Theorem Proving (Resolution) "Resolution is proofs by **refutation**, in another word, to prove a goal statement must make it with negation. Resolution attempts to show that the negation of statement produce contradiction with known statements".

John Allan Robinson, born 1928, discovered the technique called resolution.

Outline of this lecture

- Definitions of Resolution
- The Basic of Resolution
- Clouse Form steps and Example
- Resolution in Propositional Logic
- Example.

Definitions of Resolution

Resolution is a technique for proving theorem in the *propositional logic* or *predicate calculus* requires statement placed in *Clouse Form*.

The Basic of Resolution

Is a simple iterative process, at each step of which two clauses, are compared (resolved), yielding a new clause that has been inferred from them. Suppose that there are two clauses in system:

$$P \lor Q$$

$$\sim P \lor Q$$

That both clauses must be true, if $\sim P$ then :

$$P \qquad \sim P$$

Will introduce empty clause, and contradiction exists.

Clouse Form

Now that you have the general idea of how proof by resolution works, it is time to understand various manipulations that make harder proofs possible. Basically, the point of these manipulations is to transform arbitrary logic expressions into a form that enables resolution. Specifically, you need a way to transform the given axioms into equivalent, new axioms that are all disjunctions of literals. Said another way, you want the new axioms to be in **clause form**.

Example: convert the following WFF to Clause Form.

$$\forall_X[\mathbf{B}(\mathbf{x}) \rightarrow (\exists y\ [\mathbf{O}(\mathbf{x},\mathbf{y}) \wedge \sim \mathbf{P}(\mathbf{y})]\ \wedge\ \sim\exists Y\ [\mathbf{O}(\mathbf{x},\mathbf{y}) \wedge \mathbf{O}(\mathbf{y},\mathbf{x})]\ \wedge\ \forall Y[\sim\mathbf{B}(\mathbf{y}) \rightarrow \sim\mathbf{E}(\mathbf{x},\mathbf{y})])]$$

Next, let us consider the steps needed to transform arbitrary logical expressions into clause form. Once explained, the steps will be summarized in a procedure.

1. Eliminate implications

All you need to do is to substitute $\neg E_1 \vee E_2$ for $E_1 \Rightarrow E_2$

The Result: Step1

$$\forall_X[\sim \mathbf{B}(\mathbf{x}) \vee (\exists y\,[\mathbf{O}(\mathbf{x},\mathbf{y}) \wedge \sim \mathbf{P}(\mathbf{y})] \wedge \sim\exists Y\,[\mathbf{O}(\mathbf{x},\mathbf{y}) \wedge \mathbf{O}(\mathbf{y},\mathbf{x})] \wedge \forall Y[\sim \mathbf{B}(\mathbf{y}) \rightarrow \sim \mathbf{E}(\mathbf{x},\mathbf{y})])]$$

Step2

$$\forall_X[\sim \mathbf{B}(\mathbf{x}) \vee (\exists y\,[\mathbf{O}(\mathbf{x},\mathbf{y}) \wedge \sim \mathbf{P}(\mathbf{y})] \wedge \sim\exists Y\,[\mathbf{O}(\mathbf{x},\mathbf{y}) \wedge \mathbf{O}(\mathbf{y},\mathbf{x})] \wedge \forall Y[\sim(\sim \mathbf{B}(\mathbf{y})) \vee \sim \mathbf{E}(\mathbf{x},\mathbf{y})])]$$

2. Move negations down to the atomic formulas

Doing this step requires a number of identities, one for dealing with the negation of & expressions, one for $\vee$ expressions, one for $\neg$ expressions, and one each for $\forall$ and $\exists$:

$$\neg(E_1 \& E_2) \rightarrow (\neg E_1) \vee (\neg E_2)$$
$$\neg(E_1 \vee E_2) \rightarrow (\neg E_1) \& (\neg E_2)$$
$$\neg(\neg E_1) \rightarrow E_1$$
$$\neg\forall x[E_1(x)] \rightarrow \exists x[\neg E_1(x)]$$
$$\neg\exists x[E_1(x)] \rightarrow \forall x[\neg E_1(x)]$$

For the example, you need the third identity, which eliminates the double negations, and you need the final identity, which eliminates an $\exists$ and introduces another $\forall$, leaving this:

The Result:

$$\forall_X[\sim \mathbf{B}(\mathbf{x}) \vee (\exists y\,[\mathbf{O}(\mathbf{x},\mathbf{y}) \wedge \sim \mathbf{P}(\mathbf{y})] \wedge \forall Y\,[\sim \mathbf{O}(\mathbf{x},\mathbf{y}) \vee \sim \mathbf{O}(\mathbf{y},\mathbf{x})] \wedge \forall Y[\mathbf{B}(\mathbf{y}) \vee \sim \mathbf{E}(\mathbf{x},\mathbf{y})])]$$

3. Eliminate existential quantifiers

There is a function that take argument X and return proper Y. You do not necessarily to know how the function works, but this function must exist. Let us write it Support function.

The function that eliminate existential quantifiers called **Skolem Function**.

The Result:

$$\forall_X[\sim \mathbf{B}(\mathbf{x}) \vee (\,[\mathbf{O}(\mathbf{x},\mathbf{f}(\mathbf{x})) \wedge \sim \mathbf{P}(\mathbf{f}(\mathbf{x}))] \wedge \forall Y\,[\sim \mathbf{O}(\mathbf{x},\mathbf{y}) \vee \sim \mathbf{O}(\mathbf{y},\mathbf{x})] \wedge \forall Y[\mathbf{B}(\mathbf{y}) \vee \sim \mathbf{E}(\mathbf{x},\mathbf{y})])]$$

4. Rename variables, as necessary, so that no two variables are the same

you can rename any duplicates so that each quantifier has a unique name. You do this renaming because you want to move all the universal quantifiers together at the left of each expression in the next step, without confounding

The Result:

$$\forall_X[\sim B(x) \vee ([O(x, f(x)) \wedge \sim P(f(x))] \wedge \forall Y [\sim O(x,y) \vee \sim O(y,x)] \wedge \forall z[B(z) \vee \sim E(x,z)])]$$

5. Move the universal quantifiers to the left

This step works because, by now, each quantifier uses a unique variable name—no confusion results from leftward movement. In the example, the result is as follows:

The Result:

$$\forall_X \forall Y \forall z[\sim B(x) \vee ([O(x, f(x)) \wedge \sim P(f(x))] \wedge [\sim O(x,y) \vee \sim O(y,x)] \wedge [B(z) \vee \sim E(x,z)])]$$

6. Move the disjunction down to the literals

This step requires you to move the $\vee$s inside the &s; to do this movement, you need to use one of the distributive laws:

$$E_1 \vee (E_2 \& E_3) \Leftrightarrow (E_1 \vee E_2) \& (E_1 \vee E_3)$$

For the example, let us do the work in two steps:

First step:

$\forall_X \forall Y \forall z[\ (\sim B(x) \vee (O(x, f(x)) \wedge \sim P(f(x)))) \wedge (\sim B(x) \vee \sim O(x, y) \vee \sim O(y, x)) \wedge$

$(\sim Bx \vee Bz \vee \sim E(x,z))$

Second step:

$\forall_X \forall Y \forall z[\ (\sim B(x) \vee O(x, f(x))) \wedge (\sim B(x) \vee \sim P(f(x))) \wedge (\sim B(x) \vee \sim O(x, y) \vee$

$\sim O_{y,x}) \wedge (\sim Bx \vee Bz \vee \sim E(x,z))$

7. Eliminate the conjunctions

Actually, you do not really eliminate them. Instead, you simply write each part of a conjunction as though it were a separate axiom. This way of writing a conjunction makes sense, because each part of a conjunction must be true if the whole conjunction is true. Here is the result:

The Result:

$\forall_X[\ \sim\mathbf{B}(\mathbf{x}) \vee \mathbf{O}(\mathbf{x}, \mathbf{f}(\mathbf{x}))\ ]$

$\forall_X[\ \sim\mathbf{B}(\mathbf{x}) \vee \sim \mathbf{P}\ (\mathbf{f}(\mathbf{x}))\ ]$

$\forall_X \forall Y[\ \sim\mathbf{B}(\mathbf{x}) \vee \sim\mathbf{O}(\mathbf{x}, \mathbf{y}) \vee \sim\mathbf{O}(\mathbf{y}, \mathbf{x})\ ]$

$\forall_X \forall z[\ \sim\mathbf{B}(\mathbf{x}) \vee \mathbf{B}(\mathbf{z}) \vee \sim\mathbf{E}(\mathbf{x}, \mathbf{z})\ ]$

8. Rename variables, as necessary, so that no two variables are the same

There is no problem with renaming variables at this step, for you are merely renaming the universally quantified variables in each part of a conjunction. Because each of the conjoined parts must be true for any variable values, it does not matter whether the variables have different names for each part.

The Result:

$$\forall_X[\ \sim\mathbf{B}(\mathbf{x}) \vee \mathbf{O}(\mathbf{x}, \mathbf{f}(\mathbf{x}))\ ]$$

$$\forall_w[\ \sim\mathbf{B}(\mathbf{w}) \vee \sim \mathbf{P}\ (\mathbf{f}(\mathbf{w}))\ ]$$

$$\forall_u \forall y[\ \sim\mathbf{B}(\mathbf{u}) \vee \sim\mathbf{O}(\mathbf{u}, \mathbf{y}) \vee \sim\mathbf{O}(\mathbf{y}, \mathbf{u})\ ]$$

$$\forall_v \forall z[\ \sim\mathbf{B}(\mathbf{v}) \vee \mathbf{B}(\mathbf{z}) \vee \sim\mathbf{E}(\mathbf{v}, \mathbf{z})\ ]$$

9. Eliminate the universal quantifiers

Actually, you do not really eliminate them. You just adopt a convention whereby all variables at this point are presumed to be universally quantified. Now, the example looks like this:

The Result:

$$\sim B(x) \vee O(x, f(x))$$

$$\sim B(w) \vee \sim P(f(w))$$

$$\sim B(u) \vee \sim O(u, y) \vee \sim O(y, u)$$

$$\sim B(v) \vee B(z) \vee \sim E(v, z)$$

Now, the result is in Clause Form(each clause consists of disjunction of literals.

Resolution in Propositional Logic

The procedure for producing a proof by resolution of proposition S with respect to a set of atomics F Is the following:

1. Convert all the propositions of F to Clause Form.
2. Negate S and convert the result to Clause Form. Add it to the set of clauses obtained in step1.
3. Repeat until either a contradiction is found or no progress can be made:
   3.1. Select two clauses, call them the Parent Clauses.
   3.2. Resolve them together, the resulting clause called "Resolvent", will be the disjunction of all literals of both Parent Clauses with the following exception:

   IF there are any pairs of literals L and ~ L such that one of the parent clauses cantinas L and the other contains ~ L
   THEN eliminate both L and ~ L from Resolvent.

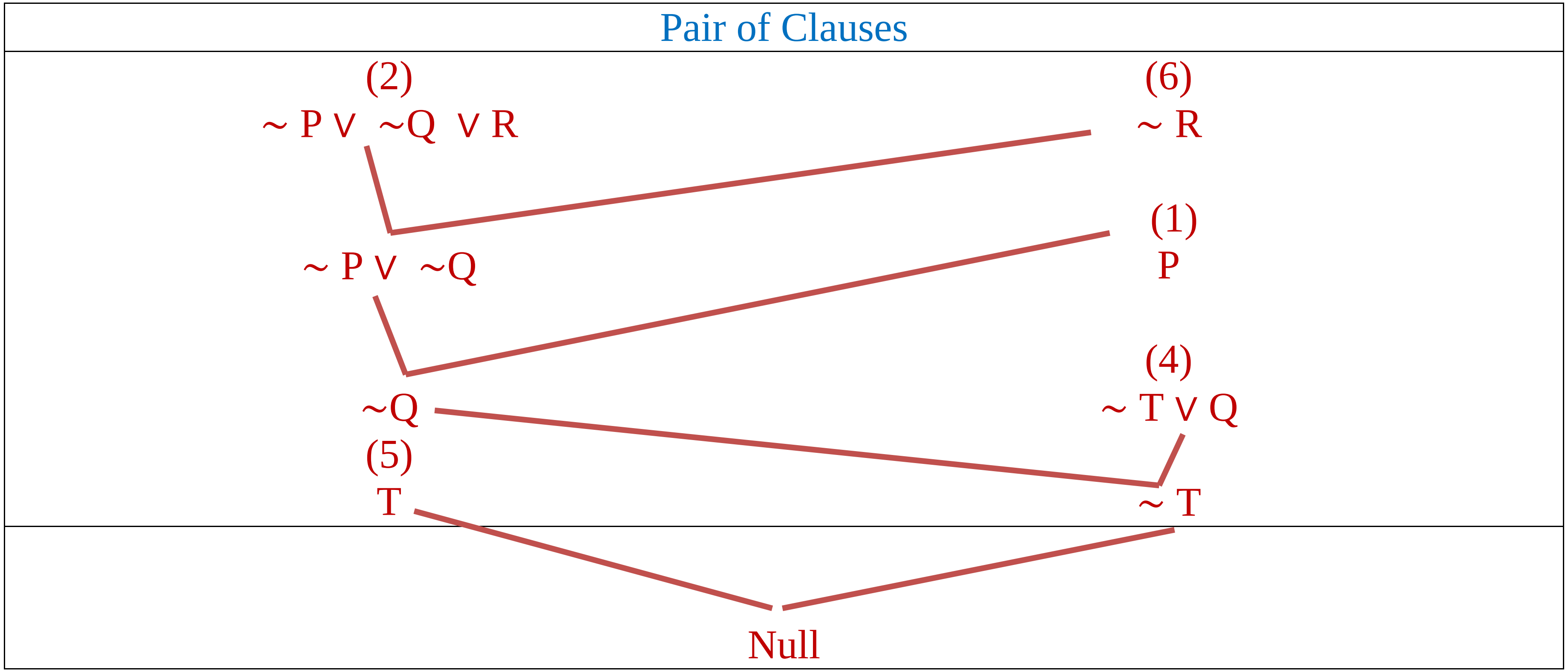
   3.3. IF the Resolvent is the Empty
   THEN Contradiction has been found.
   Else add it to the set of clauses an available to the procedure.

Example:

Suppose we are given the axioms shown in the following:
We want to prove R. it must negate R

Given Axioms	Clause Form
P	P …(1)
$(P \land Q) \rightarrow R$	$\sim (P \land Q) \lor R$ $\sim P \lor \sim Q \lor R$ …(2)
$(S \lor T) \rightarrow Q$	$\sim (S \lor T) \lor Q$ $(\sim S \land \sim T) \lor Q$ $(\sim S \lor Q) \land (\sim T \lor Q)$ $\sim S \lor Q$ …(3) $\sim T \lor Q$ …(4)
T	T …(5)
R Goal	$\sim R$ …(6)

We begin select paring of clauses and resolve them:



Next Lecture

Resolution in Predicate Calculus