece to bring this home. Given a non-CNF sentence α, can we convert this into a CNF sentence β such that α is satisfiable iff β is satisfiable?

RESOLUTION

(12) The answer is yes. Let's go through the recipe, using example (non-CNF) sentence $(\text{Bird} \Leftrightarrow (\text{Penguin} \vee \text{Fly}))$

(i) Replace $(\alpha \Leftrightarrow \beta)$ with $((\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha))$:

$$((B \Rightarrow (P \vee F)) \wedge ((P \vee F) \Rightarrow B))$$

(ii) Replace $(\alpha \Rightarrow \beta)$ with $(\neg \alpha \vee \beta)$:

$$((\neg B \vee (P \vee F)) \wedge (\neg (P \vee F) \vee B))$$

(iii) Move "$\neg$" "inwards" with three replacements: $\neg\neg\alpha$ with α, $\neg(\alpha \wedge \beta)$ with $(\neg\alpha \vee \neg\beta)$, and $\neg(\alpha \vee \beta)$ with $(\neg\alpha \wedge \neg\beta)$:

$$((\neg B \vee (P \vee F)) \wedge ((\neg P \wedge F) \vee B))$$

(iv) Distribute ands-over-ors and ors-over-ands with two replacements: $(\alpha \wedge (\beta \vee \gamma))$ with $((\alpha \wedge \beta) \vee (\alpha \wedge \gamma))$ and $(\alpha \vee (\beta \wedge \gamma))$ with $((\alpha \vee \beta) \wedge (\alpha \vee \gamma))$

$$(\underbrace{(\neg B \vee (P \vee F))}_{\text{Clause 1}} \wedge (\underbrace{(\neg P \vee B)}_{\text{clause 2}} \wedge \underbrace{(F \vee B)}_{\text{clause 3}}))$$

$$\neg B \vee P \vee F \quad \wedge \quad \neg P \vee B \quad \wedge \quad F \vee B$$

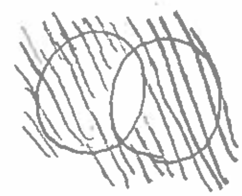
RESOLUTION

(13) It is relatively straightforward to prove the correctness of this conversion by showing the correctness of each step. For instance, we can show that

$$I(\neg(\alpha \wedge \beta)) = I((\neg\alpha \vee \neg\beta))$$

as follows:

$$\begin{aligned}
I(\neg(\alpha \wedge \beta)) &= M(\Sigma) - I((\alpha \wedge \beta)) \\
&= M(\Sigma) - (I(\alpha) \cap I(\beta))
\end{aligned}$$



$$= (M(\Sigma) - I(\alpha)) \cup (M(\Sigma) - I(\beta))$$

$$\begin{aligned}
&= I(\neg\alpha) \cup I(\neg\beta) \\
&= I((\neg\alpha \vee \neg\beta))
\end{aligned}$$