# **EDAA40 Discrete Structures in Computer Science**

8: Propositional logic

# mechanizing belief management

A major goal of any logic is to mechanize reasoning: we figure out truth through manipulating strings of symbols according to some rules.

Ultimately, this can be done by a machine.



Gottfried Wilhelm von Leibniz 1646-1716

#### We need:

- 1. a way to represent the truth/falsehood of propositions
- 2. a system of symbols for constructing, combining, relating propositions
- 3. rules for manipulating and reasoning about them

We assume that all propositions are either true or false (bivalence). Truth and falsehood are represented by 1 and 0, respectively.

# basic logic connectives

$\alpha$	$\neg \alpha$
1	0
0	1

$\alpha$	$\beta$	$\alpha \wedge \beta$	$\alpha \vee \beta$	$\alpha \to \beta$	$\alpha \leftrightarrow \beta$	$\alpha \veebar \beta$	$\alpha \overline{\wedge} \beta$
1	1	1	1	1	1	0	0
1	0	O	1	0	0	1	1
0	1	O	1	1	0	1	1
0	0	0	O	1	1	0	1
		and	or	implies	iff	xor	nand



The "and" connective is also called *conjunction*, the "or" connective *disjunction*.

#### from truth table to formula

р	q	r	f(p, q, r)		
1	1	1	1	$p \wedge q \wedge r$	Can this expression be
1	1	0	0		simplified?
1	0	1	1		Are there truth tables fo
1	0	0	0		which this method does
0	1	1	0		not produce a formula?
0	1	0	0		What does this mean for connectives not, and, or?
0	0	1	1	$\neg p \land \neg q \land r$	connectives not, and, of p
0	0	0	0		

 $f:(p,q,r)\mapsto (p\wedge q\wedge r)\vee (p\wedge \neg q\wedge r)\vee (\neg p\wedge \neg q\wedge r)$ 

# relating connectives

$\alpha$	$\beta$	$\alpha \wedge \beta$	$\alpha \vee \beta$	$\alpha \to \beta$	$\alpha \leftrightarrow \beta$	$\alpha \vee \beta$	$\alpha \overline{\wedge} \beta$
1	1	1	1	1	1	O	0
1	0	O	1	0	0	1	1
0	1	0	1	1	0	1	1
0	0	0	0	1	1	0	1
		and	or	implies	iff	xor	nand



Express or in terms of not and and.  $\alpha \vee \beta = \dots$ 

Is there one connective to express them all?

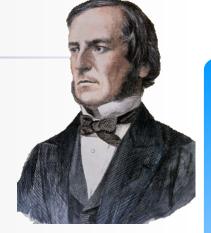
$$\neg \alpha = \alpha \,\overline{\wedge}\, \alpha$$
$$\alpha \wedge \beta = \neg(\alpha \,\overline{\wedge}\, \beta) = (\alpha \,\overline{\wedge}\, \beta) \,\overline{\wedge}\, (\alpha \,\overline{\wedge}\, \beta)$$

# **Boolean algebra(s)**

Propositional logic is a Boolean algebra.

A Boolean algebra is a set B, with one unary and two binary operations that have the following properties, for all  $x, y, z \in B$ :

there is a 
$$z \in B : x \lor z = x$$
 and  $x \land \neg x = z$   
there is a  $e \in B : x \land e = x$  and  $x \lor \neg x = e$   
 $x \lor y = y \lor x$  and  $x \land y = y \land x$   
 $x \land (y \lor z) = (x \land y) \lor (x \land z)$   
 $x \lor (y \land z) = (x \lor y) \land (x \lor z)$ 



George Boole 1815-1864

Propositional logic is a Boolean algebra with  $B=\{1, 0\}, z=0, e=1$ , and the operations and ( $\land$ ), or ( $\lor$ ), not ( $\neg$ ).



Can you think of other Boolean algebras?

(n)

**---**-

0

5

3)

## language layers in logic

In logic, we are concerned with several layers of language, and the truth of statements in them.

1. Elementary propositions.

These are treated as "atomic", all we care about is that they are either true or false.

In logic, we represent them with elementary letters.

- "All birds can fly."
- "17 is a prime number."
- "4 > 11"

 $p, q, r, \dots$ 

2. Formulae.

Expressions constructed recursively from elementary letters and *connectives*.

We represent them with greek symbols.

 $(p \land q) \to (s \lor t)$  $\neg p \lor q$  $\alpha, \beta, \gamma, \dots$ 

3. Relations between formulae.

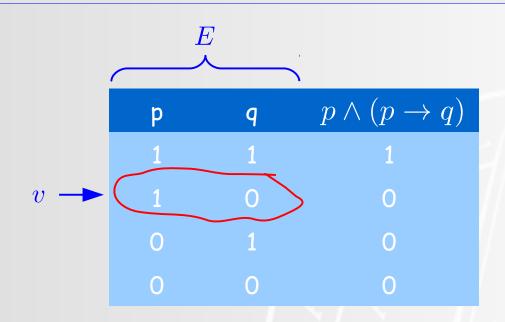
This is what we use to reason about formulae.

 $\alpha_1, ..., \alpha_n \vdash \beta$   $\alpha \dashv \vdash \beta$ 

### assignments & valuations

Given a set of elementary letters E, an assignment is a function

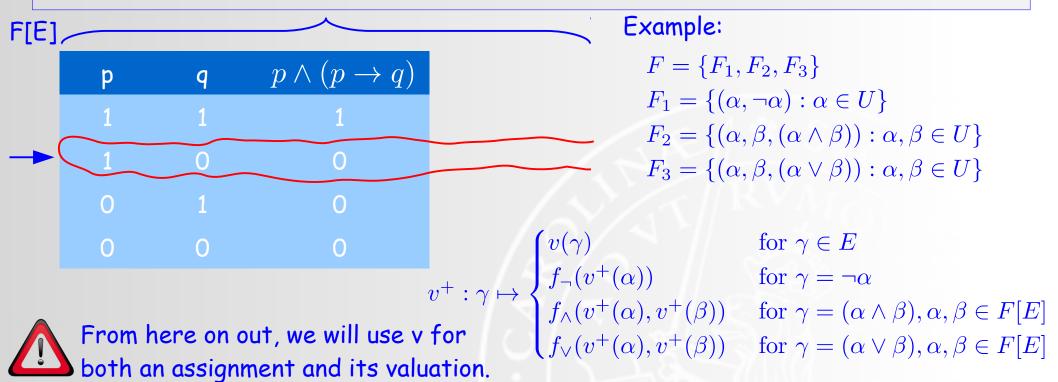
$$v: E \longrightarrow \{1, 0\}$$



# assignments & valuations

Given a set of elementary letters E and an assignment v, a *valuation* is the recursive extension of v over the set F[E] of formulae in E generated by a set of rules F:

$$v^+: F[E] \longrightarrow \{1,0\}$$



## tautological implication

Given a set of formulae A and a formula  $\beta$  we say that A tautologically implies  $\beta$  if there is no valuation v such that

$$A \models \beta$$

 $A \vdash \beta$ 

$$v(\alpha) = 1$$
 for all  $\alpha \in A$  and  $v(\beta) = 0$ 

#### Example:

(modus ponens)

$$p,p\to q\vdash q$$

р	q	$p \rightarrow q$
1	1	1
1	0	0
0	1	1
0	0	1

Name	LHS	RHS
Simplification, ∧ <sup>−</sup>	αΛβ	α
	αΛβ	β
Conjunction, ∧ <sup>+</sup>	α, β	αΛβ
Disjunction, ∨ <sup>+</sup>	α	ανβ
	β	$\alpha \vee \beta$
Modus ponens, MP, $\rightarrow$	$\alpha, \alpha \rightarrow \beta$	β
Modus tollens, MT	$\neg \beta, \alpha \rightarrow \beta$	$\neg \alpha$
Disjunctive syllogism, DS	$\alpha \vee \beta$ , $\neg \alpha$	β
Transitivity	$\alpha \rightarrow \beta, \beta \rightarrow \gamma$	$\alpha \rightarrow \gamma$
Material implication	β	$\alpha \rightarrow \beta$
	$\neg \alpha$	$\alpha \rightarrow \beta$
Limiting cases	γ	β∨¬β
	$\alpha \wedge \neg \alpha$	γ

**SLAM**, p. 196

### tautological equivalence

Given two formulae  $\alpha$  and  $\beta$ , we say that they are tautologically equivalent if they tautologically imply each other.

$$\alpha + \beta$$

Name	LHS	RHS	
Double negation	α	$\neg \neg \alpha$	
Commutation for ∧	αΛβ	βΛα	
Association for ∧	$\alpha \wedge (\beta \wedge \gamma)$	$(\alpha \land \beta) \land \gamma$	
Commutation for ∨	α∨β	β∨α	
Association for ∨	$\alpha \vee (\beta \vee \gamma)$	$(\alpha \lor \beta) \lor \gamma$	
Distribution of ∧ over ∨	$\alpha \wedge (\beta \vee \gamma)$	$(\alpha \land \beta) \lor (\alpha \land \gamma)$	
Distribution of ∨ over ∧	$\alpha \vee (\beta \wedge \gamma)$	$(\alpha \vee \beta) \wedge (\alpha \vee \gamma)$	
Absorption	α	$\alpha \wedge (\alpha \vee \beta)$	
	α	$\alpha \vee (\alpha \wedge \beta)$	
Expansion	α	$(\alpha \land \beta) \lor (\alpha \land \neg \beta)$	
	α	$(\alpha \lor \beta) \land (\alpha \lor \neg \beta)$	
de Morgan	$\neg(\alpha \land \beta)$	$\neg \alpha \lor \neg \beta$	
	$\neg(\alpha \lor \beta)$	$\neg \alpha \land \neg \beta$	
	αΛβ	$\neg(\neg\alpha\lor\neg\beta)$	
	α∨β	$\neg(\neg\alpha\wedge\neg\beta)$	
Limiting cases	α∧¬α	β∧¬β	
	$\alpha \lor \neg \alpha$	$\beta \vee \neg \beta$	

# tautological equivalence

Name	LHS	RHS
Contraposition	$\alpha \rightarrow \beta$	$\neg \beta \rightarrow \neg \alpha$
	$\alpha \rightarrow \neg \beta$	$\beta \rightarrow \neg \alpha$
	$\neg \alpha \rightarrow \beta$	$\neg \beta \rightarrow \alpha$
Import/export	$\alpha \rightarrow (\beta \rightarrow \gamma)$	$(\alpha \land \beta) \rightarrow \gamma$
	$\alpha \rightarrow (\beta \rightarrow \gamma)$	$\beta \rightarrow (\alpha \rightarrow \gamma)$
Consequential mirabilis (miraculous consequence)	$\alpha \rightarrow \neg \alpha$	$\neg \alpha$
	$\neg \alpha \rightarrow \alpha$	α
Commutation for ↔	$\alpha \leftrightarrow \beta$	β↔α
Association for ↔	$\alpha \leftrightarrow (\beta \leftrightarrow \gamma)$	$(\alpha \leftrightarrow \beta) \leftrightarrow \gamma$
¬ through ↔	$\neg(\alpha \leftrightarrow \beta)$	$\alpha \leftrightarrow \neg \beta$
Translations between connectives	α↔β	$(\alpha \rightarrow \beta) \land (\beta \rightarrow \alpha)$
	$\alpha \leftrightarrow \beta$	$(\alpha \land \beta) \lor (\neg \alpha \land \neg \beta)$
	$\alpha \rightarrow \beta$	$\neg(\alpha \land \neg \beta)$
	$\alpha \rightarrow \beta$	$\neg \alpha \lor \beta$
	α∨β	$\neg \alpha \rightarrow \beta$
Translations of negations of connectives	$\neg(\alpha \rightarrow \beta)$	α∧¬β
	$\neg(\alpha \land \beta)$	$\alpha \rightarrow \neg \beta$
	$\neg(\alpha \leftrightarrow \beta)$	$(\alpha \land \neg \beta) \lor (\beta \land \neg \alpha)$

SLAM, p. 199

## tautology and contradiction

```
A formula \alpha ... is a tautology iff v(\alpha)=1 for every valuation v ... is a contradiction iff v(\alpha)=0 for every valuation v ... contingent otherwise.
```

### **DNF: disjunctive normal form**

Anormal form is a transformation of a formula into another equivalent form that has particular properties. Some definitions:

- A literal is an elementary letter or its negation.
- A basic conjunction is a conjunction of literals, without repetition.

**Examples:** 
$$p \wedge q \wedge \neg r$$
  $p \qquad \neg p \wedge \neg q$ 

A formula is in disjunctive normal form if it is a disjunction of basic conjunctions.

**Example:** 
$$(p \land q \land \neg r) \lor (\neg p \land \neg q)$$

A formula is in *full disjunctive normal form* every letter occurs in every basic conjunction.

**Example:** 
$$(p \land q \land \neg r) \lor (\neg p \land \neg q \land r) \lor (\neg p \land \neg q \land \neg r)$$

### constructing the DNF

Given a formula, there are two common ways of constructing its DNF:

- 1. via its truth table
- 2. through successive syntactic transformations

#### Basic algorithm for building a DNF (SLAM p. 204):

- 1. express all connectives through negation, conjunction, disjunction
- 2. move negations inward (de Morgan), eliminate double negation
- 3. move conjunctions inward (distribution)
- 4. remove repetitions in basic conjunctions
- 5. remove basic conjunctions with a letter and its negation

Example: 
$$\neg((p \lor \neg q) \to (p \land \neg q))$$

1. 
$$\neg(\neg(p \lor \neg q) \lor (p \land \neg q))$$
 2.3  $(p \lor \neg q) \land (\neg p \lor q)$ 

$$\textbf{2.1} \ \neg \neg (p \lor \neg q) \land \neg (p \land \neg q) \quad \textbf{3.} \quad (p \land \neg p) \lor (p \land q) \lor (\neg q \land \neg p) \lor (\neg q \land q)$$

**2.2** 
$$\neg\neg(p \lor \neg q) \land (\neg p \lor q)$$
 **5.**  $(p \land q) \lor (\neg p \land \neg q)$