Outline

A Puzzle

Isaac and Albert were excitedly describing the result of the Third Annual International Science Fair Extravaganza in Sweden. There were three contestants, Louis, Rene, and Johannes.

Isaac reported that Louis won the fair, while Rene came in second. Albert, on the other hand, reported that Johannes won the fair, while Louis came in second.

A Puzzle

Isaac and Albert were excitedly describing the result of the Third Annual International Science Fair Extravaganza in Sweden. There were three contestants, Louis, Rene, and Johannes.

Isaac reported that Louis won the fair, while Rene came in second. Albert, on the other hand, reported that Johannes won the fair, while Louis came in second.

In fact, neither Isaac nor Albert had given a correct report of the results of the science fair. Each of them had given one true statement and one false statement. What was the actual placing of the three contestants?

A Puzzle

Isaac and Albert were excitedly describing the result of the Third Annual International Science Fair Extravaganza in Sweden. There were three contestants, Louis, Rene, and Johannes.

Isaac reported that Louis won the fair, while Rene came in second. Albert, on the other hand, reported that Johannes won the fair, while Louis came in second.

In fact, neither Isaac nor Albert had given a correct report of the results of the science fair. Each of them had given one true statement and one false statement. What was the actual placing of the three contestants?

How can we solve this kind of puzzle?

Propositional Satisfiability Problem

Given a propositional formula A, check whether it is satisfiable or unsatisfiable.

Propositional Satisfiability Problem

Given a propositional formula A, check whether it is satisfiable or unsatisfiable.

If A is satisfiable, we also want to find a satisfying assignment for A, that is, a model of A.

Propositional Satisfiability Problem

Given a propositional formula A, check whether it is satisfiable or unsatisfiable.

If A is satisfiable, we also want to find a satisfying assignment for A, that is, a model of A.

It is one of the most famous combinatorial problems in computer science.

Propositional Satisfiability Problem

Given a propositional formula A, check whether it is satisfiable or unsatisfiable.

If A is satisfiable, we also want to find a satisfying assignment for A, that is, a model of A.

It is one of the most famous combinatorial problems in computer science.

It is a very hard problem with a surprisingly large number of practical applications.

Propositional Satisfiability Problem

Given a propositional formula A, check whether it is satisfiable or unsatisfiable.

If A is satisfiable, we also want to find a satisfying assignment for A, that is, a model of A.

It is one of the most famous combinatorial problems in computer science.

It is a very hard problem with a surprisingly large number of practical applications.

It is also the first ever problem to be proved NP-complete.

Russian Spy Puzzle

There are three persons: Stirlitz, Müller, and Eismann. It is known that exactly one of them is Russian, while the other two are Germans. Moreover, every Russian must be a spy.

(Images from http://hu.wikipedia.org/ and http://www.elomagazin.com)

Russian Spy Puzzle

There are three persons: Stirlitz, Müller, and Eismann. It is known that exactly one of them is Russian, while the other two are Germans. Moreover, every Russian must be a spy.

When Stirlitz meets Müller in a corridor, he makes the following joke: “you know, Müller, you are as German as I am Russian”. It is known that Stirlitz always tells the truth when he is joking.

(Images from http://hu.wikipedia.org/ and http://www.elomagazin.com)

Russian Spy Puzzle

There are three persons: Stirlitz, Müller, and Eismann. It is known that exactly one of them is Russian, while the other two are Germans. Moreover, every Russian must be a spy.

When Stirlitz meets Müller in a corridor, he makes the following joke: "you know, Müller, you are as German as I am Russian". It is known that Stirlitz always tells the truth when he is joking.

We have to show that Eismann is not a Russian spy.

(Images from `http://hu.wikipedia.org/` and `http://www.elomagazin.com`)

Russian Spy Puzzle

There are three persons: Stirlitz, Müller, and Eismann. It is known that exactly one of them is Russian, while the other two are Germans. Moreover, every Russian must be a spy.

When Stirlitz meets Müller in a corridor, he makes the following joke: “you know, Müller, you are as German as I am Russian”. It is known that Stirlitz always tells the truth when he is joking.

We have to show that Eismann is not a Russian spy. How can we solve problems of this kind?

(Images from http://hu.wikipedia.org/ and http://www.elomagazin.com)

Formalisation in Propositional Logic

Introduce nine propositional variables as in the following table:

	Stirlitz	Müller	Eismann
Russian	RS	RM	RE
German	GS	GM	GE
Spy	SS	SM	SE

Formalisation in Propositional Logic

Introduce nine propositional variables as in the following table:

	Stirlitz	Müller	Eismann
Russian	RS	RM	RE
German	GS	GM	GE
Spy	SS	SM	SE

For example,

SE : Eismann is a Spy
RS : Stirlitz is Russian

Formalisation in Propositional Logic

There are three persons: Stirlitz, Müller, and Eismann. It is known that exactly one of them is Russian, while the other two are Germans.

Moreover, every Russian must be a spy.

When Stirlitz meets Müller in a corridor, he makes the following joke: “you know, Müller, you are as German as I am Russian”.

We have to show that Eismann is not a Russian spy.

Formalisation in Propositional Logic

There are three persons: Stirlitz, Müller, and Eismann. It is known that exactly one of them is Russian, while the other two are Germans.

$$(RS \land GM \land GE) \lor (GS \land RM \land GE) \lor (GS \land GM \land RE).$$

Moreover, every Russian must be a spy.

When Stirlitz meets Müller in a corridor, he makes the following joke: “you know, Müller, you are as German as I am Russian”.

We have to show that Eismann is not a Russian spy.

Formalisation in Propositional Logic

There are three persons: Stirlitz, Müller, and Eismann. It is known that exactly one of them is Russian, while the other two are Germans.

$$(RS \land GM \land GE) \lor (GS \land RM \land GE) \lor (GS \land GM \land RE).$$

Moreover, every Russian must be a spy.

$$(RS \to SS) \land (RM \to SM) \land (RE \to SE).$$

When Stirlitz meets Müller in a corridor, he makes the following joke: “you know, Müller, you are as German as I am Russian”.

We have to show that Eismann is not a Russian spy.

Formalisation in Propositional Logic

There are three persons: Stirlitz, Müller, and Eismann. It is known that exactly one of them is Russian, while the other two are Germans.

$$(RS \land GM \land GE) \lor (GS \land RM \land GE) \lor (GS \land GM \land RE).$$

Moreover, every Russian must be a spy.

$$(RS \to SS) \land (RM \to SM) \land (RE \to SE).$$

When Stirlitz meets Müller in a corridor, he makes the following joke: "you know, Müller, you are as German as I am Russian".

$$RS \leftrightarrow GM.$$

We have to show that Eismann is not a Russian spy.

Formalisation in Propositional Logic

There are three persons: Stirlitz, Müller, and Eismann. It is known that exactly one of them is Russian, while the other two are Germans.

$$(RS \wedge GM \wedge GE) \vee (GS \wedge RM \wedge GE) \vee (GS \wedge GM \wedge RE).$$

Moreover, every Russian must be a spy.

$$(RS \rightarrow SS) \wedge (RM \rightarrow SM) \wedge (RE \rightarrow SE).$$

When Stirlitz meets Müller in a corridor, he makes the following joke: “you know, Müller, you are as German as I am Russian”.

$$RS \leftrightarrow GM.$$

Hidden: Russians are not Germans.

$$(RS \leftrightarrow \neg GS) \wedge (RM \leftrightarrow \neg GM) \wedge (RE \leftrightarrow \neg GE).$$

We have to show that Eismann is not a Russian spy.

Formalisation in Propositional Logic

There are three persons: Stirlitz, Müller, and Eismann. It is known that exactly one of them is Russian, while the other two are Germans.

$$(RS \wedge GM \wedge GE) \vee (GS \wedge RM \wedge GE) \vee (GS \wedge GM \wedge RE).$$

Moreover, every Russian must be a spy.

$$(RS \to SS) \wedge (RM \to SM) \wedge (RE \to SE).$$

When Stirlitz meets Müller in a corridor, he makes the following joke: “you know, Müller, you are as German as I am Russian”.

$$RS \leftrightarrow GM.$$

Hidden: Russians are not Germans.

$$(RS \leftrightarrow \neg GS) \wedge (RM \leftrightarrow \neg GM) \wedge (RE \leftrightarrow \neg GE).$$

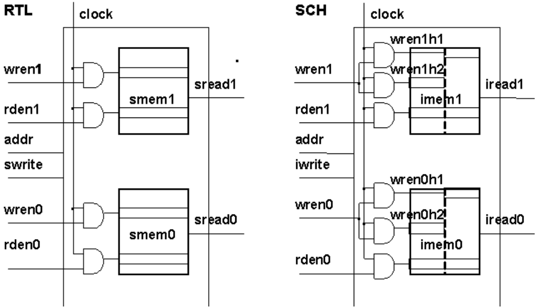
We have to show that Eismann is not a Russian spy.
To this end, we add the following formula

$$RE \wedge SE.$$

and check whether the resulting set of formulas is satisfiable. If it is unsatisfiable, then Eismann cannot be a Russian spy.

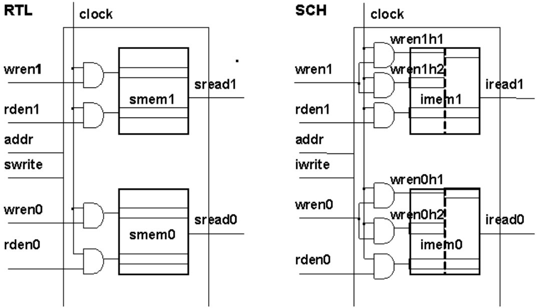
Circuit Equivalence

Given two circuits, check if they are equivalent. For example:



Circuit Equivalence

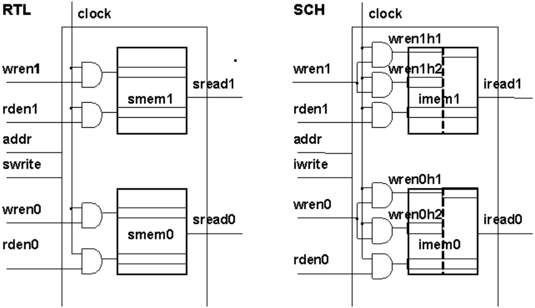
Given two circuits, check if they are equivalent. For example:



Every circuit is, in fact, a propositional formula, specifying the relation between the inputs and the outputs of the circuit.

Circuit Equivalence

Given two circuits, check if they are equivalent. For example:



Every circuit is, in fact, a propositional formula, specifying the relation between the inputs and the outputs of the circuit.

We know that equivalence-checking for propositional formulas can be reduced to unsatisfiability-checking.

Idea: Use Formula Evaluation Methods

Consider $\neg((p \to q) \wedge (p \wedge q \to r) \to (p \to r)$.
We can evaluate it in any interpretation, for example, $\{p \mapsto 0, q \mapsto 0, r \mapsto 0\}$:

	subformula	I_0
1	$\neg((p \to q) \wedge (p \wedge q \to r) \to (p \to r))$	0
2	$(p \to q) \wedge (p \wedge q \to r) \to (p \to r)$	1
3	$p \to r$	1
4	$(p \to q) \wedge (p \wedge q \to r)$	1
5	$p \wedge q \to r$	1
6	$p \to q$	1
7	$p \wedge q$	0
8	p p p	0
9	q q	0
10	r r	0

Truth Tables

$\neg((p \to q) \wedge (p \wedge q \to r) \to (p \to r)$.
Likewise, we can evaluate it in all interpretations:

	subformula	I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8
1	$\neg((p \to q) \wedge (p \wedge q \to r) \to (p \to r))$	0	0	0	0	0	0	0	0
2	$(p \to q) \wedge (p \wedge q \to r) \to (p \to r)$	1	1	1	1	1	1	1	1
3	$p \to r$	1	1	1	1	0	1	0	1
4	$(p \to q) \wedge (p \wedge q \to r)$	1	1	1	1	0	0	0	1
5	$p \wedge q \to r$	1	1	1	1	1	1	0	1
6	$p \to q$	1	1	1	1	0	0	1	1
7	$p \wedge q$	0	0	0	0	0	0	1	1
8	p p p	0	0	0	0	1	1	1	1
9	q q	0	0	1	1	0	0	1	1
10	r r	0	1	0	1	0	1	0	1

Truth Tables

$\neg((p \to q) \land (p \land q \to r) \to (p \to r)$.

Likewise, we can evaluate it in all interpretations:

	subformula	I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8
1	$\neg((p \to q) \land (p \land q \to r) \to (p \to r))$	0	0	0	0	0	0	0	0
2	$(p \to q) \land (p \land q \to r) \to (p \to r)$	1	1	1	1	1	1	1	1
3	$p \to r$	1	1	1	1	0	1	0	1
4	$(p \to q) \land (p \land q \to r)$	1	1	1	1	0	0	0	1
5	$p \land q \to r$	1	1	1	1	1	1	0	1
6	$p \to q$	1	1	1	1	0	0	1	1
7	$p \land q$	0	0	0	0	0	0	1	1
8	p p p	0	0	0	0	1	1	1	1
9	q q	0	0	1	1	0	0	1	1
10	r r	0	1	0	1	0	1	0	1

The formula is unsatisfiable since it is false in every interpretation.

Truth Tables

$\neg((p \to q) \wedge (p \wedge q \to r) \to (p \to r)$.

Likewise, we can evaluate it in all interpretations:

	subformula	I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8
1	$\neg((p \to q) \wedge (p \wedge q \to r) \to (p \to r))$	0	0	0	0	0	0	0	0
2	$(p \to q) \wedge (p \wedge q \to r) \to (p \to r)$	1	1	1	1	1	1	1	1
3	$p \to r$	1	1	1	1	0	1	0	1
4	$(p \to q) \wedge (p \wedge q \to r)$	1	1	1	1	0	0	0	1
5	$p \wedge q \to r$	1	1	1	1	1	1	0	1
6	$p \to q$	1	1	1	1	0	0	1	1
7	$p \wedge q$	0	0	0	0	0	0	1	1
8	p p p	0	0	0	0	1	1	1	1
9	q q	0	0	1	1	0	0	1	1
10	r r	0	1	0	1	0	1	0	1

The formula is unsatisfiable since it is false in every interpretation.

Problem: a formula with n propositional variables has 2^n different interpretations.

Compact Truth Table

Idea: we can sometimes evaluate a formula based on values of only a subset of all variables.

subformula	
$\neg((p \to q) \land (p \land q \to r) \to (p \to r))$	
$(p \to q) \land (p \land q \to r) \to (p \to r)$	
$p \to r$	
$(p \to q) \land (p \land q \to r)$	
$p \land q \to r$	
$p \to q$	
$p \land q$	
$p \quad p \quad p$	
$q \quad q$	
$r \quad r$	

Compact Truth Table

Idea: we can sometimes evaluate a formula based on values of only a subset of all variables.

subformula	I_1
$\neg((p \to q) \land (p \land q \to r) \to (p \to r))$	
$(p \to q) \land (p \land q \to r) \to (p \to r)$	
$p \to r$	
$(p \to q) \land (p \land q \to r)$	
$p \land q \to r$	
$p \to q$	
$p \land q$	
p	
q	
r	1

Compact Truth Table

Idea: we can sometimes evaluate a formula based on values of only a subset of all variables.

subformula	I_1
$\neg((p \rightarrow q) \wedge (p \wedge q \rightarrow r) \rightarrow (p \rightarrow r))$	0
$(p \rightarrow q) \wedge (p \wedge q \rightarrow r) \rightarrow (p \rightarrow r)$	1
$p \rightarrow r$	1
$(p \rightarrow q) \wedge (p \wedge q \rightarrow r)$	
$p \wedge q \rightarrow r$	1
$p \rightarrow q$	
$p \wedge q$	
p p p	
q q	
r r	1

Compact Truth Table

Idea: we can sometimes evaluate a formula based on values of only a subset of all variables.

subformula	I_2	I_1
$\neg((p \to q) \wedge (p \wedge q \to r) \to (p \to r))$		0
$(p \to q) \wedge (p \wedge q \to r) \to (p \to r)$		1
$p \to r$		1
$(p \to q) \wedge (p \wedge q \to r)$		
$p \wedge q \to r$		1
$p \to q$		
$p \wedge q$		
p		
q		
r	0	1

Compact Truth Table

Idea: we can sometimes evaluate a formula based on values of only a subset of all variables.

subformula	I_2	I_1
$\neg((p \to q) \land (p \land q \to r) \to (p \to r))$		0
$(p \to q) \land (p \land q \to r) \to (p \to r)$		1
$p \to r$		1
$(p \to q) \land (p \land q \to r)$		
$p \land q \to r$		1
$p \to q$		
$p \land q$		
$p \quad p \quad p$	0	
$q \quad q$		
$r \quad r$	0	1

Compact Truth Table

Idea: we can sometimes evaluate a formula based on values of only a subset of all variables.

subformula	I_2	I_1
$\neg((p \to q) \wedge (p \wedge q \to r) \to (p \to r))$	0	0
$(p \to q) \wedge (p \wedge q \to r) \to (p \to r)$	1	1
$p \to r$	1	1
$(p \to q) \wedge (p \wedge q \to r)$		
$p \wedge q \to r$		1
$p \to q$		
$p \wedge q$		
$p \quad p \quad p$	0	
$q \quad q$		
$r \quad r$	0	1

Compact Truth Table

Idea: we can sometimes evaluate a formula based on values of only a subset of all variables.

subformula	I_2	I_3	I_1
$\neg((p \to q) \wedge (p \wedge q \to r) \to (p \to r))$	0		0
$(p \to q) \wedge (p \wedge q \to r) \to (p \to r)$	1		1
$p \to r$	1		1
$(p \to q) \wedge (p \wedge q \to r)$			
$p \wedge q \to r$			1
$p \to q$			
$p \wedge q$			
p p p	0	1	
q q			
r r	0	0	1

Compact Truth Table

Idea: we can sometimes evaluate a formula based on values of only a subset of all variables.

subformula	I_2	I_3	I_1
$\neg((p \rightarrow q) \wedge (p \wedge q \rightarrow r) \rightarrow (p \rightarrow r))$	0		0
$(p \rightarrow q) \wedge (p \wedge q \rightarrow r) \rightarrow (p \rightarrow r)$	1		1
$p \rightarrow r$	1	0	1
$(p \rightarrow q) \wedge (p \wedge q \rightarrow r)$			
$p \wedge q \rightarrow r$			1
$p \rightarrow q$			
$p \wedge q$			
p $\quad p$ $\quad p$	0	1	
q $\quad q$			
r $\quad r$	0	0	1

Compact Truth Table

Idea: we can sometimes evaluate a formula based on values of only a subset of all variables.

subformula	I_2	I_3	I_1
$\neg((p \to q) \land (p \land q \to r) \to (p \to r))$	0		0
$(p \to q) \land (p \land q \to r) \to (p \to r)$	1		1
$p \to r$	1	0	1
$(p \to q) \land (p \land q \to r)$			
$p \land q \to r$			1
$p \to q$			
$p \land q$			
p $\quad$ p $\quad$ p	0	1	
q $\quad$ q		0	
r $\quad$ r	0	0	1

Compact Truth Table

Idea: we can sometimes evaluate a formula based on values of only a subset of all variables.

subformula	I_2	I_3	I_1
$\neg((p \to q) \land (p \land q \to r) \to (p \to r))$	0	0	0
$(p \to q) \land (p \land q \to r) \to (p \to r)$	1	1	1
$p \to r$	1	0	1
$(p \to q) \land (p \land q \to r)$		0	
$p \land q \to r$		1	1
$p \to q$		0	
$p \land q$		0	
p	0	1	
q		0	
r	0	0	1

Compact Truth Table

Idea: we can sometimes evaluate a formula based on values of only a subset of all variables.

subformula	I_2	I_3	I_4	I_1
$\neg((p \to q) \wedge (p \wedge q \to r) \to (p \to r))$	0	0		0
$(p \to q) \wedge (p \wedge q \to r) \to (p \to r)$	1	1		1
$p \to r$	1	0		1
$(p \to q) \wedge (p \wedge q \to r)$		0		
$p \wedge q \to r$		1		1
$p \to q$		0		
$p \wedge q$		0		
p p p	0	1	1	
q q		0	1	
r r	0	0	0	1

Compact Truth Table

Idea: we can sometimes evaluate a formula based on values of only a subset of all variables.

subformula	I_2	I_3	I_4	I_1
$\neg((p \to q) \land (p \land q \to r) \to (p \to r))$	0	0	0	0
$(p \to q) \land (p \land q \to r) \to (p \to r)$	1	1	1	1
$p \to r$	1	0	0	1
$(p \to q) \land (p \land q \to r)$		0	0	
$p \land q \to r$		1	0	1
$p \to q$		0	1	
$p \land q$		0	1	
p p p	0	1	1	
q q		0	1	
r r	0	0	0	1

Compact Truth Table

Idea: we can sometimes evaluate a formula based on values of only a subset of all variables.

subformula	I_2	I_3	I_4	I_1
$\neg((p \rightarrow q) \wedge (p \wedge q \rightarrow r) \rightarrow (p \rightarrow r))$	0	0	0	0
$(p \rightarrow q) \wedge (p \wedge q \rightarrow r) \rightarrow (p \rightarrow r)$	1	1	1	1
$p \rightarrow r$	1	0	0	1
$(p \rightarrow q) \wedge (p \wedge q \rightarrow r)$		0	0	
$p \wedge q \rightarrow r$		1	0	1
$p \rightarrow q$		0	1	
$p \wedge q$		0	1	
p p p	0	1	1	
q q		0	1	
r r	0	0	0	1

The formula is unsatisfiable.

Compact Truth Table

Idea: we can sometimes evaluate a formula based on values of only a subset of all variables.

subformula	I_2	I_3	I_4	I_1
$\neg((p \to q) \land (p \land q \to r) \to (p \to r))$	0	0	0	0
$(p \to q) \land (p \land q \to r) \to (p \to r)$	1	1	1	1
$p \to r$	1	0	0	1
$(p \to q) \land (p \land q \to r)$		0	0	
$p \land q \to r$		1	0	1
$p \to q$		0	1	
$p \land q$		0	1	
p p p	0	1	1	
q q		0	1	
r r	0	0	0	1

The formula is unsatisfiable.

Note: the size of the compact table (but not the result) depends on the order of variables!

Compact Truth Table

Idea: we can sometimes evaluate a formula based on values of only a subset of all variables.

subformula	I_2	I_3	I_4	I_1
$\neg((p \rightarrow q) \wedge (p \wedge q \rightarrow r) \rightarrow (p \rightarrow r))$	0	0	0	0
$(p \rightarrow q) \wedge (p \wedge q \rightarrow r) \rightarrow (p \rightarrow r)$	1	1	1	1
$p \rightarrow r$	1	0	0	1
$(p \rightarrow q) \wedge (p \wedge q \rightarrow r)$		0	0	
$p \wedge q \rightarrow r$		1	0	1
$p \rightarrow q$		0	1	
$p \wedge q$		0	1	
$p \quad p \quad p$	0	1	1	
$q \quad q$		0	1	
$r \quad r$	0	0	0	1

The formula is unsatisfiable.
Note: the size of the compact table (but not the result) depends on the order of variables!
The ideas of guessing variable values (or case analysis) and propagation are the key ideas in nearly all propositional satisfiability algorithms.

Splitting: Idea

$A_p^\perp$ and $A_p^\top$: the formulas obtained by replacing in A all occurrences of p by $\perp$ and $\top$, respectively.

Splitting: Idea

$A_p^\perp$ and $A_p^\top$: the formulas obtained by replacing in A all occurrences of p by $\perp$ and $\top$, respectively.

Lemma

Let p be a variable, A be a formula, and I be an interpretation.

Splitting: Idea

$A_p^\perp$ and $A_p^\top$: the formulas obtained by replacing in A all occurrences of p by $\perp$ and $\top$, respectively.

Lemma

Let p be a variable, A be a formula, and I be an interpretation.

1. *If $I \not\models p$, then A is equivalent to $A_p^\perp$ in I.*
2. *If $I \models p$, then A is equivalent to $A_p^\top$ in I.*

Splitting: Idea

$A_p^\perp$ and $A_p^\top$: the formulas obtained by replacing in A all occurrences of p by $\perp$ and $\top$, respectively.

Lemma

Let p be a variable, A be a formula, and I be an interpretation.

1. *If $I \not\models p$, then A is equivalent to $A_p^\perp$ in I.*
2. *If $I \models p$, then A is equivalent to $A_p^\top$ in I.*

- Pick a variable p and perform case analysis on this variable:
  - If p is false, replace p by $\perp$;
  - If p is true, replace p by $\top$.

Splitting: Idea

$A_p^{\perp}$ and $A_p^{\top}$: the formulas obtained by replacing in A all occurrences of p by $\perp$ and $\top$, respectively.

Lemma

Let p be a variable, A be a formula, and I be an interpretation.

1. *If $I \not\models p$, then A is equivalent to $A_p^{\perp}$ in I.*
2. *If $I \models p$, then A is equivalent to $A_p^{\top}$ in I.*

- Pick a variable p and perform case analysis on this variable:
  - If p is false, replace p by $\perp$;
  - If p is true, replace p by $\top$.
- When a formula contains occurrences of $\top$ or $\perp$, simplify it.

Simplification Rules for $\top$ and $\bot$

Note: we need new simplification rules since formulas we simplify may contain propositional variables.

Simplification rules for $\top$:

$$\neg\top \Rightarrow \bot$$
$$\top \wedge A_1 \wedge \ldots \wedge A_n \Rightarrow A_1 \wedge \ldots \wedge A_n$$
$$\top \vee A_1 \vee \ldots \vee A_n \Rightarrow \top$$
$$A \rightarrow \top \Rightarrow \top \qquad \top \rightarrow A \Rightarrow A$$
$$A \leftrightarrow \top \Rightarrow A \qquad \top \leftrightarrow A \Rightarrow A$$

Simplification rules for $\bot$:

$$\neg\bot \Rightarrow \top$$
$$\bot \wedge A_1 \wedge \ldots \wedge A_n \Rightarrow \bot$$
$$\bot \vee A_1 \vee \ldots \vee A_n \Rightarrow A_1 \vee \ldots \vee A_n$$
$$A \rightarrow \bot \Rightarrow \neg A \qquad \bot \rightarrow A \Rightarrow \top$$
$$A \leftrightarrow \bot \Rightarrow \neg A \qquad \bot \leftrightarrow A \Rightarrow \neg A$$

Simplification Rules for $\top$ and $\bot$

Note: we need new simplification rules since formulas we simplify may contain propositional variables.

Simplification rules for $\top$:

$$\begin{array}{rcl} \neg\top & \Rightarrow & \bot \\ \top \wedge A_1 \wedge \ldots \wedge A_n & \Rightarrow & A_1 \wedge \ldots \wedge A_n \\ \top \vee A_1 \vee \ldots \vee A_n & \Rightarrow & \top \end{array}$$

$$A \rightarrow \top \Rightarrow \top \qquad \top \rightarrow A \Rightarrow A$$

$$A \leftrightarrow \top \Rightarrow A \qquad \top \leftrightarrow A \Rightarrow A$$

Simplification rules for $\bot$:

$$\begin{array}{rcl} \neg\bot & \Rightarrow & \top \\ \bot \wedge A_1 \wedge \ldots \wedge A_n & \Rightarrow & \bot \\ \bot \vee A_1 \vee \ldots \vee A_n & \Rightarrow & A_1 \vee \ldots \vee A_n \end{array}$$

$$A \rightarrow \bot \Rightarrow \neg A \qquad \bot \rightarrow A \Rightarrow \top$$

$$A \leftrightarrow \bot \Rightarrow \neg A \qquad \bot \leftrightarrow A \Rightarrow \neg A$$

Note that they cover all cases when $\bot$ or $\top$ occurs in the formula apart from the trivial ones.

Simplification Rules for $\top$ and $\bot$

Note: we need new simplification rules since formulas we simplify may contain propositional variables.

Simplification rules for $\top$:

$$
\begin{array}{rcl}
\neg\top & \Rightarrow & \bot \\
\top \wedge A_1 \wedge \ldots \wedge A_n & \Rightarrow & A_1 \wedge \ldots \wedge A_n \\
\top \vee A_1 \vee \ldots \vee A_n & \Rightarrow & \top
\end{array}
$$

$$
\begin{array}{rclcrcl}
A \rightarrow \top & \Rightarrow & \top & \quad & \top \rightarrow A & \Rightarrow & A \\
A \leftrightarrow \top & \Rightarrow & A & \quad & \top \leftrightarrow A & \Rightarrow & A
\end{array}
$$

Simplification rules for $\bot$:

$$
\begin{array}{rcl}
\neg\bot & \Rightarrow & \top \\
\bot \wedge A_1 \wedge \ldots \wedge A_n & \Rightarrow & \bot \\
\bot \vee A_1 \vee \ldots \vee A_n & \Rightarrow & A_1 \vee \ldots \vee A_n
\end{array}
$$

$$
\begin{array}{rclcrcl}
A \rightarrow \bot & \Rightarrow & \neg A & \quad & \bot \rightarrow A & \Rightarrow & \top \\
A \leftrightarrow \bot & \Rightarrow & \neg A & \quad & \bot \leftrightarrow A & \Rightarrow & \neg A
\end{array}
$$

Note that they cover all cases when $\bot$ or $\top$ occurs in the formula apart from the trivial ones.
Thus, if we apply these rules until they are no more applicable we obtain either $\bot$, or $\top$, or a formula containing neither $\bot$ nor $\top$.

Splitting Algorithm

```
procedure split(G)
parameters: function select
input: formula G
output: "satisfiable" or "unsatisfiable"
begin
  G := simplify(G)
  if G = ⊤ then return "satisfiable"
  if G = ⊥ then return "unsatisfiable"
  (p, b) := select(G)
  case b of
  1 ⇒
    if split(G_p^⊤) ="satisfiable"
      then return "satisfiable"
      else return split(G_p^⊥)
  0 ⇒
    if split(G_p^⊥) ="satisfiable"
      then return "satisfiable"
      else return split(G_p^⊤)
end
```

Splitting Algorithm, Example

$$\neg((p \to q) \land (p \land q \to r) \to (p \to r))$$

Splitting Algorithm, Example

$$\neg((p \to q) \land (p \land q \to r) \to (p \to r))$$

$q = 0$

$$\neg((p \to \bot) \land (p \land \bot \to r) \to (p \to r))$$

Splitting Algorithm, Example

$$\neg((p \to q) \land (p \land q \to r) \to (p \to r))$$

$q = 0$

$$\neg((p \to \bot) \land (p \land \bot \to r) \to (p \to r))$$

$$\neg(\neg p \to (p \to r))$$

Splitting Algorithm, Example

$$\neg((p \to q) \land (p \land q \to r) \to (p \to r))$$

$q = 0$

$$\neg((p \to \bot) \land (p \land \bot \to r) \to (p \to r))$$

$$\neg(\neg p \to (p \to r))$$

$p = 1$

$$\neg(\neg \top \to (\top \to r))$$

Splitting Algorithm, Example

$$\neg((p \to q) \land (p \land q \to r) \to (p \to r))$$

$q = 0$

$$\neg((p \to \bot) \land (p \land \bot \to r) \to (p \to r))$$

$$\neg(\neg p \to (p \to r))$$

$p = 1$

$$\neg(\neg\top \to (\top \to r))$$

$$\bot$$

Splitting Algorithm, Example

$$\neg((p \to q) \land (p \land q \to r) \to (p \to r))$$

$q = 0$

$$\neg((p \to \bot) \land (p \land \bot \to r) \to (p \to r))$$

$$\neg(\neg p \to (p \to r))$$

$p = 1$ $\qquad$ $p = 0$

$$\neg(\neg\top \to (\top \to r)) \qquad \neg(\neg\bot \to (\bot \to r))$$

$$\bot$$

Splitting Algorithm, Example

$$\neg((p \to q) \wedge (p \wedge q \to r) \to (p \to r))$$

$q = 0$

$$\neg((p \to \bot) \wedge (p \wedge \bot \to r) \to (p \to r))$$

$$\neg(\neg p \to (p \to r))$$

$p = 1$

$$\neg(\neg\top \to (\top \to r))$$

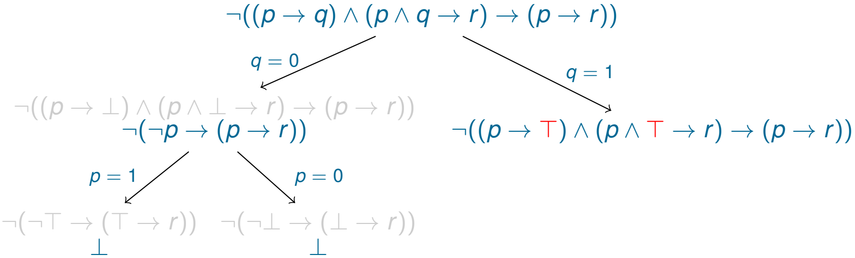
$$\bot$$

$p = 0$

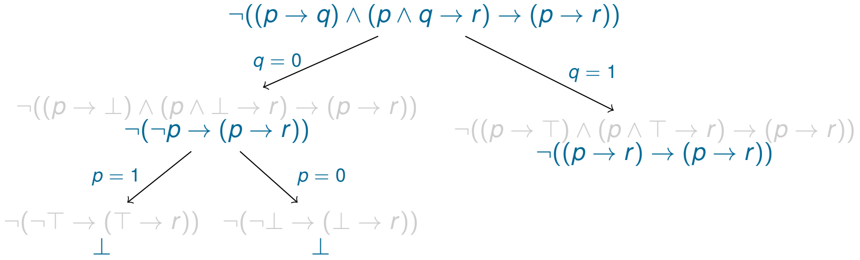
$$\neg(\neg\bot \to (\bot \to r))$$

$$\bot$$

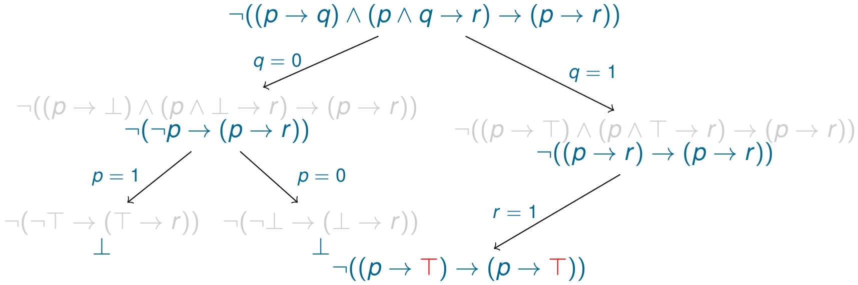
Splitting Algorithm, Example



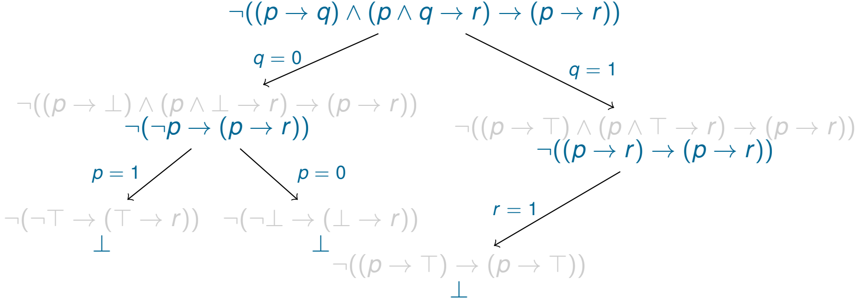
Splitting Algorithm, Example



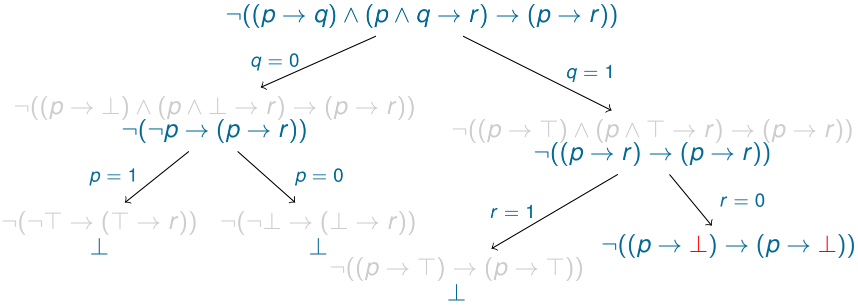
Splitting Algorithm, Example



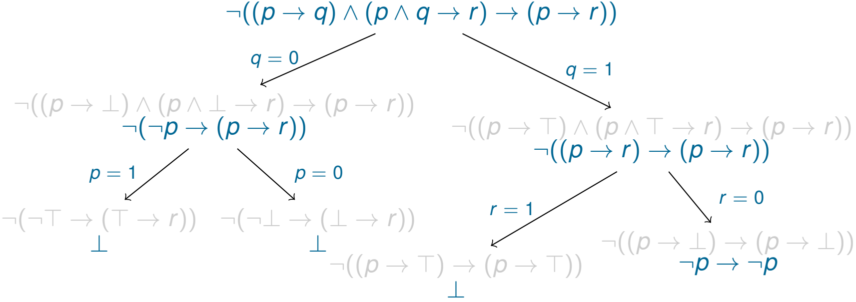
Splitting Algorithm, Example



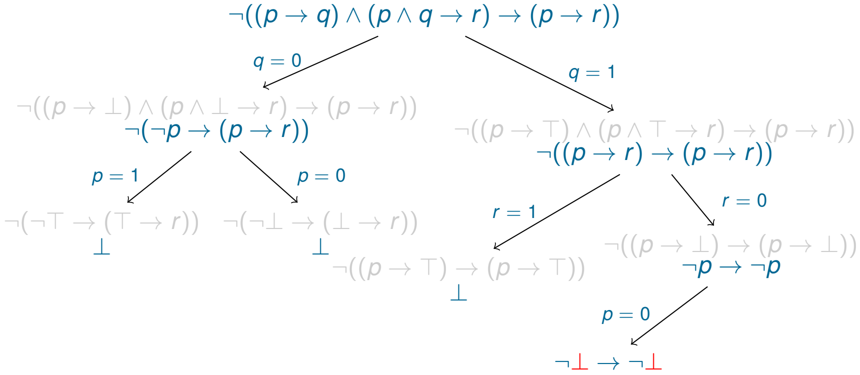
Splitting Algorithm, Example



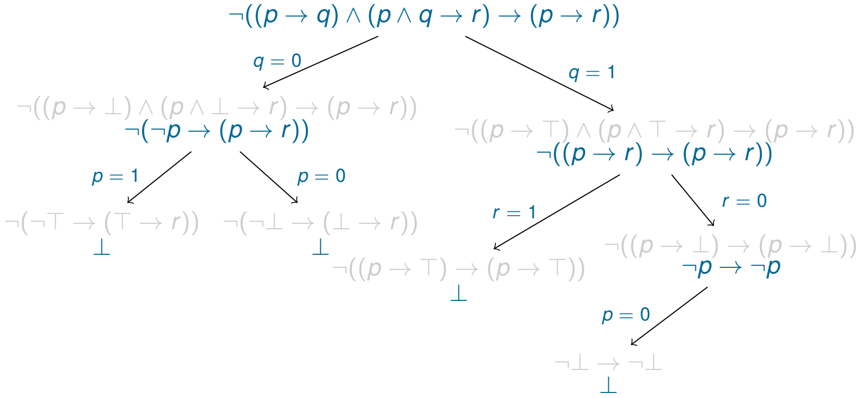
Splitting Algorithm, Example



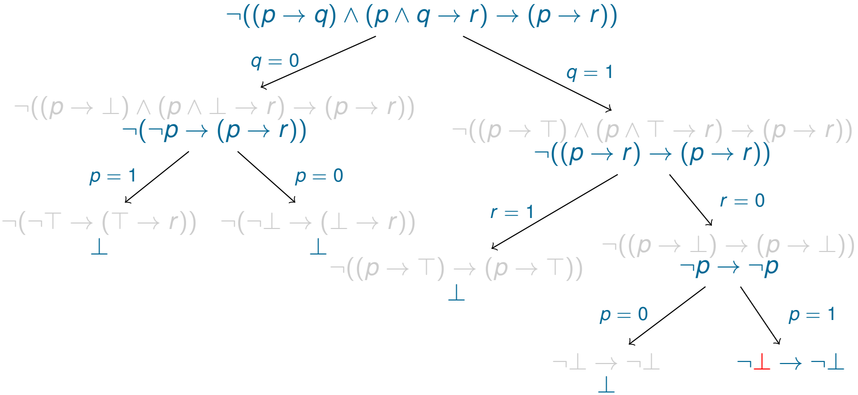
Splitting Algorithm, Example



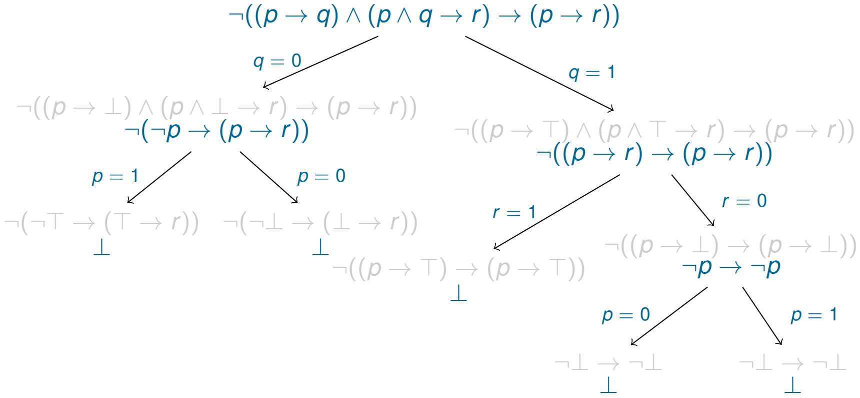
Splitting Algorithm, Example



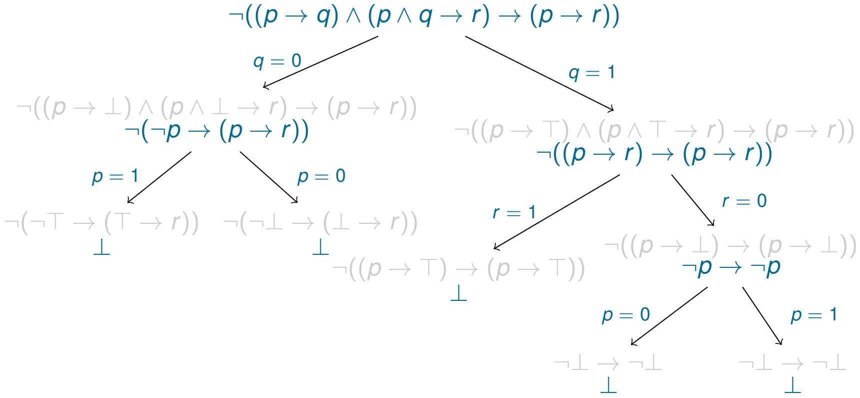
Splitting Algorithm, Example



Splitting Algorithm, Example

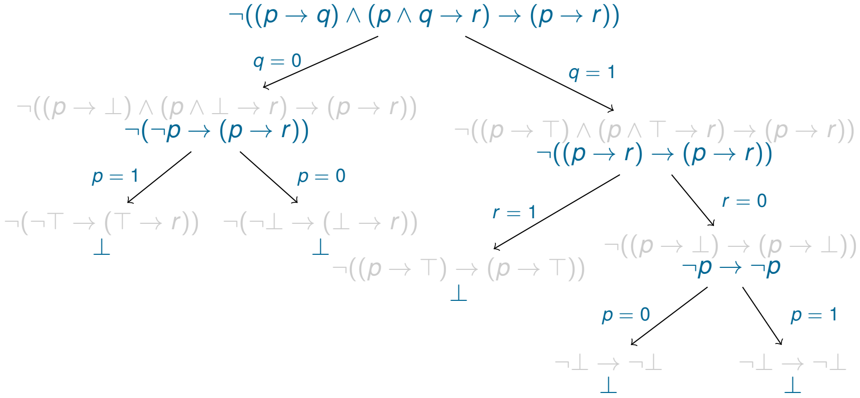


Splitting Algorithm, Example



The formula is unsatisfiable.

Splitting Algorithm, Example



The formula is unsatisfiable.

What this algorithm does is essentially the same as compact truth tables, but on the syntactic level.

Splitting Algorithm, Example 2

$$\neg((p \to q) \land (p \land q \to r) \to (\neg p \to r))$$

Splitting Algorithm, Example 2

$$\neg((p \to q) \land (p \land q \to r) \to (\neg p \to r))$$

$$\downarrow p = 0$$

$$\neg((\bot \to q) \land (\bot \land q \to r) \to (\neg \bot \to r))$$

Splitting Algorithm, Example 2

$$\neg((p \to q) \land (p \land q \to r) \to (\neg p \to r))$$

$$\Big\downarrow \; p = 0$$

$$\neg((\bot \to q) \land (\bot \land q \to r) \to (\neg\bot \to r))$$

$$\neg r$$

Splitting Algorithm, Example 2

$$\neg((p \to q) \land (p \land q \to r) \to (\neg p \to r))$$

$p = 0$ ↓

$$\neg((\bot \to q) \land (\bot \land q \to r) \to (\neg \bot \to r))$$

$$\neg r$$

$r = 0$ ↓

$$\neg \bot$$

Splitting Algorithm, Example 2

$$\neg((p \rightarrow q) \land (p \land q \rightarrow r) \rightarrow (\neg p \rightarrow r))$$

$p = 0$ $\downarrow$

$$\neg((\bot \rightarrow q) \land (\bot \land q \rightarrow r) \rightarrow (\neg\bot \rightarrow r))$$

$$\neg r$$

$r = 0$ $\downarrow$

$$\neg\bot$$

$$\top$$

Splitting Algorithm, Example 2

$$\neg((p \to q) \land (p \land q \to r) \to (\neg p \to r))$$

$p = 0 \downarrow$

$$\neg((\bot \to q) \land (\bot \land q \to r) \to (\neg\bot \to r))$$

$$\neg r$$

$r = 0 \downarrow$

$$\neg\bot$$

$$\top$$

The formula is satisfiable.

Splitting Algorithm, Example 2

$$\neg((p \to q) \land (p \land q \to r) \to (\neg p \to r))$$

$$\downarrow \; p = 0$$

$$\neg((\bot \to q) \land (\bot \land q \to r) \to (\neg \bot \to r))$$

$$\neg r$$

$$\downarrow \; r = 0$$

$$\neg \bot$$

$$\top$$

The formula is satisfiable.

To find a model of this formula, we should simply collect choices made on the branch terminating at $\top$.

Splitting Algorithm, Example 2

$$\neg((p \to q) \wedge (p \wedge q \to r) \to (\neg p \to r))$$

$$\Big\downarrow \; p = 0$$

$$\neg((\bot \to q) \wedge (\bot \wedge q \to r) \to (\neg \bot \to r))$$

$$\neg r$$

$$\Big\downarrow \; r = 0$$

$$\neg \bot$$

$$\top$$

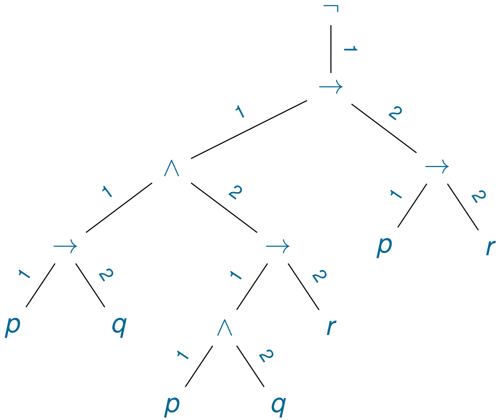
The formula is satisfiable.

To find a model of this formula, we should simply collect choices made on the branch terminating at $\top$.

Any interpretation I such that $I(p) = I(r) = 0$ satisfies the formula, for example the interpretation $\{p \mapsto 0, q \mapsto 0, r \mapsto 0\}$.

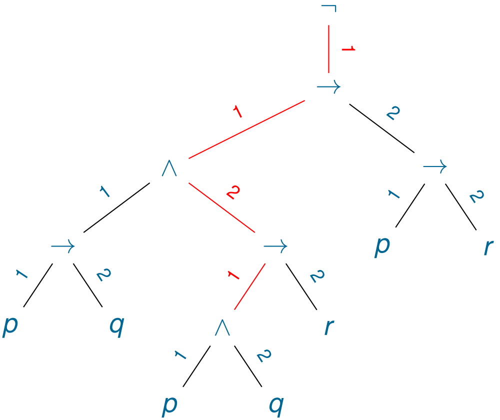
Parse Tree

$A \stackrel{\text{def}}{=} \neg((p \to q) \land (p \land q \to r) \to (p \to r)).$



Parse Tree

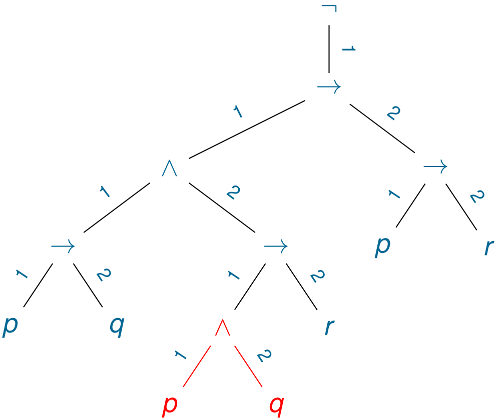
$A \stackrel{\text{def}}{=} \neg((p \to q) \wedge (p \wedge q \to r) \to (p \to r))$.



- Position in the formula: 1.1.2.1;

Parse Tree

$A \stackrel{\text{def}}{=} \neg((p \to q) \land (p \land q \to r) \to (p \to r))$.



- Position in the formula: 1.1.2.1;
- Subformula at this position: $p \land q$.

Positions and Subformulas

- Position is any sequence of positive integers $a_1, \ldots, a_n$, where $n \geq 0$, written as $a_1.a_2. \cdots .a_n$.
- Empty position, denoted by ϵ: when $n = 0$.
- Position π in a formula A, subformula at a position, denoted $A|_\pi$.

Positions and Subformulas

- Position is any sequence of positive integers $a_1, \ldots, a_n$, where $n \geq 0$, written as $a_1.a_2. \cdots .a_n$.
- Empty position, denoted by ϵ: when $n = 0$.
- Position π in a formula A, subformula at a position, denoted $A|_\pi$.

1. For every formula A, ϵ is a position in A and $A|_\epsilon \stackrel{\text{def}}{=} A$.
2. Let $A|_\pi = B$.
   2.1 If B has the form $B_1 \wedge \ldots \wedge B_n$ or $B_1 \vee \ldots \vee B_n$, then for all $i \in \{1, \ldots, n\}$ the position $\pi.i$ is a position in A, $A|_{\pi.i} \stackrel{\text{def}}{=} B_i$.
   2.2 If B has the form $\neg B_1$, then $\pi.1$ is a position in A, $A|_{\pi.1} \stackrel{\text{def}}{=} B_1$.
   2.3 If B has the form $B_1 \rightarrow B_2$, then $\pi.1$ and $\pi.2$ are positions in A and we have $A|_{\pi.1} \stackrel{\text{def}}{=} B_1$, $A|_{\pi.2} \stackrel{\text{def}}{=} B_2$;
   2.4 If B has the form $B_1 \leftrightarrow B_2$, then $\pi.1$ and $\pi.2$ are positions in A and $A|_{\pi.i} \stackrel{\text{def}}{=} B_i$.

If $A|_\pi = B$, we also say that B occurs in A at the position π.

Polarity

1. For every formula A, ϵ is a position in A, $A|_\epsilon \stackrel{\text{def}}{=} A$

2. Let $A|_\pi = B$.

   2.1 If B has the form $B_1 \wedge \ldots \wedge B_n$ or $B_1 \vee \ldots \vee B_n$, then for all $i \in \{1, \ldots, n\}$ the position $\pi.i$ is a position in A, $A|_{\pi.i} \stackrel{\text{def}}{=} B_i$

   2.2 If B has the form $\neg B_1$, then $\pi.1$ is a position in A, $A|_{\pi.1} \stackrel{\text{def}}{=} B_1$

   2.3 If B has the form $B_1 \rightarrow B_2$, then $\pi.1$ and $\pi.2$ are positions in A and we have $A|_{\pi.1} \stackrel{\text{def}}{=} B_1$, $A|_{\pi.2} \stackrel{\text{def}}{=} B_2$

   2.4 If B has the form $B_1 \leftrightarrow B_2$, then $\pi.1$ and $\pi.2$ are positions in A and $A|_{\pi.i} \stackrel{\text{def}}{=} B_i$ for $i = 1, 2$.

Polarity

Polarity of subformula at a position. Notation: $pol(A, \pi)$.

1. For every formula A, ϵ is a position in A, $A|_\epsilon \stackrel{\text{def}}{=} A$

2. Let $A|_\pi = B$.

   2.1 If B has the form $B_1 \wedge \ldots \wedge B_n$ or $B_1 \vee \ldots \vee B_n$, then for all $i \in \{1, \ldots, n\}$ the position $\pi.i$ is a position in A, $A|_{\pi.i} \stackrel{\text{def}}{=} B_i$

   2.2 If B has the form $\neg B_1$, then $\pi.1$ is a position in A, $A|_{\pi.1} \stackrel{\text{def}}{=} B_1$

   2.3 If B has the form $B_1 \rightarrow B_2$, then $\pi.1$ and $\pi.2$ are positions in A and we have $A|_{\pi.1} \stackrel{\text{def}}{=} B_1$, $A|_{\pi.2} \stackrel{\text{def}}{=} B_2$

   2.4 If B has the form $B_1 \leftrightarrow B_2$, then $\pi.1$ and $\pi.2$ are positions in A and $A|_{\pi.i} \stackrel{\text{def}}{=} B_i$ for $i = 1, 2$.

Polarity

Polarity of subformula at a position. Notation: $pol(A, \pi)$.

1. For every formula A, ϵ is a position in A, $A|_\epsilon \stackrel{\text{def}}{=} A$ and $pol(A, \epsilon) \stackrel{\text{def}}{=} 1$.
2. Let $A|_\pi = B$.
   - 2.1 If B has the form $B_1 \land \ldots \land B_n$ or $B_1 \lor \ldots \lor B_n$, then for all $i \in \{1, \ldots, n\}$ the position $\pi.i$ is a position in A, $A|_{\pi.i} \stackrel{\text{def}}{=} B_i$, and $pol(A, \pi.i) \stackrel{\text{def}}{=} pol(A, \pi)$.
   - 2.2 If B has the form $\neg B_1$, then $\pi.1$ is a position in A, $A|_{\pi.1} \stackrel{\text{def}}{=} B_1$ and $pol(A, \pi.1) \stackrel{\text{def}}{=} -pol(A, \pi)$.
   - 2.3 If B has the form $B_1 \to B_2$, then $\pi.1$ and $\pi.2$ are positions in A and we have $A|_{\pi.1} \stackrel{\text{def}}{=} B_1$, $A|_{\pi.2} \stackrel{\text{def}}{=} B_2$, $pol(A, \pi.1) \stackrel{\text{def}}{=} -pol(A, \pi)$, $pol(A, \pi.2) \stackrel{\text{def}}{=} pol(A, \pi)$.
   - 2.4 If B has the form $B_1 \leftrightarrow B_2$, then $\pi.1$ and $\pi.2$ are positions in A and $A|_{\pi.i} \stackrel{\text{def}}{=} B_i$ and $pol(A, \pi.i) \stackrel{\text{def}}{=} 0$ for $i = 1, 2$.

Polarity

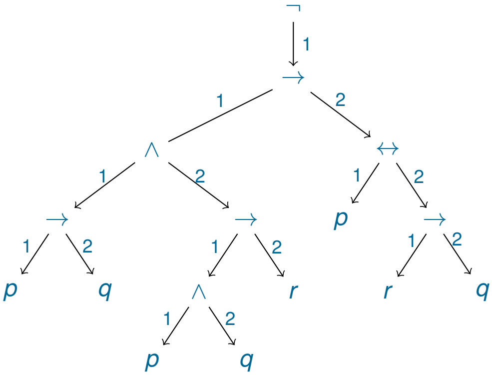
Polarity of subformula at a position. Notation: $pol(A, \pi)$.

1. For every formula A, ϵ is a position in A, $A|_\epsilon \stackrel{\text{def}}{=} A$ and $pol(A, \epsilon) \stackrel{\text{def}}{=} 1$.
2. Let $A|_\pi = B$.
   - 2.1 If B has the form $B_1 \wedge \ldots \wedge B_n$ or $B_1 \vee \ldots \vee B_n$, then for all $i \in \{1, \ldots, n\}$ the position $\pi.i$ is a position in A, $A|_{\pi.i} \stackrel{\text{def}}{=} B_i$, and $pol(A, \pi.i) \stackrel{\text{def}}{=} pol(A, \pi)$.
   - 2.2 If B has the form $\neg B_1$, then $\pi.1$ is a position in A, $A|_{\pi.1} \stackrel{\text{def}}{=} B_1$ and $pol(A, \pi.1) \stackrel{\text{def}}{=} -pol(A, \pi)$.
   - 2.3 If B has the form $B_1 \rightarrow B_2$, then $\pi.1$ and $\pi.2$ are positions in A and we have $A|_{\pi.1} \stackrel{\text{def}}{=} B_1$, $A|_{\pi.2} \stackrel{\text{def}}{=} B_2$, $pol(A, \pi.1) \stackrel{\text{def}}{=} -pol(A, \pi)$, $pol(A, \pi.2) \stackrel{\text{def}}{=} pol(A, \pi)$.
   - 2.4 If B has the form $B_1 \leftrightarrow B_2$, then $\pi.1$ and $\pi.2$ are positions in A and $A|_{\pi.i} \stackrel{\text{def}}{=} B_i$ and $pol(A, \pi.i) \stackrel{\text{def}}{=} 0$ for $i = 1, 2$.

- If $pol(A, \pi) = 1$ and $A|_\pi = B$, then we call the occurrence of B at the position π in A positive.
- If $pol(A, \pi) = -1$ and $A|_\pi = B$, then we call the occurrence of B at the position π in A negative.

The Colouring Algorithm for Determining Polarity

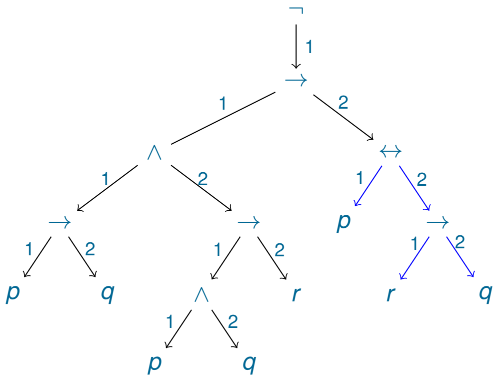
$\neg((p \to q) \land (p \land q \to r) \to (p \leftrightarrow (r \to q)))$.



The Colouring Algorithm for Determining Polarity

$\neg((p \rightarrow q) \wedge (p \wedge q \rightarrow r) \rightarrow (p \leftrightarrow (r \rightarrow q)))$.

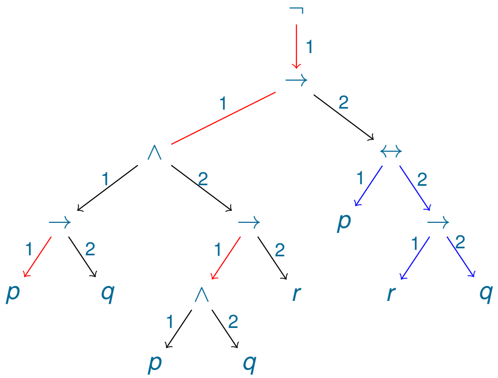
- Color in blue all arcs below an equivalence.



The Colouring Algorithm for Determining Polarity

$\neg((p \to q) \land (p \land q \to r) \to (p \leftrightarrow (r \to q)))$.

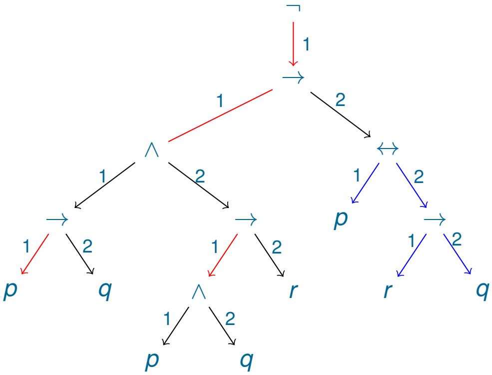
- Color in blue all arcs below an equivalence.
- Color in red all uncoloured arcs going down from a negation or left-hand side of an implication.



The Colouring Algorithm for Determining Polarity

$\neg((p \to q) \land (p \land q \to r) \to (p \leftrightarrow (r \to q)))$.

- Color in blue all arcs below an equivalence.
- Color in red all uncoloured arcs going down from a negation or left-hand side of an implication.

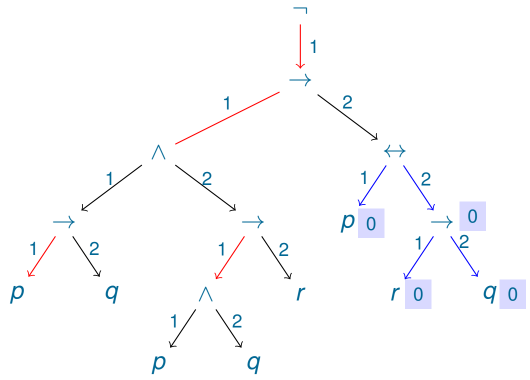


- If a position has at least one blue arc above it, its polarity is 0.

The Colouring Algorithm for Determining Polarity

$\neg((p \to q) \land (p \land q \to r) \to (p \leftrightarrow (r \to q)))$.

- Color in blue all arcs below an equivalence.
- Color in red all uncoloured arcs going down from a negation or left-hand side of an implication.

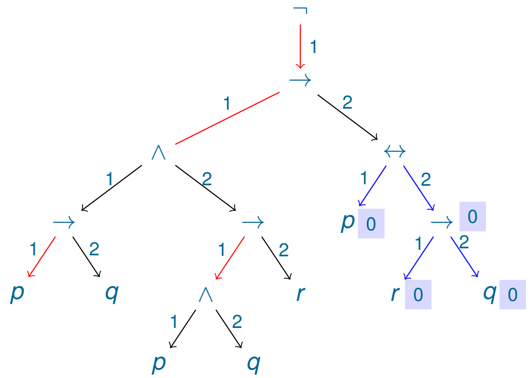


- If a position has at least one blue arc above it, its polarity is 0.

The Colouring Algorithm for Determining Polarity

$\neg((p \to q) \land (p \land q \to r) \to (p \leftrightarrow (r \to q)))$.

- Color in blue all arcs below an equivalence.
- Color in red all uncoloured arcs going down from a negation or left-hand side of an implication.

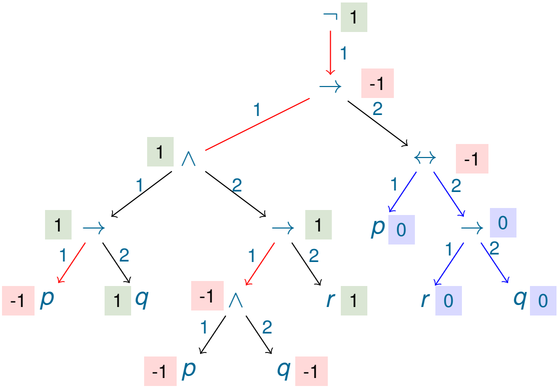


- If a position has at least one blue arc above it, its polarity is 0.
- Otherwise, its polarity is -1 if it has an odd number of red arcs above it.

The Colouring Algorithm for Determining Polarity

$\neg((p \to q) \land (p \land q \to r) \to (p \leftrightarrow (r \to q)))$.

- Color in blue all arcs below an equivalence.
- Color in red all uncoloured arcs going down from a negation or left-hand side of an implication.



- If a position has at least one blue arc above it, its polarity is 0.
- Otherwise, its polarity is -1 if it has an odd number of red arcs above it.

Position and Polarity, Again

position	subformula	polarity
ϵ	$\neg((p \to q) \wedge (p \wedge q \to r) \to (p \to r))$	1
1	$(p \to q) \wedge (p \wedge q \to r) \to (p \to r)$	-1
1.1	$(p \to q) \wedge (p \wedge q \to r)$	1
1.1.1	$p \to q$	1
1.1.1.1	p	-1
1.1.1.2	q	1
1.1.2	$p \wedge q \to r$	1
1.1.2.1	$p \wedge q$	-1
1.1.2.1.1	p	-1
1.1.2.1.2	q	-1
1.1.2.2	r	1
1.2	$p \to r$	-1
1.2.1	p	1
1.2.2	r	-1

Monotonic Replacement

Notation: $A[B]_\pi$:

- formula A with the subformula B at the position π;
- formula A with the subformula at the position π replaced by B.

Monotonic Replacement

Notation: $A[B]_\pi$:

- formula A with the subformula B at the position π;
- formula A with the subformula at the position π replaced by B.

Lemma (Monotonic Replacement)

Let A, B, B' be formulas, I be an interpretation, and $I \models B \rightarrow B'$. If $pol(A, \pi) = 1$, then $I \models A[B]_\pi \rightarrow A[B']_\pi$. Likewise, if $pol(A, \pi) = -1$, then $I \models A[B']_\pi \rightarrow A[B]_\pi$.

Monotonic Replacement

Notation: $A[B]_\pi$:

- formula A with the subformula B at the position π;
- formula A with the subformula at the position π replaced by B.

Lemma (Monotonic Replacement)

Let A, B, B' be formulas, I be an interpretation, and $I \models B \rightarrow B'$. If $pol(A, \pi) = 1$, then $I \models A[B]_\pi \rightarrow A[B']_\pi$. Likewise, if $pol(A, \pi) = -1$, then $I \models A[B']_\pi \rightarrow A[B]_\pi$.

While monotonic? Note that $I \models B \rightarrow B'$ is the same as $I(B) \leq I(B')$.

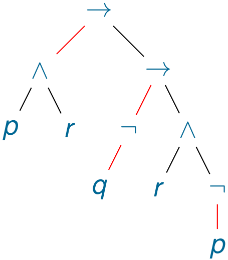
Pure Variable

Variable p is pure in a formula A, if either all occurrences of p in A are positive or all occurrences of p in A are negative.

Pure Variable

Variable p is pure in a formula A, if either all occurrences of p in A are positive or all occurrences of p in A are negative.

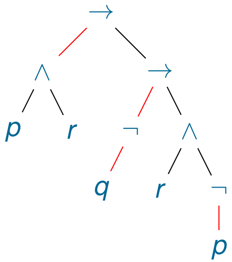
$$p \land r \to (\neg q \to (r \land \neg p))$$



Pure Variable

Variable p is pure in a formula A, if either all occurrences of p in A are positive or all occurrences of p in A are negative.

$$p \land r \rightarrow (\neg q \rightarrow (r \land \neg p))$$

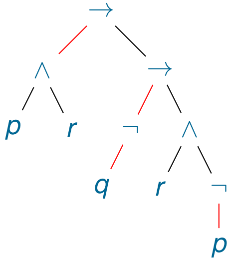


- Both occurrences of p are negative, so p is pure.

Pure Variable

Variable p is pure in a formula A, if either all occurrences of p in A are positive or all occurrences of p in A are negative.

$$p \land r \to (\neg q \to (r \land \neg p))$$

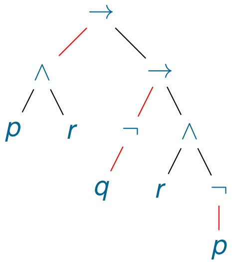


- Both occurrences of p are negative, so p is pure.
- The only occurrence of q is positive, so q is pure.

Pure Variable

Variable p is pure in a formula A, if either all occurrences of p in A are positive or all occurrences of p in A are negative.

$$p \wedge r \rightarrow (\neg q \rightarrow (r \wedge \neg p))$$



- Both occurrences of p are negative, so p is pure.
- The only occurrence of q is positive, so q is pure.
- r is not pure, since it has both negative and positive occurrences.

Properties of Pure Variables

Lemma (Pure Variable)

Let p has only positive occurrences in A and $I \models A$. Define

$$I' \stackrel{\text{def}}{=} I + (p \mapsto 1)$$

Then $I' \models A$.

Properties of Pure Variables

Lemma (Pure Variable)

Let p has only positive occurrences in A and $I \models A$. Define

$$I' \stackrel{\text{def}}{=} I + (p \mapsto 1)$$

Then $I' \models A$.
Likewise, let p has only negative occurrences in A and $I \models A$. Define

$$I' \stackrel{\text{def}}{=} I + (p \mapsto 0)$$

Then $I' \models A$.

Properties of Pure Variables

Lemma (Pure Variable)

Let p has only positive occurrences in A and $I \models A$. Define

$$I' \stackrel{\text{def}}{=} I + (p \mapsto 1)$$

Then $I' \models A$.
Likewise, let p has only negative occurrences in A and $I \models A$. Define

$$I' \stackrel{\text{def}}{=} I + (p \mapsto 0)$$

Then $I' \models A$.

Theorem (Pure Variable)

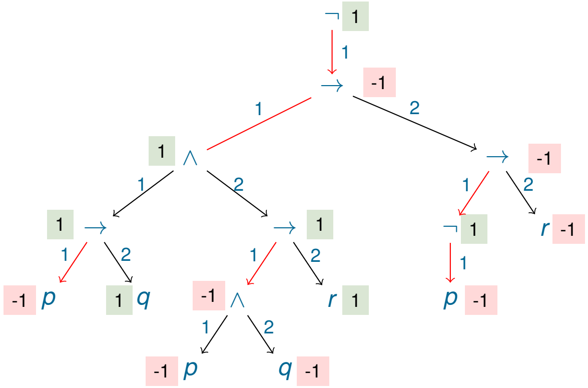
Let a variable p has only positive (respectively, only negative) occurrences in A. Then A is satisfiable if and only if so is $A_p^\top$ (respectively, $A_p^\perp$).

Pure Variable, Example

Consider $\neg((p \to q) \land (p \land q \to r) \to (\neg p \to r))$.

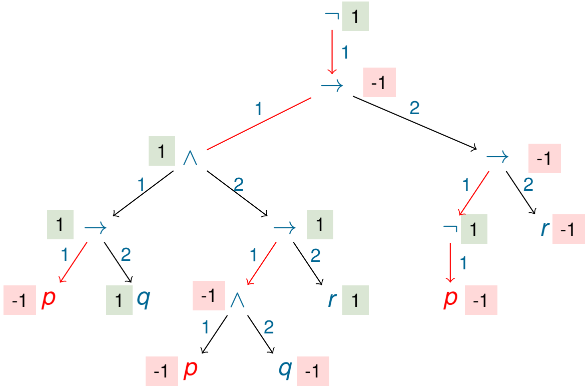
Pure Variable, Example

Consider $\neg((p \to q) \wedge (p \wedge q \to r) \to (\neg p \to r))$.



Pure Variable, Example

Consider $\neg((p \to q) \land (p \land q \to r) \to (\neg p \to r))$.



All occurrences of p are negative, so, for the purpose of checking satisfiability we can replace p by $\bot$.

Example, Continued

$$\neg((p \to q) \wedge (p \wedge q \to r) \to (\neg p \to r))$$

All occurrences of p are negative

Example, Continued

$$\neg((p \to q) \land (p \land q \to r) \to (\neg p \to r)) \quad \Rightarrow$$
$$\neg((\bot \to q) \land (\bot \land q \to r) \to (\neg\bot \to r))$$

All occurrences of p are negative, so, for the purpose of checking satisfiability we can replace p by $\bot$.

Example, Continued

$$\neg((p \to q) \land (p \land q \to r) \to (\neg p \to r)) \quad \Rightarrow$$
$$\neg((\bot \to q) \land (\bot \land q \to r) \to (\neg \bot \to r)) \quad \Rightarrow$$
$$\neg(\top \land (\bot \land q \to r) \to (\neg \bot \to r))$$

Example, Continued

$$\begin{aligned}
\neg((p \to q) \land (p \land q \to r) \to (\neg p \to r)) &\quad \Rightarrow \\
\neg((\bot \to q) \land (\bot \land q \to r) \to (\neg \bot \to r)) &\quad \Rightarrow \\
\neg(\top \land (\bot \land q \to r) \to (\neg \bot \to r)) &\quad \Rightarrow \\
\neg((\bot \land q \to r) \to (\neg \bot \to r)) &
\end{aligned}$$

Example, Continued

$$
\begin{array}{rl}
\neg((p \to q) \land (p \land q \to r) \to (\neg p \to r)) & \Rightarrow \\
\neg((\bot \to q) \land (\bot \land q \to r) \to (\neg \bot \to r)) & \Rightarrow \\
\neg(\top \land (\bot \land q \to r) \to (\neg \bot \to r)) & \Rightarrow \\
\neg((\bot \land q \to r) \to (\neg \bot \to r)) & \Rightarrow \\
\neg((\bot \to r) \to (\neg \bot \to r)) &
\end{array}
$$

Example, Continued

$$\neg((p \to q) \land (p \land q \to r) \to (\neg p \to r)) \quad \Rightarrow$$
$$\neg((\bot \to q) \land (\bot \land q \to r) \to (\neg\bot \to r)) \quad \Rightarrow$$
$$\neg(\top \land (\bot \land q \to r) \to (\neg\bot \to r)) \quad \Rightarrow$$
$$\neg((\bot \land q \to r) \to (\neg\bot \to r)) \quad \Rightarrow$$
$$\neg((\bot \to r) \to (\neg\bot \to r)) \quad \Rightarrow$$
$$\neg(\top \to (\neg\bot \to r))$$

Example, Continued

$$\begin{array}{rl}
\neg((p \to q) \land (p \land q \to r) \to (\neg p \to r)) & \Rightarrow \\
\neg((\bot \to q) \land (\bot \land q \to r) \to (\neg\bot \to r)) & \Rightarrow \\
\neg(\top \land (\bot \land q \to r) \to (\neg\bot \to r)) & \Rightarrow \\
\neg((\bot \land q \to r) \to (\neg\bot \to r)) & \Rightarrow \\
\neg((\bot \to r) \to (\neg\bot \to r)) & \Rightarrow \\
\neg(\top \to (\neg\bot \to r)) & \Rightarrow \\
\neg(\neg\bot \to r) &
\end{array}$$

Example, Continued

$$
\begin{array}{cc}
\neg((p \to q) \land (p \land q \to r) \to (\neg p \to r)) & \Rightarrow \\
\neg((\bot \to q) \land (\bot \land q \to r) \to (\neg \bot \to r)) & \Rightarrow \\
\neg(\top \land (\bot \land q \to r) \to (\neg \bot \to r)) & \Rightarrow \\
\neg((\bot \land q \to r) \to (\neg \bot \to r)) & \Rightarrow \\
\neg((\bot \to r) \to (\neg \bot \to r)) & \Rightarrow \\
\neg(\top \to (\neg \bot \to r)) & \Rightarrow \\
\neg(\neg \bot \to r) & \Rightarrow \\
\neg(\top \to r) &
\end{array}
$$

Example, Continued

$$\neg((p \to q) \land (p \land q \to r) \to (\neg p \to r)) \quad \Rightarrow$$
$$\neg((\bot \to q) \land (\bot \land q \to r) \to (\neg \bot \to r)) \quad \Rightarrow$$
$$\neg(\top \land (\bot \land q \to r) \to (\neg \bot \to r)) \quad \Rightarrow$$
$$\neg((\bot \land q \to r) \to (\neg \bot \to r)) \quad \Rightarrow$$
$$\neg((\bot \to r) \to (\neg \bot \to r)) \quad \Rightarrow$$
$$\neg(\top \to (\neg \bot \to r)) \quad \Rightarrow$$
$$\neg(\neg \bot \to r) \quad \Rightarrow$$
$$\neg(\top \to r) \quad \Rightarrow$$
$$\neg r$$

After the simplification all occurrences of r are negative

Example, Continued

$$
\begin{array}{cc}
\neg((p \to q) \land (p \land q \to r) \to (\neg p \to r)) & \Rightarrow \\
\neg((\bot \to q) \land (\bot \land q \to r) \to (\neg \bot \to r)) & \Rightarrow \\
\neg(\top \land (\bot \land q \to r) \to (\neg \bot \to r)) & \Rightarrow \\
\neg((\bot \land q \to r) \to (\neg \bot \to r)) & \Rightarrow \\
\neg((\bot \to r) \to (\neg \bot \to r)) & \Rightarrow \\
\neg(\top \to (\neg \bot \to r)) & \Rightarrow \\
\neg(\neg \bot \to r) & \Rightarrow \\
\neg(\top \to r) & \Rightarrow \\
\neg r & \Rightarrow \\
\neg \bot &
\end{array}
$$

After the simplification all occurrences of r are negative, so, for the purpose of checking satisfiability we can replace r by $\bot$.

Example, Continued

$$\neg((p \to q) \land (p \land q \to r) \to (\neg p \to r)) \quad \Rightarrow$$
$$\neg((\bot \to q) \land (\bot \land q \to r) \to (\neg\bot \to r)) \quad \Rightarrow$$
$$\neg(\top \land (\bot \land q \to r) \to (\neg\bot \to r)) \quad \Rightarrow$$
$$\neg((\bot \land q \to r) \to (\neg\bot \to r)) \quad \Rightarrow$$
$$\neg((\bot \to r) \to (\neg\bot \to r)) \quad \Rightarrow$$
$$\neg(\top \to (\neg\bot \to r)) \quad \Rightarrow$$
$$\neg(\neg\bot \to r) \quad \Rightarrow$$
$$\neg(\top \to r) \quad \Rightarrow$$
$$\neg r \quad \Rightarrow$$
$$\neg\bot \quad \Rightarrow$$
$$\top$$

We have shown satisfiability of this formula deterministically, using only the pure variable rule.

End of Lecture 4

Slides for lecture 4 end here . . .