Lecture 3

Equivalences and normal forms

Boolean algebras, equational reasoning, normal forms

Introduction to Logic for Computer Science

Prof Hongseok Yang
KAIST

These slides are minor variants of those made by Prof Worrell and Dr Haase for their logic course at Oxford.

Recap

Syntax of formulas of propositional logic:

1. *true* and *false* are formulas.
2. Every propositional variable x_i is a formula.
3. If F is a formula, then $\neg F$ is a formula.
4. If F and G are formulas, then $(F \wedge G)$ and $(F \vee G)$ are formulas.

Semantics of formulas via assignments, which are functions $\mathcal{A} \colon X \to \{0, 1\}$ that inductively extend to formulas.

Two formulas F and G are logically equivalent if $\mathcal{A}(F) = \mathcal{A}(G)$ for every assignment $\mathcal{A}$, written $F \equiv G$.

Agenda

1. **Boolean algebras**
2. **Equational reasoning**
3. **Normal forms**

Decision problems

A decision problem is a problem of finding an algorithm that outputs either “yes” or “no” for a given input (according to a specification).

- **Satisfiability**: Given a formula F, is F satisfiable?
- **Validity**: Given a formula F, is F valid?
- **Entailment**: Given formulas F and G, does $F \models G$ hold?
- **Equivalence**: Given formulas F and G, does $F \equiv G$ hold?

Decision problems

A decision problem is a problem of finding an algorithm that outputs either “yes” or “no” for a given input (according to a specification).

- **Satisfiability**: Given a formula F, is F satisfiable?
- **Validity**: Given a formula F, is F valid?
- **Entailment**: Given formulas F and G, does $F \models G$ hold?
- **Equivalence**: Given formulas F and G, does $F \equiv G$ hold?

Given a solver for one, we can build solvers for the others.

Decision problems

A decision problem is a problem of finding an algorithm that outputs either “yes” or “no” for a given input (according to a specification).

- **Satisfiability**: Given a formula F, is F satisfiable?
- **Validity**: Given a formula F, is F valid?
- **Entailment**: Given formulas F and G, does $F \models G$ hold?
- **Equivalence**: Given formulas F and G, does $F \equiv G$ hold?

Given a solver for one, we can build solvers for the others.

Ex1: Suppose that we are given a solver for equivalence:

$$\text{ck\_eq} : \mathcal{F}(X) \times \mathcal{F}(X) \to \{yes, no\}.$$

Use it to implement solvers for SAT, VALID, and entailment.

Ex2: Do you think that EQ is in NP?

Decision problems

A decision problem asks for finding an algorithm that outputs either "yes" or "no" for a given input (according to a specification).

- **Satisfiability**: Given a formula F, is F satisfiable?
- **Validity**: Given a formula F, is F valid?
- **Entailment**: Given formulas F and G, does $F \models G$ hold?
- **Equivalence**: Given formulas F and G, does $F \equiv G$ hold?

Given a solver for one, we can build solvers for the others.

Ex1: Suppose that we are given a solver for equivalence:

$$\text{ck\_eq} : \mathcal{F}(X) \times \mathcal{F}(X) \to \{yes, no\}.$$

Use it to implement solvers for SAT, VALID, and entailment.

Ex2: Do you think that EQ is in NP?

Equational reasoning

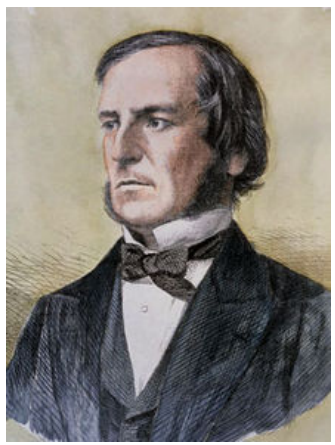
- Can show logical equivalence with brute force via truth tables. But exponential time always.

Equational reasoning

- Can show logical equivalence with brute force via truth tables. But exponential time always.
- **Equational reasoning** is more practical in many cases.
- Idea: Design rules for proving equivalences. Then, find a strategy for applying the rules.

Equational reasoning

- Can show logical equivalence with brute force via truth tables. But exponential time always.
- **Equational reasoning** is more practical in many cases.
- Idea: Design rules for proving equivalences. Then, find a strategy for applying the rules.
- Rule design step 1: Establish basic equivalences (e.g., axioms of Boolean algebra).
- Rule design step 2: Permit basic equivalences to be applied to subformulas.



Core axioms of Boolean algebra

$$F \wedge \mathit{true} \equiv F$$
$$F \vee \mathit{false} \equiv F$$

Identity Laws

$$F \wedge G \equiv G \wedge F$$
$$F \vee G \equiv G \vee F$$

Commutativity

$$(F \wedge G) \wedge H \equiv F \wedge (G \wedge H)$$
$$(F \vee G) \vee H \equiv F \vee (G \vee H)$$

Associativity

$$F \wedge (F \vee G) \equiv F$$
$$F \vee (F \wedge G) \equiv F$$

Absorption

$$F \wedge (G \vee H) \equiv (F \wedge G) \vee (F \wedge H)$$
$$F \vee (G \wedge H) \equiv (F \vee G) \wedge (F \vee H)$$

Distributivity

$$F \wedge \neg F \equiv \mathit{false}$$
$$F \vee \neg F \equiv \mathit{true}$$

Complementation

Core axioms of Boolean algebra – Every rule comes in pairs

$$F \wedge \mathit{true} \equiv F$$
$$F \vee \mathit{false} \equiv F$$

Identity Laws

$$F \wedge G \equiv G \wedge F$$
$$F \vee G \equiv G \vee F$$

Commutativity

$$(F \wedge G) \wedge H \equiv F \wedge (G \wedge H)$$
$$(F \vee G) \vee H \equiv F \vee (G \vee H)$$

Associativity

$$F \wedge (F \vee G) \equiv F$$
$$F \vee (F \wedge G) \equiv F$$

Absorption

$$F \wedge (G \vee H) \equiv (F \wedge G) \vee (F \wedge H)$$
$$F \vee (G \wedge H) \equiv (F \vee G) \wedge (F \vee H)$$

Distributivity

$$F \wedge \neg F \equiv \mathit{false}$$
$$F \vee \neg F \equiv \mathit{true}$$

Complementation

Core axioms of Boolean algebra – Two commutative monoids

$$F \wedge \mathit{true} \equiv F$$
$$F \vee \mathit{false} \equiv F$$

Identity Laws

$$F \wedge G \equiv G \wedge F$$
$$F \vee G \equiv G \vee F$$

Commutativity

$$(F \wedge G) \wedge H \equiv F \wedge (G \wedge H)$$
$$(F \vee G) \vee H \equiv F \vee (G \vee H)$$

Associativity

$$F \wedge (F \vee G) \equiv F$$
$$F \vee (F \wedge G) \equiv F$$

Absorption

$$F \wedge (G \vee H) \equiv (F \wedge G) \vee (F \wedge H)$$
$$F \vee (G \wedge H) \equiv (F \vee G) \wedge (F \vee H)$$

Distributivity

$$F \wedge \neg F \equiv \mathit{false}$$
$$F \vee \neg F \equiv \mathit{true}$$

Complementation

Core axioms of Boolean algebra – Bounded lattice

$$F \wedge \mathit{true} \equiv F$$
$$F \vee \mathit{false} \equiv F$$

Identity Laws

$$F \wedge G \equiv G \wedge F$$
$$F \vee G \equiv G \vee F$$

Commutativity

$$(F \wedge G) \wedge H \equiv F \wedge (G \wedge H)$$
$$(F \vee G) \vee H \equiv F \vee (G \vee H)$$

Associativity

$$F \wedge (F \vee G) \equiv F$$
$$F \vee (F \wedge G) \equiv F$$

Absorption

$$F \wedge (G \vee H) \equiv (F \wedge G) \vee (F \wedge H)$$
$$F \vee (G \wedge H) \equiv (F \vee G) \wedge (F \vee H)$$

Distributivity

$$F \wedge \neg F \equiv \mathit{false}$$
$$F \vee \neg F \equiv \mathit{true}$$

Complementation

Core axioms of Boolean algebra – Distributive lattice

$$F \wedge \mathit{true} \equiv F$$
$$F \vee \mathit{false} \equiv F$$

Identity Laws

$$F \wedge G \equiv G \wedge F$$
$$F \vee G \equiv G \vee F$$

Commutativity

$$(F \wedge G) \wedge H \equiv F \wedge (G \wedge H)$$
$$(F \vee G) \vee H \equiv F \vee (G \vee H)$$

Associativity

$$F \wedge (F \vee G) \equiv F$$
$$F \vee (F \wedge G) \equiv F$$

Absorption

$$F \wedge (G \vee H) \equiv (F \wedge G) \vee (F \wedge H)$$
$$F \vee (G \wedge H) \equiv (F \vee G) \wedge (F \vee H)$$

Distributivity

$$F \wedge \neg F \equiv \mathit{false}$$
$$F \vee \neg F \equiv \mathit{true}$$

Complementation

Core axioms of Boolean algebra – Complemented distributive lattice

$$F \wedge \mathit{true} \equiv F$$
$$F \vee \mathit{false} \equiv F$$

Identity Laws

$$F \wedge G \equiv G \wedge F$$
$$F \vee G \equiv G \vee F$$

Commutativity

$$(F \wedge G) \wedge H \equiv F \wedge (G \wedge H)$$
$$(F \vee G) \vee H \equiv F \vee (G \vee H)$$

Associativity

$$F \wedge (F \vee G) \equiv F$$
$$F \vee (F \wedge G) \equiv F$$

Absorption

$$F \wedge (G \vee H) \equiv (F \wedge G) \vee (F \wedge H)$$
$$F \vee (G \wedge H) \equiv (F \vee G) \wedge (F \vee H)$$

Distributivity

$$F \wedge \neg F \equiv \mathit{false}$$
$$F \vee \neg F \equiv \mathit{true}$$

Complementation

Derived axioms of Boolean algebra

$$F \wedge F \equiv F$$
$$F \vee F \equiv F$$ **Idempotence**

$$\neg\neg F \equiv F$$ **Double negation**

$$\neg(F \wedge G) \equiv (\neg F \vee \neg G)$$
$$\neg(F \vee G) \equiv (\neg F \wedge \neg G)$$ **De Morgan's laws**

$$F \wedge \mathit{false} \equiv \mathit{false}$$
$$F \vee \mathit{true} \equiv \mathit{true}$$ **Zero Laws**

Derived axioms of Boolean algebra

$F \wedge F \equiv F$	
$F \vee F \equiv F$	**Idempotence**
$\neg\neg F \equiv F$	**Double negation**
$\neg(F \wedge G) \equiv (\neg F \vee \neg G)$	
$\neg(F \vee G) \equiv (\neg F \wedge \neg G)$	**De Morgan's laws**
$F \wedge \mathit{false} \equiv \mathit{false}$	
$F \vee \mathit{true} \equiv \mathit{true}$	**Zero Laws**

Ex: Derive these axioms from the core axioms in the previous slides.

Boolean algebras

A **Boolean algebra** is a set A together with two elements

$$\mathit{true}, \mathit{false} \in A,$$

one unary operation

$$\neg \colon A \to A,$$

and two binary operations

$$\wedge, \vee \colon A \times A \to A$$

satisfying the core Boolean algebra axioms (i.e., the six axioms for complemented distributive lattices).

Boolean algebras

A **Boolean algebra** is a set A together with two elements

$$\mathit{true}, \mathit{false} \in A,$$

one unary operation

$$\neg \colon A \to A,$$

and two binary operations

$$\wedge, \vee \colon A \times A \to A$$

satisfying the core Boolean algebra axioms (i.e., the six axioms for complemented distributive lattices).

- Example 1: $A = \{0, 1\}$.

Boolean algebras

A **Boolean algebra** is a set A together with two elements

$$\mathit{true}, \mathit{false} \in A,$$

one unary operation

$$\neg\colon A \to A,$$

and two binary operations

$$\wedge, \vee\colon A \times A \to A$$

satisfying the core Boolean algebra axioms (i.e., the six axioms for complemented distributive lattices).

- Example 1: $A = \{0, 1\}$.
- Example 2: For any set X, take $A = 2^X$ with $\mathit{true} = X$, $\mathit{false} = \emptyset$, $\wedge = \cap$, $\vee = \cup$, $\neg S = X \setminus S$.

Boolean algebras

A **Boolean algebra** is a set A together with two elements

$$\mathit{true}, \mathit{false} \in A,$$

one unary operation

$$\neg\colon A \to A,$$

and two binary operations

$$\wedge, \vee\colon A \times A \to A$$

satisfying the core Boolean algebra axioms (i.e., the six axioms for complemented distributive lattices).

- Example 1: $A = \{0, 1\}$.
- Example 2: For any set X, take $A = 2^X$ with $\mathit{true} = X$, $\mathit{false} = \emptyset$, $\wedge = \cap$, $\vee = \cup$, $\neg S = X \setminus S$.
- Ex: In fact, any finite Boolean algebra is of the form 2^X. What about infinite Boolean algebras?

Boolean algebras and Boolean rings

- A **Boolean ring** is a ring A with 1 in which every element satisfies $a^2 = a$.

Boolean algebras and Boolean rings

- A **Boolean ring** is a ring A with 1 in which every element satisfies $a^2 = a$.
- Any Boolean ring A gives a Boolean algebra by

$$\begin{aligned} \mathit{true} &:= 1, & \mathit{false} &:= 0, \\ a \wedge b &:= ab, & a \vee b &:= a + b + ab, & \neg a &:= 1 + a. \end{aligned}$$

- Ex1: Prove this.

Boolean algebras and Boolean rings

- A **Boolean ring** is a ring A with 1 in which every element satisfies $a^2 = a$.
- Any Boolean ring A gives a Boolean algebra by

$$\begin{aligned} \mathit{true} &:= 1, & \mathit{false} &:= 0, \\ a \wedge b &:= ab, & a \vee b &:= a + b + ab, & \neg a &:= 1 + a. \end{aligned}$$

- Ex1: Prove this.
- Any Boolean algebra A gives a Boolean ring by

$$\begin{aligned} ab &:= a \wedge b, \\ a + b &:= (a \wedge \neg b) \vee (\neg a \wedge b). \end{aligned}$$

- Ex2: Prove this.

Boolean algebras and Boolean rings

- A **Boolean ring** is a ring A with 1 in which every element satisfies $a^2 = a$.
- Any Boolean ring A gives a Boolean algebra by

$$\begin{aligned} \mathit{true} &:= 1, & \mathit{false} &:= 0, \\ a \wedge b &:= ab, & a \vee b &:= a + b + ab, & \neg a &:= 1 + a. \end{aligned}$$

- Ex1: Prove this.
- Any Boolean algebra A gives a Boolean ring by

$$\begin{aligned} ab &:= a \wedge b, \\ a + b &:= (a \wedge \neg b) \vee (\neg a \wedge b). \end{aligned}$$

- Ex2: Prove this.
- So **Boolean algebras = Boolean rings**.

Substitution

The essence of equational reasoning is the **substitution** of equals for equals.

Informally, $G[F/H]$ means *"substitute F for H in G"*. E.g.:

$$(p_1 \wedge (p_2 \vee p_1))[\neg q_1/p_1] = \neg q_1 \wedge (p_2 \vee \neg q_1)$$

Substitution

The essence of equational reasoning is the **substitution** of equals for equals.

Informally, $G[F/H]$ means *"substitute F for H in G"*. E.g.:

$$(p_1 \wedge (p_2 \vee p_1))[\neg q_1/p_1] = \neg q_1 \wedge (p_2 \vee \neg q_1)$$

Formally, $G[F/H] := F$ if $G = H$. Whenever $G \neq H$, we proceed by induction:

- Base cases:

$$\begin{aligned} x[F/H] &:= x && \text{for all } x \in X \\ \mathit{true}[F/H] &:= \mathit{true} \\ \mathit{false}[F/H] &:= \mathit{false} \end{aligned}$$

- Induction steps:

$$\begin{aligned} (\neg G)[F/H] &:= \neg(G[F/H]) \\ (G_1 \wedge G_2)[F/H] &:= G_1[F/H] \wedge G_2[F/H] \\ (G_1 \vee G_2)[F/H] &:= G_1[F/H] \vee G_2[F/H]. \end{aligned}$$

The substitution theorem

Theorem (Substitution Theorem)

Let F, G, G', H be formulas such that $G' = G[F/H]$ and $F \equiv H$. Then $G' \equiv G$.

Equational reasoning

The equivalence

$$(P \vee (Q \vee R)) \wedge (R \vee \neg P) \equiv R \vee (\neg P \wedge Q)\,.$$

has the following equational proof:

$$\begin{aligned}
(P \vee (Q \vee R)) \wedge (R \vee \neg P) &\equiv ((P \vee Q) \vee R) \wedge (R \vee \neg P) \\
&\equiv (R \vee (P \vee Q)) \wedge (R \vee \neg P) \\
&\equiv R \vee ((P \vee Q) \wedge \neg P) \\
&\equiv R \vee (\neg P \wedge (P \vee Q)) \\
&\equiv R \vee ((\neg P \wedge P) \vee (\neg P \wedge Q)) \\
&\equiv R \vee (\mathit{false} \vee (\neg P \wedge Q)) \\
&\equiv R \vee (\neg P \wedge Q).
\end{aligned}$$

Equational reasoning

The equivalence

$$(P \vee (Q \vee R)) \wedge (R \vee \neg P) \equiv R \vee (\neg P \wedge Q)\,.$$

has the following equational proof:

$$\begin{aligned}
(P \vee (Q \vee R)) \wedge (R \vee \neg P) &\equiv ((P \vee Q) \vee R) \wedge (R \vee \neg P) \\
&\equiv (R \vee (P \vee Q)) \wedge (R \vee \neg P) \\
&\equiv R \vee ((P \vee Q) \wedge \neg P) \\
&\equiv R \vee (\neg P \wedge (P \vee Q)) \\
&\equiv R \vee ((\neg P \wedge P) \vee (\neg P \wedge Q)) \\
&\equiv R \vee (\mathit{false} \vee (\neg P \wedge Q)) \\
&\equiv R \vee (\neg P \wedge Q).
\end{aligned}$$

Equational reasoning

The equivalence

$$(P \vee (Q \vee R)) \wedge (R \vee \neg P) \equiv R \vee (\neg P \wedge Q)\,.$$

has the following equational proof:

$$\begin{aligned}
(P \vee (Q \vee R)) \wedge (R \vee \neg P) &\equiv ((P \vee Q) \vee R) \wedge (R \vee \neg P) \\
&\equiv (R \vee (P \vee Q)) \wedge (R \vee \neg P) \\
&\equiv R \vee ((P \vee Q) \wedge \neg P) \\
&\equiv R \vee (\neg P \wedge (P \vee Q)) \\
&\equiv R \vee ((\neg P \wedge P) \vee (\neg P \wedge Q)) \\
&\equiv R \vee (\mathit{false} \vee (\neg P \wedge Q)) \\
&\equiv R \vee (\neg P \wedge Q).
\end{aligned}$$

Equational reasoning

The equivalence

$$(P \vee (Q \vee R)) \wedge (R \vee \neg P) \equiv R \vee (\neg P \wedge Q)\,.$$

has the following equational proof:

$$\begin{aligned}
(P \vee (Q \vee R)) \wedge (R \vee \neg P) &\equiv ((P \vee Q) \vee R) \wedge (R \vee \neg P) \\
&\equiv (R \vee (P \vee Q)) \wedge (R \vee \neg P) \\
&\equiv R \vee ((P \vee Q) \wedge \neg P) \\
&\equiv R \vee (\neg P \wedge (P \vee Q)) \\
&\equiv R \vee ((\neg P \wedge P) \vee (\neg P \wedge Q)) \\
&\equiv R \vee (\mathit{false} \vee (\neg P \wedge Q)) \\
&\equiv R \vee (\neg P \wedge Q).
\end{aligned}$$

Equational reasoning

The equivalence

$$(P \vee (Q \vee R)) \wedge (R \vee \neg P) \equiv R \vee (\neg P \wedge Q)\,.$$

has the following equational proof:

$$\begin{aligned}
(P \vee (Q \vee R)) \wedge (R \vee \neg P) &\equiv ((P \vee Q) \vee R) \wedge (R \vee \neg P) \\
&\equiv (R \vee (P \vee Q)) \wedge (R \vee \neg P) \\
&\equiv R \vee ((P \vee Q) \wedge \neg P) \\
&\equiv R \vee (\neg P \wedge (P \vee Q)) \\
&\equiv R \vee ((\neg P \wedge P) \vee (\neg P \wedge Q)) \\
&\equiv R \vee (\mathit{false} \vee (\neg P \wedge Q)) \\
&\equiv R \vee (\neg P \wedge Q).
\end{aligned}$$

Equational reasoning

The equivalence

$$(P \vee (Q \vee R)) \wedge (R \vee \neg P) \equiv R \vee (\neg P \wedge Q)\,.$$

has the following equational proof:

$$\begin{aligned}
(P \vee (Q \vee R)) \wedge (R \vee \neg P) &\equiv ((P \vee Q) \vee R) \wedge (R \vee \neg P) \\
&\equiv (R \vee (P \vee Q)) \wedge (R \vee \neg P) \\
&\equiv R \vee ((P \vee Q) \wedge \neg P) \\
&\equiv R \vee (\neg P \wedge (P \vee Q)) \\
&\equiv R \vee ((\neg P \wedge P) \vee (\neg P \wedge Q)) \\
&\equiv R \vee (\mathit{false} \vee (\neg P \wedge Q)) \\
&\equiv R \vee (\neg P \wedge Q).
\end{aligned}$$

Equational reasoning

The equivalence

$$(P \vee (Q \vee R)) \wedge (R \vee \neg P) \equiv R \vee (\neg P \wedge Q)\,.$$

has the following equational proof:

$$\begin{aligned}
(P \vee (Q \vee R)) \wedge (R \vee \neg P) &\equiv ((P \vee Q) \vee R) \wedge (R \vee \neg P) \\
&\equiv (R \vee (P \vee Q)) \wedge (R \vee \neg P) \\
&\equiv R \vee ((P \vee Q) \wedge \neg P) \\
&\equiv R \vee (\neg P \wedge (P \vee Q)) \\
&\equiv R \vee ((\neg P \wedge P) \vee (\neg P \wedge Q)) \\
&\equiv R \vee (\mathit{false} \vee (\neg P \wedge Q)) \\
&\equiv R \vee (\neg P \wedge Q).
\end{aligned}$$

Equational reasoning

The equivalence

$$(P \vee (Q \vee R)) \wedge (R \vee \neg P) \equiv R \vee (\neg P \wedge Q)\,.$$

has the following equational proof:

$$\begin{aligned}
(P \vee (Q \vee R)) \wedge (R \vee \neg P) &\equiv ((P \vee Q) \vee R) \wedge (R \vee \neg P) \\
&\equiv (R \vee (P \vee Q)) \wedge (R \vee \neg P) \\
&\equiv R \vee ((P \vee Q) \wedge \neg P) \\
&\equiv R \vee (\neg P \wedge (P \vee Q)) \\
&\equiv R \vee ((\neg P \wedge P) \vee (\neg P \wedge Q)) \\
&\equiv R \vee (\mathit{false} \vee (\neg P \wedge Q)) \\
&\equiv R \vee (\neg P \wedge Q).
\end{aligned}$$

Equational reasoning

The equivalence

$$(P \vee (Q \vee R)) \wedge (R \vee \neg P) \equiv R \vee (\neg P \wedge Q)\,.$$

has the following equational proof:

$$\begin{aligned}
(P \vee (Q \vee R)) \wedge (R \vee \neg P) &\equiv ((P \vee Q) \vee R) \wedge (R \vee \neg P) \\
&\equiv (R \vee (P \vee Q)) \wedge (R \vee \neg P) \\
&\equiv R \vee ((P \vee Q) \wedge \neg P) \\
&\equiv R \vee (\neg P \wedge (P \vee Q)) \\
&\equiv R \vee ((\neg P \wedge P) \vee (\neg P \wedge Q)) \\
&\equiv R \vee (\mathit{false} \vee (\neg P \wedge Q)) \\
&\equiv R \vee (\neg P \wedge Q).
\end{aligned}$$

Equational reasoning

The equivalence

$$(P \vee (Q \vee R)) \wedge (R \vee \neg P) \equiv R \vee (\neg P \wedge Q)\,.$$

has the following equational proof:

$$\begin{aligned}
(P \vee (Q \vee R)) \wedge (R \vee \neg P) &\equiv ((P \vee Q) \vee R) \wedge (R \vee \neg P) \\
&\equiv (R \vee (P \vee Q)) \wedge (R \vee \neg P) \\
&\equiv R \vee ((P \vee Q) \wedge \neg P) \\
&\equiv R \vee (\neg P \wedge (P \vee Q)) \\
&\equiv R \vee ((\neg P \wedge P) \vee (\neg P \wedge Q)) \\
&\equiv R \vee (\mathit{false} \vee (\neg P \wedge Q)) \\
&\equiv R \vee (\neg P \wedge Q).
\end{aligned}$$

Equational reasoning

The equivalence

$$(P \vee (Q \vee R)) \wedge (R \vee \neg P) \equiv R \vee (\neg P \wedge Q)\,.$$

has the following equational proof:

$$\begin{aligned}
(P \vee (Q \vee R)) \wedge (R \vee \neg P) &\equiv ((P \vee Q) \vee R) \wedge (R \vee \neg P) \\
&\equiv (R \vee (P \vee Q)) \wedge (R \vee \neg P) \\
&\equiv R \vee ((P \vee Q) \wedge \neg P) \\
&\equiv R \vee (\neg P \wedge (P \vee Q)) \\
&\equiv R \vee ((\neg P \wedge P) \vee (\neg P \wedge Q)) \\
&\equiv R \vee (\mathit{false} \vee (\neg P \wedge Q)) \\
&\equiv R \vee (\neg P \wedge Q).
\end{aligned}$$

Normal forms

- A **literal** is a propositional variable or its negation: x or $\neg x$.

Normal forms

- A **literal** is a propositional variable or its negation: x or $\neg x$.
- A formula F is in **conjunctive normal form** (**CNF**) if it is a conjunction of disjunctions of literals $L_{i,j}$:

$$F = \bigwedge_{i=1}^{n} \left(\bigvee_{j=1}^{m_i} L_{i,j} \right).$$

Normal forms

- A **literal** is a propositional variable or its negation: x or $\neg x$.
- A formula F is in **conjunctive normal form** (**CNF**) if it is a conjunction of disjunctions of literals $L_{i,j}$:

$$F = \bigwedge_{i=1}^{n} \left(\bigvee_{j=1}^{m_i} L_{i,j} \right).$$

- A formula F is in **disjunctive normal form** (**DNF**) if it is a disjunction of conjunctions of literals $L_{i,j}$:

$$F = \bigvee_{i=1}^{n} \left(\bigwedge_{j=1}^{m_i} L_{i,j} \right).$$

- Convention: 1) $\bigvee L_{i,j}$ in CNF and $\bigwedge L_{i,j}$ in DNF are called **clauses**. 2) *true* is CNF with no clauses. 3) *false* is CNF with a single clause without literals.

Normal forms

- A **literal** is a propositional variable or its negation: x or $\neg x$.
- A **clause** is a disjunction of literals:

$$C = \bigvee_{j=1}^{m} L_j.$$

- A formula F is in **conjunctive normal form** (**CNF**) if it is a conjunction of disjunctions of literals $L_{i,j}$ (i.e., a conj. of clauses):

$$F = \bigwedge_{i=1}^{n} \left(\bigvee_{j=1}^{m_i} L_{i,j} \right).$$

- A formula F is in **disjunctive normal form** (**DNF**) if it is a disjunction of conjunctions of literals $L_{i,j}$:

$$F = \bigvee_{i=1}^{n} \left(\bigwedge_{j=1}^{m_i} L_{i,j} \right).$$

- Ex: Which of SAT and VALID can be decided in polynomial time for DNF formulas? What about CNF formulas?

Theorem (Normalisation Theorem)

For every formula, there are an equivalent formula in CNF and an equivalent formula in DNF.

Theorem (Normalisation Theorem)

For every formula, there are an equivalent formula in CNF and an equivalent formula in DNF.

Proof.

Trivial. □

“Methods of proof”

- **Proof by example**: only give the case $n = 2$.
- **Proof by intimidation**: “trivial”.
- **Proof by vigorous handwaving**: works well in classroom.
- **Proof by cumbersome notation**: use four alphabets.
- **Proof by exhaustion**: devote journal issue to your proof.
- **Proof by omission**: “the other 253 cases are analogous”.
- **Proof by obfuscation**: long plotless sequence of statements.
- **Proof by importance**: many useful consequences follow.
- **Proof by evidence**: long search gave no counterexample.
- **Proof by metaproof**: give method to construct desired proof.
- **Proof by semantic shift**: change standard definitions.
- **Proof by reference to inaccessible literature**: cite a simple corollary of a theorem to be found in a privately circulated memoir of the Slovenian Philological Society, 1883.

Proof by truth table

x	y	z	F
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

- Each row with value 1 gives a clause in the DNF formula.
- For each propositional variable x, the clause contains the literal x if 1 appears in column x, and $\neg x$ otherwise.

Proof by truth table

x	y	z	F
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

- Each row with value 1 gives a clause in the DNF formula.
- For each propositional variable x, the clause contains the literal x if 1 appears in column x, and $\neg x$ otherwise.

Proof by truth table

x	y	z	F
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

- Each row with value 1 gives a clause in the DNF formula.
- For each propositional variable x, the clause contains the literal x if 1 appears in column x, and $\neg x$ otherwise.

- Each row with value 0 gives a clause in the CNF formula.
- For each propositional variable x, the clause contains the literal x if 0 appears in column x, and $\neg x$ otherwise.

Proof by truth table

x	y	z	F
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

- Each row with value 1 gives a clause in the DNF formula.
- For each propositional variable x, the clause contains the literal x if 1 appears in column x, and $\neg x$ otherwise.

- Each row with value 0 gives a clause in the CNF formula.
- For each propositional variable x, the clause contains the literal x if 0 appears in column x, and $\neg x$ otherwise.

Equational transformation to CNF

Is this efficient?

Equational transformation to CNF

Is this efficient? No! Better way:

1. Use **double negation** and **De Morgan's laws** to substitute in F every occurrence of a subformula of the form

$$\begin{aligned} \neg\neg G \quad & \text{by} \quad G, \\ \neg(G \wedge H) \quad & \text{by} \quad (\neg G \vee \neg H), \\ \neg(G \vee H) \quad & \text{by} \quad (\neg G \wedge \neg H), \\ \neg \mathit{true} \quad & \text{by} \quad \mathit{false}, \\ \neg \mathit{false} \quad & \text{by} \quad \mathit{true}, \end{aligned}$$

   until no such formulas occur (i.e., push all negations inward until negation is only applied to propositional variables), yielding the **negation normal form**.

Equational transformation to CNF

2. Use **distributivity** to substitute in F every occurrence of a subformula of the form

$$\begin{aligned} G \vee (H \wedge R) \quad & \text{by} \quad (G \vee H) \wedge (G \vee R), \\ (H \wedge R) \vee G \quad & \text{by} \quad (H \vee G) \wedge (R \vee G), \\ G \vee \mathit{true} \quad & \text{by} \quad \mathit{true}, \\ \mathit{true} \vee G \quad & \text{by} \quad \mathit{true}, \end{aligned}$$

until no such formulas occur (i.e., push all disjunctions inward until no conjunction occurs under a disjunction).

Equational transformation to CNF

2. Use **distributivity** to substitute in F every occurrence of a subformula of the form

$$\begin{aligned} G \vee (H \wedge R) \quad & \text{by} \quad (G \vee H) \wedge (G \vee R), \\ (H \wedge R) \vee G \quad & \text{by} \quad (H \vee G) \wedge (R \vee G), \\ G \vee \mathit{true} \quad & \text{by} \quad \mathit{true}, \\ \mathit{true} \vee G \quad & \text{by} \quad \mathit{true}, \end{aligned}$$

   until no such formulas occur (i.e., push all disjunctions inward until no conjunction occurs under a disjunction).

3. Use the **identity** and **zero laws** to remove *false* from any clause and to delete all clauses containing *true*.

Summary

- Equational reasoning is often more practical than using truth tables.
- It is based on axioms and derived rules of Boolean algebra, and also on substitution.
- Equational reasoning allows reduction to normal forms.
- CNF and DNF formulas are equally expressive as the class of all formulas.

Summary

- Equational reasoning is often more practical than using truth tables.
- It is based on axioms and derived rules of Boolean algebra, and also on substitution.
- Equational reasoning allows reduction to normal forms.
- CNF and DNF formulas are equally expressive as the class of all formulas.

Note:

- CNF can be exponentially shorter than DNF.
- SAT is trivial for DNF formulas.
- Later: SAT for CNF formulas = SAT for any formula.