Propositional Logic for Forgetters

Uwe Egly

Vienna University of Technology
Institute of Information Systems
Knowledge-Based Systems Group

Outline

Human Reasoning

- From given knowledge W, infer new knowledge B
  Make implicitly represented knowledge explicit
- If possible then (W, B) is in the inference relation R, i.e.,

$$(W, B) \in R : \quad (\text{raining}, \text{street\_is\_wet}) \in R$$

- Otherwise, it is not, i.e.,

$$(W, B) \notin R : \quad (\text{raining}, \text{light\_is\_green}) \notin R$$

- How can a computer perform inferences (i.e., decide R)?
  - ➥ Represent W and B and
  - ➥ apply deduction methods on the representation of W and B

Human Reasoning

- From given knowledge W, infer new knowledge B
  Make implicitly represented knowledge explicit
- If possible then (W, B) is in the inference relation R, i.e.,

$$(W, B) \in R : \quad (\text{raining}, \text{street\_is\_wet}) \in R$$

- Otherwise, it is not, i.e.,

$$(W, B) \notin R : \quad (\text{raining}, \text{light\_is\_green}) \notin R$$

- How can a computer perform inferences (i.e., decide R)?
  - ➥ Represent W and B and
  - ➥ apply deduction methods on the representation of W and B

Formalizing a Logical System

...or the ingredients for an automated inference system

- Old dream of mankind
  (At least of some "logicians" like G. W. Leibniz (1646–1716))
- Goal of artificial intelligence
- What is needed?
  - Syntax (= construction rules for formulas)
  - Semantics (= meaning of formulas)
  - Inference rules
  - Correctness (no false conclusions are drawn)
  - Completeness (all correct conclusions are drawn)

Formalizing a Logical System

... or the ingredients for an automated inference system

- Old dream of mankind
  (At least of some “logicians” like G. W. Leibniz (1646–1716))
- Goal of artificial intelligence
- What is needed?
  - Syntax (= construction rules for formulas)
  - Semantics (= meaning of formulas)
  - Inference rules
  - Correctness (no false conclusions are drawn)
  - Completeness (all correct conclusions are drawn)

The Syntax of PL0

- Given a countable set, $\mathcal{BV}$, of boolean variables
- Boolean variables like $p, q, r, p_1, \ldots$ represent facts: e.g., r represents “it is raining”
- Inductive definition of the set of propositional formulas

  B1: Every $p \in \mathcal{BV}$ is a formula (called atomic formula or atom)
  B2: $\top$ (verum) and $\bot$ (falsum) are formulas
  S1: If ϕ is a formula, then so is $(\neg\phi)$ (negation)
  S2: If ϕ_1 and ϕ_2 are formulas, then so are $(\phi_1 \wedge \phi_2)$ (conjunction), $(\phi_1 \vee \phi_2)$ (disjunction), $(\phi_1 \to \phi_2)$ (implication) and $(\phi_1 \leftrightarrow \phi_2)$ (equivalence)

- To save parenthesis, use the following ranking of binding strength: $\neg$, $\wedge$, $\vee$, $\to$, $\leftrightarrow$ (i.e., $\neg$ binds stronger than $\wedge$, etc.)
- Example: $\neg p \wedge q \to r \vee s$ means $((\neg p) \wedge q) \to (r \vee s)$

The Syntax of PL0

- Given a countable set, $\mathcal{BV}$, of boolean variables
- Boolean variables like $p, q, r, p_1, \ldots$ represent facts: e.g., r represents "it is raining"
- Inductive definition of the set of propositional formulas
  - B1: Every $p \in \mathcal{BV}$ is a formula (called atomic formula or atom)
  - B2: $\top$ (verum) and $\bot$ (falsum) are formulas
  - S1: If ϕ is a formula, then so is $(\neg\phi)$ (negation)
  - S2: If ϕ_1 and ϕ_2 are formulas, then so are $(\phi_1 \wedge \phi_2)$ (conjunction), $(\phi_1 \vee \phi_2)$ (disjunction), $(\phi_1 \rightarrow \phi_2)$ (implication) and $(\phi_1 \leftrightarrow \phi_2)$ (equivalence)
- To save parenthesis, use the following ranking of binding strength: $\neg$, $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$ (i.e., $\neg$ binds stronger than $\wedge$, etc.)
- Example: $\neg p \wedge q \rightarrow r \vee s$ means $((\neg p) \wedge q) \rightarrow (r \vee s)$

The Syntax of PL0

- Given a countable set, $\mathcal{BV}$, of boolean variables
- Boolean variables like $p, q, r, p_1, \ldots$ represent facts: e.g., r represents "it is raining"
- Inductive definition of the set of propositional formulas

  B1: Every $p \in \mathcal{BV}$ is a formula (called atomic formula or atom)
  B2: $\top$ (verum) and $\bot$ (falsum) are formulas
  S1: If ϕ is a formula, then so is $(\neg\phi)$ (negation)
  S2: If ϕ_1 and ϕ_2 are formulas, then so are $(\phi_1 \wedge \phi_2)$ (conjunction), $(\phi_1 \vee \phi_2)$ (disjunction), $(\phi_1 \rightarrow \phi_2)$ (implication) and $(\phi_1 \leftrightarrow \phi_2)$ (equivalence)

- To save parenthesis, use the following ranking of binding strength: $\neg$, $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$ (i.e., $\neg$ binds stronger than $\wedge$, etc.)
- Example: $\neg p \wedge q \rightarrow r \vee s$ means $((\neg p) \wedge q) \rightarrow (r \vee s)$

The Syntax of PL0 Cont'd

- A literal is an atom or a negated atom
- A clause is a disjunction of literals
- Immediate subformula (relation)
  - ϕ is an immediate subformula (isf) of $\neg\phi$
  - ϕ_1 and ϕ_2 are isfs of $(\phi_1 \circ \phi_2)$ for $\circ \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$
- Subformula relation: reflexive-transitive closure of the immediate subformula relation
- Example: all subformulas of $\neg p \wedge q \rightarrow r \vee s$

  $\neg p \wedge q \rightarrow r \vee s$, $\neg p \wedge q$, $r \vee s$, $\neg p$, q, r, s, p

The Syntax of PL0 Cont'd

- A literal is an atom or a negated atom
- A clause is a disjunction of literals
- Immediate subformula (relation)
  - ϕ is an immediate subformula (isf) of $\neg\phi$
  - ϕ_1 and ϕ_2 are isfs of $(\phi_1 \circ \phi_2)$ for $\circ \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$
- Subformula relation: reflexive-transitive closure of the immediate subformula relation
- Example: all subformulas of $\neg p \wedge q \rightarrow r \vee s$
  $\neg p \wedge q \rightarrow r \vee s$, $\neg p \wedge q$, $r \vee s$, $\neg p$, q, r, s, p

The Syntax of PL0 Cont'd

- A literal is an atom or a negated atom
- A clause is a disjunction of literals
- Immediate subformula (relation)
  - ϕ is an immediate subformula (isf) of $\neg\phi$
  - ϕ_1 and ϕ_2 are isfs of $(\phi_1 \circ \phi_2)$ for $\circ \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$
- Subformula relation: reflexive-transitive closure of the immediate subformula relation
- Example: all subformulas of $\neg p \wedge q \rightarrow r \vee s$

  $\neg p \wedge q \rightarrow r \vee s$, $\neg p \wedge q$, $r \vee s$, $\neg p$, q, r, s, p

The Syntax of PL0 Cont'd

- A literal is an atom or a negated atom
- A clause is a disjunction of literals
- Immediate subformula (relation)
  - ϕ is an immediate subformula (isf) of $\neg\phi$
  - ϕ_1 and ϕ_2 are isfs of $(\phi_1 \circ \phi_2)$ for $\circ \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$
- Subformula relation: reflexive-transitive closure of the immediate subformula relation
- Example: all subformulas of $\neg p \wedge q \rightarrow r \vee s$

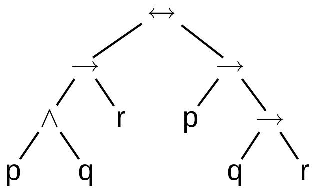
  $\neg p \wedge q \rightarrow r \vee s$, $\neg p \wedge q$, $r \vee s$, $\neg p$, q, r, s, p

The Syntax of PL0 Cont'd

- A literal is an atom or a negated atom
- A clause is a disjunction of literals
- Immediate subformula (relation)
  - ϕ is an immediate subformula (isf) of $\neg\phi$
  - ϕ_1 and ϕ_2 are isfs of $(\phi_1 \circ \phi_2)$ for $\circ \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$
- Subformula relation: reflexive-transitive closure of the immediate subformula relation
- Example: all subformulas of $\neg p \wedge q \rightarrow r \vee s$
  $\neg p \wedge q \rightarrow r \vee s$, $\neg p \wedge q$, $r \vee s$, $\neg p$, q, r, s, p

Propositional Formulas as Trees

- Formulas can be depicted as formula trees
- Example: $((p \wedge q) \rightarrow r) \leftrightarrow (p \rightarrow (q \rightarrow r))$



The Semantics of PL0

- A boolean value (or truth value) is either 0 (false) or 1 (true)
- An interpretation function for a set $\mathcal{P}$ of boolean variables:

$$\text{mapping} \quad I\colon \mathcal{P} \mapsto \{0, 1\}$$

- Interpretation functions are often called truth assignments
  e.g., I assigns 1 to $p \in \mathcal{P}$, i.e., $I(p) = 1$
- Since we want to "evaluate" formulas under I, we have to extend I to formulas

The Extension of I to Formulas

- $I(\top) = 1$ and $I(\bot) = 0$
- $I(\neg\phi) = 1$ iff $I(\phi) = 0$
- $I(\phi \wedge \psi) = 1$ iff $I(\phi) = I(\psi) = 1$
- $I(\phi \vee \psi) = 1$ iff $I(\phi) = 1$ or $I(\psi) = 1$
- $I(\phi \rightarrow \psi) = 1$ iff $I(\phi) = 0$ or $I(\psi) = 1$
- $I(\phi \leftrightarrow \psi) = 1$ iff $I(\phi) = I(\psi)$

Equivalent notations:

ϕ is true under I	iff	I satisfies ϕ	iff	$I(\phi) = 1$	iff	$I \models \phi$
ϕ is false under I	iff	I does not satisfy ϕ	iff	$I(\phi) = 0$	iff	$I \not\models \phi$

Truth Tables

- Another representation of the extension of I from above
- One table for each connective

$\neg$	
1	0
0	1

$\wedge$	1	0
1	1	0
0	0	0

$\vee$	1	0
1	1	1
0	1	0

$\rightarrow$	1	0
1	1	0
0	1	1

$\leftrightarrow$	1	0
1	1	0
0	0	1

Even More Notations

- If I satisfies ϕ, we call I a model of ϕ
- $Mod(\psi)$ is the set of all models of ψ
- ϕ is satisfiable (valid) if ϕ is true in some (all) interpretations
- ϕ is unsatisfiable if ϕ is false in all interpretations
- Formulas ϕ and ψ are equivalent, denoted by $\phi \equiv \psi$, iff they have exactly the same models, i.e., $Mod(\phi) = Mod(\psi)$
- Example: $(\phi \rightarrow \psi) \equiv (\neg\psi \rightarrow \neg\phi)$: Take $I \in Mod(\phi \rightarrow \psi)$

$$
\begin{aligned}
I(\phi \rightarrow \psi) = 1 \quad & \text{iff } I(\phi) = 0 \text{ or } I(\psi) = 1 \\
& \text{iff } I(\neg\phi) = 1 \text{ or } I(\psi) = 1 \\
& \text{iff } I(\neg\phi) = 1 \text{ or } I(\neg\psi) = 0 \\
& \text{iff } I(\neg\psi \rightarrow \neg\phi) = 1
\end{aligned}
$$

Some Useful Equivalences

Commutativity	$\phi \circ \psi$	$\equiv$	$\psi \circ \phi$	for $\circ \in \{\vee, \wedge, \leftrightarrow\}$
Idempotence	$\phi \circ \phi$	$\equiv$	ϕ	for $\circ \in \{\vee, \wedge\}$
Tautology	$\phi \vee \top$	$\equiv$	$\top$	
Unsatisfiability	$\phi \wedge \bot$	$\equiv$	$\bot$	
Neutrality	$\phi \wedge \top$	$\equiv$	ϕ	
	$\phi \vee \bot$	$\equiv$	ϕ	
Negation	$\phi \vee \neg\phi$	$\equiv$	$\top$	
	$\phi \wedge \neg\phi$	$\equiv$	$\bot$	
Double Negation	$\neg\neg\phi$	$\equiv$	ϕ	
Implication	$\phi \rightarrow \psi$	$\equiv$	$\neg\phi \vee \psi$	

Some Useful Equivalences Cont'd

De Morgan	$\neg(\phi \vee \psi)$	$\equiv$	$\neg\phi \wedge \neg\psi$
	$\neg(\phi \wedge \psi)$	$\equiv$	$\neg\phi \vee \neg\psi$
Absorption	$\phi \vee (\phi \wedge \psi)$	$\equiv$	ϕ
	$\phi \wedge (\phi \vee \psi)$	$\equiv$	ϕ
Distributivity	$\phi \wedge (\psi \vee \chi)$	$\equiv$	$(\phi \wedge \psi) \vee (\phi \wedge \chi)$
	$\phi \vee (\psi \wedge \chi)$	$\equiv$	$(\phi \vee \psi) \wedge (\phi \vee \chi)$
Associativity	$\phi \vee (\psi \vee \chi)$	$\equiv$	$(\phi \vee \psi) \vee \chi$
	$\phi \wedge (\psi \wedge \chi)$	$\equiv$	$(\phi \wedge \psi) \wedge \chi$

Connections Between the Different Notations

- Distinguish between
  - tautologies: all interpretations are models
  - satisfiable formulas: some interpretations are models
  - contradictions: no interpretation is a model
- A formula ϕ is valid iff $\neg\phi$ is unsatisfiable
  Recall that this is the basis for a "proof by contradiction"
- A formula ϕ is satisfiable iff $\neg\phi$ is not valid
- Two formulas ϕ and ψ are equivalent iff $\phi \leftrightarrow \psi$ is valid
- A formula ϕ is valid iff ϕ is equivalent to $\top$
- A formula ϕ is unsatisfiable iff ϕ is equivalent to $\bot$

Entailment

- So far, $\models$ relates an interpretation and a formula
- Allow also a set of formulas on the left side
- Important: a set of formulas coincides with the conjunction of its elements, i.e., $\{\phi_1, \ldots, \phi_n\}$ is $\bigwedge_{i=1}^{n} \phi_i$
- Important: an empty conjunction is 1 in all interpretations i.e., it is equivalent to $\top$
- W entails ϕ, $W \models \phi$, iff $Mod(W) \subseteq Mod(\phi)$
- $W \models \phi$ iff $I \models \phi$ for all models I of W (for all $I \in Mod(W)$)
- Important for KBSs: Does KB W entails query ϕ

Properties of Entailment

- $W \models \psi$ implies $W \cup \{\phi\} \models \psi$ — Monotonicity for PL0
- $W \cup \{\phi\} \models \psi$ iff $W \models \phi \rightarrow \psi$ — Deduction Thm
- $W \cup \{\phi\} \models \neg\psi$ iff $W \cup \{\psi\} \models \neg\phi$ — Contraposition Thm
- $W \cup \{\phi\}$ is unsatisfiable iff $W \models \neg\phi$ — Contradiction Thm

Important Questions to Answer

Q1 How can we algorithmicly check whether $I \models \phi$ holds?

- ➥ Evaluation of a formula ϕ under a given interpretation I

Q2 When does $W \models \phi$ hold?

Q2' When does $\models \phi$ hold?

- This is a special case of Q2 with empty W!
- Q2' asks: when is ϕ a tautology (1 in all interpretations)?

Q3 Can we define deduction (based on syntactic rules), s.t. the result of deduction and entailment coincides?

- Write $W \vdash_i \phi$ if ϕ is derived from W by deduction proc. i
- i is sound: if $W \vdash_i \phi$ holds, then $W \models \phi$ also holds
- i is complete: if $W \models \phi$ holds, then $W \vdash_i \phi$ also holds

Evaluation of a Formula Under an Interpretation

Evaluate ϕ: $(p \to q) \wedge (p \wedge q \to r) \to (p \to r)$ under $I = \{p \mapsto 1, q \mapsto 0, r \mapsto 1\}$ starting with p

	formula	value
1	$(p \to q) \wedge (p \wedge q \to r) \to (p \to r)$	1
2	$p \to r$	1
3	$(p \to q) \wedge (p \wedge q \to r)$	0
4	$p \wedge q \to r$	1
5	$p \to q$	0
6	$p \wedge q$	0
7	p p p	1
8	q q	0
9	r r	1

Evaluation of a Formula Under an Interpretation

Evaluate ϕ: $(p \to q) \land (p \land q \to r) \to (p \to r)$ under $I = \{p \mapsto 1, q \mapsto 0, r \mapsto 1\}$ starting with p

	formula	value
1	$(p \to q) \land (p \land q \to r) \to (p \to r)$	1
2	$p \to r$	1
3	$(p \to q) \land (p \land q \to r)$	0
4	$p \land q \to r$	1
5	$p \to q$	0
6	$p \land q$	0
7	p p p	1
8	q q	0
9	r r	1

Evaluation of a Formula Under an Interpretation

Evaluate ϕ: $(p \to q) \land (p \land q \to r) \to (p \to r)$ under $I = \{p \mapsto 1, q \mapsto 0, r \mapsto 1\}$ starting with p

	formula	value
1	$(p \to q) \land (p \land q \to r) \to (p \to r)$	1
2	$p \to r$	1
3	$(p \to q) \land (p \land q \to r)$	0
4	$p \land q \to r$	1
5	$p \to q$	0
6	$p \land q$	0
7	p $\quad p$ $\quad p$	1
8	q $\quad q$	0
9	r $\quad r$	1

Evaluation of a Formula Under an Interpretation

Evaluate ϕ: $(p \to q) \wedge (p \wedge q \to r) \to (p \to r)$ under $I = \{p \mapsto 1, q \mapsto 0, r \mapsto 1\}$ starting with p

	formula	value
1	$(p \to q) \wedge (p \wedge q \to r) \to (p \to r)$	1
2	$p \to r$	1
3	$(p \to q) \wedge (p \wedge q \to r)$	0
4	$p \wedge q \to r$	1
5	$p \to q$	0
6	$p \wedge q$	0
7	p p p	1
8	q q	0
9	r r	1

Evaluation of a Formula Under an Interpretation

Evaluate ϕ: $(p \to q) \land (p \land q \to r) \to (p \to r)$ under $I = \{p \mapsto 1, q \mapsto 0, r \mapsto 1\}$ starting with p

	formula	value
1	$(p \to q) \land (p \land q \to r) \to (p \to r)$	1
2	$p \to r$	1
3	$(p \to q) \land (p \land q \to r)$	0
4	$p \land q \to r$	1
5	$p \to q$	0
6	$p \land q$	0
7	p p p	1
8	q q	0
9	r r	1

Evaluation of a Formula Under an Interpretation

Evaluate ϕ: $(p \to q) \land (p \land q \to r) \to (p \to r)$ under $I = \{p \mapsto 1, q \mapsto 0, r \mapsto 1\}$ starting with p

	formula	value
1	$(p \to q) \land (p \land q \to r) \to (p \to r)$	1
2	$p \to r$	1
3	$(p \to q) \land (p \land q \to r)$	0
4	$p \land q \to r$	1
5	$p \to q$	0
6	$p \land q$	0
7	p p p	1
8	q q	0
9	r r	1

Evaluation of a Formula Under an Interpretation

Evaluate ϕ: $(p \to q) \land (p \land q \to r) \to (p \to r)$ under $I = \{p \mapsto 1, q \mapsto 0, r \mapsto 1\}$ starting with p

	formula	value
1	$(p \to q) \land (p \land q \to r) \to (p \to r)$	1
2	$p \to r$	1
3	$(p \to q) \land (p \land q \to r)$	0
4	$p \land q \to r$	1
5	$p \to q$	0
6	$p \land q$	0
7	p p p	1
8	q q	0
9	r r	1

Evaluation of a Formula Under an Interpretation Cont'd

Evaluate ϕ: $(p \to q) \land (p \land q \to r) \to (p \to r)$ under $I = \{p \mapsto 1, q \mapsto 0, r \mapsto 1\}$ starting with r

	formula	value
1	$(p \to q) \land (p \land q \to r) \to (p \to r)$	1
2	$p \to r$	1
3	$(p \to q) \land (p \land q \to r)$	0
4	$p \land q \to r$	1
5	$p \to q$	0
6	$p \land q$	0
7	p p p	1
8	q q	0
9	r r	1

Compare former evaluation with current one: order matters!

Evaluation of a Formula Under an Interpretation Cont'd

Evaluate ϕ: $(p \to q) \wedge (p \wedge q \to r) \to (p \to r)$ under
$I = \{p \mapsto 1, q \mapsto 0, r \mapsto 1\}$ starting with r

	formula	value
1	$(p \to q) \wedge (p \wedge q \to r) \to (p \to r)$	1
2	$p \to r$	1
3	$(p \to q) \wedge (p \wedge q \to r)$	0
4	$p \wedge q \to r$	1
5	$p \to q$	0
6	$p \wedge q$	0
7	p p p	1
8	q q	0
9	r r	1

Compare former evaluation with current one: order matters!

Evaluation of a Formula Under an Interpretation Cont'd

Evaluate ϕ: $(p \to q) \land (p \land q \to r) \to (p \to r)$ under
$I = \{p \mapsto 1, q \mapsto 0, r \mapsto 1\}$ starting with r

	formula	value
1	$(p \to q) \land (p \land q \to r) \to (p \to r)$	1
2	$p \to r$	1
3	$(p \to q) \land (p \land q \to r)$	0
4	$p \land q \to r$	1
5	$p \to q$	0
6	$p \land q$	0
7	p p p	1
8	q q	0
9	r r	1

Compare former evaluation with current one: order matters!

Evaluation of a Formula Under an Interpretation Cont'd

Evaluate ϕ: $(p \to q) \land (p \land q \to r) \to (p \to r)$ under $I = \{p \mapsto 1, q \mapsto 0, r \mapsto 1\}$ starting with r

	formula	value
1	$(p \to q) \land (p \land q \to r) \to (p \to r)$	1
2	$p \to r$	1
3	$(p \to q) \land (p \land q \to r)$	0
4	$p \land q \to r$	1
5	$p \to q$	0
6	$p \land q$	0
7	p $\quad p$ $\quad p$	1
8	q $\quad q$	0
9	r $\quad r$	1

Compare former evaluation with current one: order matters!

How to Check Entailments $W \models \phi$ and $\models \phi$

How can we use the evaluation procedure to answer Q2'?

- Consider formula ϕ with $\{p_1, \ldots, p_n\}$ as prop. variables
- Generate all 2^n I over these n variables one after the other
- Test each of the generated interpretation I
  - Evaluate with the above procedure ϕ under the current I
  - If $I(\phi) = 0$ then STOP and answer "ϕ is not a tautology"
- If $I(\phi) = 1$ for all I then STOP and answer "ϕ is a tautology"

How is Q2 handled (i.e., $W \models \phi$)?

- Assume W is $\{\psi_1, \ldots, \psi_m\}$ and apply the Deduction Thm
- Obtain, e.g., $\models \psi_1 \rightarrow (\psi_2 \rightarrow (\cdots(\psi_m \rightarrow \phi)\cdots))$

How to Check Entailments $W \models \phi$ and $\models \phi$

How can we use the evaluation procedure to answer Q2'?

- Consider formula ϕ with $\{p_1, \ldots, p_n\}$ as prop. variables
- Generate all 2^n I over these n variables one after the other
- Test each of the generated interpretation I
  - Evaluate with the above procedure ϕ under the current I
  - If $I(\phi) = 0$ then STOP and answer "ϕ is not a tautology"
- If $I(\phi) = 1$ for all I then STOP and answer "ϕ is a tautology"

How is Q2 handled (i.e., $W \models \phi$)?

- Assume W is $\{\psi_1, \ldots, \psi_m\}$ and apply the Deduction Thm
- Obtain, e.g., $\models \psi_1 \rightarrow (\psi_2 \rightarrow (\cdots (\psi_m \rightarrow \phi) \cdots))$

Decision Procedure for PL0

- Check proc. is based on enumeration of interpretations
  - For n atoms in the input formula: 2^n interpretations possible
  - Worst case: all 2^n interpretations have to be generated
- Check procedure for validity is a Decision Procedure
  - Possible answer is either yes or no
  - Decides entailment (and validity)
  - Decides satisfiability by reduction approach:
    ϕ is satisfiable iff $\neg\phi$ is not valid (i.e., $\neg\phi$ is not a tautology)
- Can we do better/differently?

Reduction to Satisfiability

Reduce validity, entailment, equivalence to satisfiability

1. Validity
   - $\neg\phi$ is unsatisfiable iff ϕ is valid
2. Entailment
   - ϕ entails ψ ($\phi \models \psi$) iff $\phi \rightarrow \psi$ is valid (apply Deduction Thm)
   - Hence, $\phi \models \psi$ iff $\phi \wedge \neg\psi$ (i.e., $\neg(\phi \rightarrow \psi)$) is unsatisfiable
3. Equivalence
   - ϕ is equivalent to ψ ($\phi \equiv \psi$) iff $\phi \leftrightarrow \psi$ is valid
   - Hence, $\phi \equiv \psi$ iff $\phi \models \psi$ and $\psi \models \phi$ hold
   - Consequently, $\phi \equiv \psi$ iff $\phi \wedge \neg\psi$ and $\psi \wedge \neg\phi$ are unsatisfiable

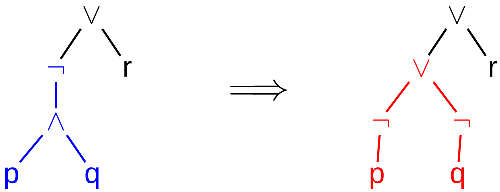
Sound and complete procedure for satisfiability is sufficient!

The Tableau Calculus for PL0 (TC0)

- TC0 is a decision procedure for satisfiability
  - Answer yes means satisfiable
  - Answer no means unsatisfiable
- For a satisfiable input formula ϕ, TC0 constructs model of ϕ
- For simplicity: Restrict the input formula to be in negation normal form (NNF)
- NNF characterized by two conditions:
  1. Negation signs occur only in front of atoms
  2. The only connectives are $\wedge$ and $\vee$
- NNF of ϕ (denoted by $nnf(\phi)$) and ϕ are equivalent!
- Translation procedures are available

Preparatory Concept for NNF: Tree Replacements

- Let $\phi[\psi]$ denote that ψ occurs in ϕ
- This means that ψ occurs as zero, one or more subtree(s) in the formula tree of ϕ
- Possibility to perform subtree replacements
- Example: $\phi: \neg(p \wedge q) \vee r$ and $\psi_1: \neg(p \wedge q)$
  - Then $\phi[\psi_1]$ indicates occurrence(s) of ψ_1 in ϕ
  - Construct $\phi[\psi_2]$ with $\psi_2: \neg p \vee \neg q$ by a tree replacement



Equivalent Replacement

Lemma (Equivalent Replacement Lemma)

Let I be an interpretation and $I \models \psi_1 \leftrightarrow \psi_2$*. Then* $I \models \phi[\psi_1] \leftrightarrow \phi[\psi_2]$*.*

Theorem (Equivalent Replacement Theorem)

Let $\psi_1 \equiv \psi_2$*. Then* $\phi[\psi_1] \equiv \phi[\psi_2]$*.*

Basics of the NNF Translation

- Replace $\leftrightarrow$ by $\rightarrow$ using $(\phi \leftrightarrow \psi) \equiv ((\phi \rightarrow \psi) \wedge (\psi \rightarrow \phi))$
- Replace $\rightarrow$ using $(\phi \rightarrow \psi) \equiv (\neg\phi \vee \psi)$
- Replace the left side of the following equivalences by the right side (order does not matter!)

$$
\begin{array}{rcl@{\qquad}rcl}
\neg(\phi \vee \psi) & \equiv & \neg\phi \wedge \neg\psi & \neg(\phi \wedge \psi) & \equiv & \neg\phi \vee \neg\psi \\
\phi \vee \top & \equiv & \top & \top \vee \phi & \equiv & \top \\
\phi \wedge \bot & \equiv & \bot & \bot \wedge \phi & \equiv & \bot \\
\phi \wedge \top & \equiv & \phi & \top \wedge \phi & \equiv & \phi \\
\phi \vee \bot & \equiv & \phi & \bot \vee \phi & \equiv & \phi \\
\neg\neg\phi & \equiv & \phi & & &
\end{array}
$$

- Translation process is terminating with the NNF

The Basic Ideas for TC0

- Decompose formulas into their immediate subformulas
- This decomposition results in a decomposition tree
- Exhaustively look at all possibilities of decomposition
- Look for clashes on the branches of the trees
- Clash on a branch (i.e, p and $\neg p$ are on the same branch) indicates that this branch is unsatifiable
- If all branches have a clash: input formula is unsatisfiable
- Eventually, a branch without clash provides a model

The Basic Steps of TC0

Input: a formula in NNF

1. Start with the input formula as the root of the refutation tree
2. Starting from the root, extend the tree by new formulas generated by completion (inference) rules
3. Two kinds of completion rules:
   - deterministic: add successor nodes to the same branch
   - nondeterministic: several alternative successor nodes (increases the number of branches)
4. Apply the completion rules until either
   - an explicit contradiction (clash) occurs in each branch or
   - there is a completed branch (no more rules are applicable)

The Completion (Inference) Rules of TC0

$$\frac{\phi \wedge \psi}{\begin{array}{c}\phi \\ \psi\end{array}}$$

If a model satisfies the conjunction, then it satisfies each of the conjuncts

$$\frac{\phi \vee \psi}{\phi \mid \psi}$$

If a model satisfies a disjunction, then it also satisfies one of the disjuncts. It is a non-deterministic rule, and it generates two alternative branches of the tableaux.

When are the Completion Rules Applicable?

$$\frac{\phi \wedge \psi}{\begin{array}{c}\phi \\ \psi\end{array}}$$

This rule can be applied if ϕ and ψ are not both on the current branch

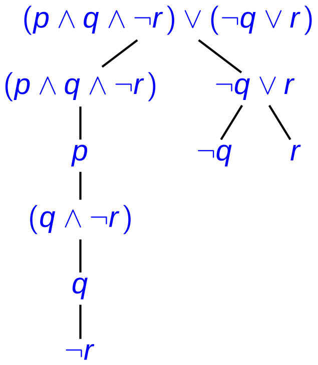
$$\frac{\phi \vee \psi}{\phi \mid \psi}$$

This rule can be applied if neither ϕ nor ψ is on the current branch

- Applicability conditions prevent redundant rule applications
- Necessary for termination

Example: Is ϕ: $((p \wedge q) \rightarrow r) \rightarrow (q \rightarrow r)$ satisfiable?

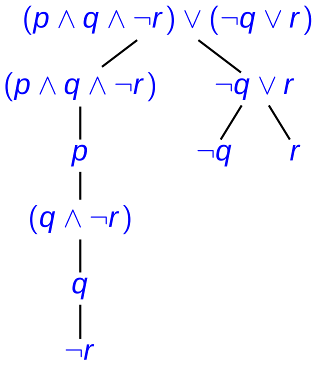
$nnf(\phi)$: $(p \wedge q \wedge \neg r) \vee (\neg q \vee r)$



- All branches are completed
- Generate models for $nnf(\phi)$
  - Take completed branch b: set every literal on b to 1
  - Left branch: $I(p) = I(q) = 1, I(r) = 0$
  - Middle brach: $I(q) = 0$
  - Right branch: $I(r) = 1$
- Are these all models?
- Only falsifying interpretation: $I(p) = I(r) = 0, I(q) = 1$

Example: Is ϕ: $((p \wedge q) \rightarrow r) \rightarrow (q \rightarrow r)$ satisfiable?

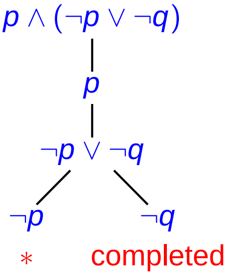
$nnf(\phi)$: $(p \wedge q \wedge \neg r) \vee (\neg q \vee r)$



- All branches are completed
- Generate models for $nnf(\phi)$
  - Take completed branch b: set every literal on b to 1
  - Left branch: $I(p) = I(q) = 1, I(r) = 0$
  - Middle brach: $I(q) = 0$
  - Right branch: $I(r) = 1$
- Are these all models?
- Only falsifying interpretation: $I(p) = I(r) = 0$, $I(q) = 1$

Example: Does $p \models p \wedge q$ hold?

- $p \models p \wedge q$ holds iff $p \wedge \neg(p \wedge q)$ is unsatisfiable
- Check for satisfiability of $nnf(p \wedge \neg(p \wedge q))$



- Left branch has a clash
- Right branch is completed
- Generate counter model I for the entailment $p \models p \wedge q$
  - Right branch: $I(p) = 1$ and $I(q) = 0$
  - Check: $I \models p$ but $I \not\models p \wedge q$
  - Consequently, $Mod(p) \not\subseteq Mod(p \wedge q)$

TC0 as a Decision Procedure

TC0 is a decision procedure for computing satisfiability, validity, and entailment in PL0

- TC0 is purely syntactical (decomposition of formulas)
- TC0 is sound
- TC0 is complete
- TC0 is terminating

Worst-case complexity depend on:

- the structure of the input formula for TC0
- the number of atoms in the input formula for truth tables

Summary (of PL0)

We recapitulated important definitions and notations for PL0 like

- formulas, atoms, literals, clauses,
- subformulas and formula trees,
- the concept of an interpretation and models,
- validity, (un)satisfiability, and entailment
- reductions to satisfiability,
- Deduction Thm, Contraposition Thm, Contradiction Thm,
- Equivalent Replacement Thm,
- negation normal form,
- TC0 and its use