Axiom Schema Instances

$$\varphi \Rightarrow (\psi \Rightarrow \varphi)$$

$p \Rightarrow (p \Rightarrow p)$	$p \Rightarrow (p \wedge p \Rightarrow p)$	$p \Rightarrow (p \vee p \Rightarrow p)$
$p \Rightarrow (q \Rightarrow p)$	$p \Rightarrow (p \wedge q \Rightarrow p)$	$p \Rightarrow (p \vee q \Rightarrow p)$
$p \Rightarrow (r \Rightarrow p)$	$p \Rightarrow (p \wedge r \Rightarrow p)$	$p \Rightarrow (p \vee r \Rightarrow p)$
$q \Rightarrow (p \Rightarrow q)$	$\ldots$	$\ldots$
$q \Rightarrow (q \Rightarrow q)$		
$q \Rightarrow (r \Rightarrow q)$		
$r \Rightarrow (p \Rightarrow r)$		
$r \Rightarrow (q \Rightarrow r)$		
$r \Rightarrow (r \Rightarrow r)$		

Proofs may be short, but many alternatives to consider.

Propositional Resolution

Propositional resolution is a rule of inference.

Using propositional resolution alone (without axiom schemata or other rules of inference), it is possible to build a theorem prover that is sound and complete for all of Propositional Logic.

The search space using propositional resolution is much smaller than for Modus Ponens and the Standard Axiom Schemata.

Clausal Form

Propositional resolution works only on expressions in *clausal form*.

Fortunately, it is possible to convert any set of propositional calculus sentences into an equivalent set of sentences in clausal form.

Clausal Form

A *literal* is either an atomic sentence or a negation of an atomic sentence.

$$p$$
$$\neg p$$

A *clausal sentence* is either a literal or a disjunction of literals.

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

A *clause* is a set of literals.

$$\{p\}$$
$$\{\neg p\}$$
$$\{p,q\}$$

Empty Sets

The empty clause {} is unsatisfiable.

Why? It is equivalent to an empty disjunction.

Conversion to Clausal Form

Implications Out:

$$\varphi_1 \Rightarrow \varphi_2 \quad \rightarrow \quad \neg\varphi_1 \vee \varphi_2$$

$$\varphi_1 \Leftarrow \varphi_2 \quad \rightarrow \quad \varphi_1 \vee \neg\varphi_2$$

$$\varphi_1 \Leftrightarrow \varphi_2 \quad \rightarrow \quad (\neg\varphi_1 \vee \varphi_2) \wedge (\varphi_1 \vee \neg\varphi_2)$$

Negations In:

$$\neg\neg\varphi \quad \rightarrow \quad \varphi$$

$$\neg(\varphi_1 \wedge \varphi_2) \quad \rightarrow \quad \neg\varphi_1 \vee \neg\varphi_2$$

$$\neg(\varphi_1 \vee \varphi_2) \quad \rightarrow \quad \neg\varphi_1 \wedge \neg\varphi_2$$

Conversion to Clausal Form

Distribution

$$\varphi_1 \vee (\varphi_2 \wedge \varphi_3) \quad \rightarrow \quad (\varphi_1 \vee \varphi_2) \wedge (\varphi_1 \vee \varphi_n)$$

$$(\varphi_1 \wedge \varphi_2) \vee \varphi_3 \quad \rightarrow \quad (\varphi_1 \vee \varphi_3) \wedge (\varphi_2 \vee \varphi_3)$$

$$\varphi_1 \vee (\varphi_2 \vee \varphi_3) \quad \rightarrow \quad (\varphi_1 \vee \varphi_2 \vee \varphi_3)$$

$$(\varphi_1 \vee \varphi_2) \vee \varphi_3 \quad \rightarrow \quad (\varphi_1 \vee \varphi_2 \vee \varphi_3)$$

$$\varphi_1 \wedge (\varphi_2 \wedge \varphi_3) \quad \rightarrow \quad (\varphi_1 \wedge \varphi_2 \wedge \varphi_3)$$

$$(\varphi_1 \wedge \varphi_2) \wedge \varphi_3 \quad \rightarrow \quad (\varphi_1 \wedge \varphi_2 \wedge \varphi_3)$$

Operators Out

$$\varphi_1 \vee \ldots \vee \varphi_n \quad \rightarrow \quad \{\varphi_1, \ldots, \varphi_n\}$$

$$\varphi_1 \wedge \ldots \wedge \varphi_n \quad \rightarrow \quad \varphi_1, \ldots, \varphi_n$$

Example

	$g \wedge (r \Rightarrow f)$
I	$g \wedge (\neg r \vee f)$
N	$g \wedge (\neg r \vee f)$
D	$g \wedge (\neg r \vee f)$
O	$\{g\}$
	$\{\neg r, f\}$

Example

$$\neg(g \wedge (r \Rightarrow f))$$

I $\quad \neg(g \wedge (\neg r \vee f))$

N $\quad \neg g \vee \neg(\neg r \vee f))$

$$\neg g \vee (\neg\neg r \wedge \neg f)$$

$$\neg g \vee (r \wedge \neg f)$$

D $\quad (\neg g \vee r) \wedge (\neg g \vee \neg f)$

O $\quad \{\neg g, r\}$

$$\{\neg g, \neg f\}$$

Resolution Principle

General:

$$\frac{\begin{array}{l}\{\varphi_1,\ldots,\chi,\ldots,\varphi_m\}\\ \{\psi_1,\ldots,\neg\chi,\ldots,\psi_n\}\end{array}}{\{\varphi_1,\ldots,\varphi_m,\psi_1,\ldots,\psi_n\}}$$

Example:

$$\frac{\begin{array}{l}\{p,q\}\\ \{\neg p,r\}\end{array}}{\{q,r\}}$$

Issues

Collapse

$$\frac{\begin{array}{l}\{\neg p, q\}\\ \{p, q\}\end{array}}{\{q\}}$$

Singletons

$$\frac{\begin{array}{l}\{\neg p, q\}\\ \{p\}\end{array}}{\{q\}} \qquad \frac{\begin{array}{l}\{p\}\\ \{\neg p\}\end{array}}{\{\,\}}$$

Issues

Multiple Conclusions

$$\frac{\begin{array}{l}\{p, q\}\\ \{\neg p, \neg q\}\end{array}}{\begin{array}{l}\{p, \neg p\}\\ \{q, \neg q\}\end{array}}$$

Single Application Only

$$\frac{\begin{array}{l}\{p, q\}\\ \{\neg p, \neg q\}\end{array}}{\{\}}$$

Wrong!!

Special Cases

Modus Ponens	Modus Tolens	Chaining
$$\frac{\begin{array}{l} p \Rightarrow q \\ p \end{array}}{q}$$	$$\frac{\begin{array}{l} p \Rightarrow q \\ \neg q \end{array}}{\neg p}$$	$$\frac{\begin{array}{l} p \Rightarrow q \\ q \Rightarrow r \end{array}}{p \Rightarrow r}$$
$$\frac{\begin{array}{l} \{\neg p, q\} \\ \{p\} \end{array}}{\{q\}}$$	$$\frac{\begin{array}{l} \{\neg p, q\} \\ \{\neg q\} \end{array}}{\{\neg p\}}$$	$$\frac{\begin{array}{l} \{\neg p, q\} \\ \{\neg q, r\} \end{array}}{\{\neg p, r\}}$$

Incompleteness?

Propositional Resolution is not *generatively* complete.

We cannot generate $p \Rightarrow (q \Rightarrow p)$ using propositional resolution. There are no premises. Consequently, there are no conclusions.

Answer

This apparent problem disappears if we take the clausal form of the premises (if any) together with the negated goal and try to derive the empty clause.

General Method: To determine whether a set Δ of sentences logically entails a sentence φ, rewrite $\Delta \cup \{\neg\varphi\}$ in clausal form and try to derive the empty clause using the resolution rule of inference.

Example

	$\neg(p \Rightarrow (q \Rightarrow p))$
I	$\neg(\neg p \vee \neg q \vee p)$
N	$\neg\neg p \wedge \neg\neg q \wedge \neg p$
	$p \wedge q \wedge \neg p$
D	$p \wedge q \wedge \neg p$
O	$\{p\}$
	$\{q\}$
	$\{\neg p\}$

Example

If Mary loves Pat, then Mary loves Quincy. If it is Monday, Mary loves Pat or Quincy. Prove that, if it is Monday, then Mary loves Quincy.

1.	$\{\neg p, q\}$	Premise
2.	$\{\neg m, p, q\}$	Premise
3.	$\{m\}$	Negated Goal
4.	$\{\neg q\}$	Negated Goal
5.	$\{p, q\}$	3,2
6.	$\{q\}$	5,1
7.	$\{\}$	6,4

Example

Heads you win. Tails I lose. Show that you always win.

1.	$\{\neg h, y\}$	Premise
2.	$\{\neg t, \neg m\}$	Premise
3.	$\{h, t\}$	Premise
4.	$\{\neg h, \neg t\}$	Premise
5.	$\{m, y\}$	Premise
6.	$\{\neg m, \neg y\}$	Premise
7.	$\{\neg y\}$	Negated Goal
8.	$\{t, y\}$	3,1
9.	$\{\neg m, y\}$	8,2
10.	$\{y\}$	9,5
11.	$\{\}$	10,7

Soundness and Completeness

A sentence is *provable* from a set of sentences by propositional resolution if and only if there is a derivation of the empty clause from the clausal form of $\Delta \cup \{\neg\varphi\}$.

Theorem: Propositional Resolution is sound and complete, i.e. $\Delta \models \varphi$ if and only if $\Delta \vdash \varphi$.

Two Finger Method

1.	$\{p, q\}$	Premise
2.	$\{\neg p, r\}$	Premise
3.	$\{\neg q, r\}$	Premise
4.	$\{\neg r\}$	Premise
5.	$\{q, r\}$	1, 2
6.	$\{p, r\}$	1, 3
7.	$\{\neg p\}$	2, 4
8.	$\{\neg q\}$	3, 4
9.	$\{r\}$	3, 5
10.	$\{q\}$	4, 5
11.	$\{r\}$	2, 6
12.	$\{p\}$	4, 6
13.	$\{q\}$	1, 7
14.	$\{r\}$	6, 7
15.	$\{p\}$	1, 8
16.	$\{r\}$	5, 8
17.	$\{\}$	4, 9

TFM With Identical Clause Elimination

1.	$\{p, q\}$	Premise
2.	$\{\neg p, r\}$	Premise
3.	$\{\neg q, r\}$	Premise
4.	$\{\neg r\}$	Premise
5.	$\{q, r\}$	$1, 2$
6.	$\{p, r\}$	$1, 3$
7.	$\{\neg p\}$	$2, 4$
8.	$\{\neg q\}$	$3, 4$
9.	$\{r\}$	$3, 5$
10.	$\{q\}$	$4, 5$
11.	$\{p\}$	$4, 6$
12.	$\{\}$	$4, 9$

TFM With ICE, Complement Detection

1.	$\{p, q\}$	Premise
2.	$\{\neg p, r\}$	Premise
3.	$\{\neg q, r\}$	Premise
4.	$\{\neg r\}$	Premise
5.	$\{q, r\}$	$1, 2$
6.	$\{p, r\}$	$1, 3$
7.	$\{\neg p\}$	$2, 4$
8.	$\{\neg q\}$	$3, 4$
9.	$\{r\}$	$3, 5$
10.	$\{\}$	$4, 9$

Termination

Theorem: There is a resolution derivation of a conclusion from a set of premises if and only if there is a derivation using the two finger method.

Theorem: Propositional resolution using the two-finger method always terminates.

Proof: There are only finitely many clauses that can be constructed from a finite set of logical constants.

Decidability of Propositional Entailment

Propositional resolution is a decision procedure for Propositional Logic.

Logical entailment for Propositional Logic is decidable.

Coin Logic

Syntax:
The logical constants: quarters, nickels, dimes
Negation: coin upside down
Disjunction: stack of coins
Conjunction: set of stacks
Almost exactly clausal form.

Propositional Resolution:
Add two stacks,
deleting at most one pair of complementary coins.

Example

Quarter means it is Monday.
Nickel means Mary loves Pat.
Dime means Mary loves Quincy.

[¬Nickel,Dime]: If Mary loves Pat, Mary loves Quincy.
[¬Quarter,Nickel,Dime]: Monday Mary loves Pat or Quincy.
[¬Nickel, ¬Dime]:Mary loves only one of the two.

[¬Quarter,Dime]: If Monday, Mary loves Quincy.
[¬Quarter,-Nickel]: If Monday, Mary does not love Pat.