



Logic

Luca Campa

Philipp Dablander

Aaron Groß

Aart Middeldorp

Alexander Montag

Johannes Niederhauser

Vera Schmitt

articify
ars.uibk.ac.at

with session ID **6893 6178** for anonymous questions



Outline

- 1. Summary of Previous Lecture**
- 2. Completeness**
- 3. Resolution**
- 4. Intermezzo**
- 5. Binary Decision Diagrams**
- 6. Further Reading**

Definitions

▶ sequent

$$\underbrace{\varphi_1, \varphi_2, \dots, \varphi_n}_{\text{premises}} \vdash \underbrace{\psi}_{\text{conclusion}}$$

with propositional formulas $\varphi_1, \varphi_2, \dots, \varphi_n, \psi$

- ▶ sequent $\varphi_1, \varphi_2, \dots, \varphi_n \vdash \psi$ is **valid** if ψ can be proved from premises $\varphi_1, \varphi_2, \dots, \varphi_n$ using **proof rules** of natural deduction
- ▶ **theorem** is formula φ such that sequent $\vdash \varphi$ is valid

Theorem

- ▶ natural deduction is **sound**: $\varphi_1, \varphi_2, \dots, \varphi_n \vdash \psi$ is valid $\implies \varphi_1, \varphi_2, \dots, \varphi_n \vDash \psi$
- ▶ proof rules MT, $\neg\neg$ i, PBC and LEM are **derivable** from basic proof rules
- ▶ proof rules LEM, PBC and $\neg\neg$ e are **inter-derivable** (with respect to other basic proof rules)

Proof Rules of Natural Deduction ①

introduction

elimination

\wedge

$$\frac{\varphi \quad \psi}{\varphi \wedge \psi} \wedge i$$

$$\frac{\varphi \wedge \psi}{\varphi} \wedge e_1 \quad \frac{\varphi \wedge \psi}{\psi} \wedge e_2$$

\vee

$$\frac{\varphi}{\varphi \vee \psi} \vee i_1 \quad \frac{\psi}{\varphi \vee \psi} \vee i_2$$

$$\frac{\varphi \vee \psi \quad \begin{array}{|c|} \hline \varphi \\ \vdots \\ \chi \\ \hline \end{array} \quad \begin{array}{|c|} \hline \psi \\ \vdots \\ \chi \\ \hline \end{array}}{\chi} \vee e$$

\rightarrow

$$\frac{\begin{array}{|c|} \hline \varphi \\ \vdots \\ \psi \\ \hline \end{array}}{\varphi \rightarrow \psi} \rightarrow i$$

$$\frac{\varphi \rightarrow \psi \quad \varphi}{\psi} \rightarrow e$$

Proof Rules of Natural Deduction ②

introduction

elimination

\perp

$$\boxed{\begin{array}{c} \varphi \\ \vdots \\ \perp \end{array}}$$

$$\frac{\perp}{\varphi} \perp e$$

\neg

$$\frac{\perp}{\neg\varphi} \neg i$$

$$\frac{\varphi \quad \neg\varphi}{\perp} \neg e$$

\top

$$\frac{}{\top} \top i$$

$$\frac{\neg\neg\varphi}{\varphi} \neg\neg e$$

$\neg\neg$

$$\boxed{\begin{array}{c} \neg\varphi \\ \vdots \\ \perp \end{array}}$$

$$\frac{}{\varphi} \text{PBC}$$

$$\frac{\varphi}{\neg\neg\varphi} \neg\neg i$$

$$\frac{}{\varphi \vee \neg\varphi} \text{LEM}$$

derived proof rules

$$\frac{\varphi \rightarrow \psi \quad \neg\psi}{\neg\varphi} \text{MT}$$

Example

$((p \rightarrow q) \rightarrow p) \rightarrow p$ is valid:

1	$(p \rightarrow q) \rightarrow p$	assumption
2	$p \vee \neg p$	LEM
3	p	assumption
4	$\neg p$	assumption
5	p	assumption
6	\perp	\neg e 5, 4
7	q	\perp e 6
8	$p \rightarrow q$	\rightarrow i 5-7
9	p	\rightarrow e 1, 8
10	p	\vee e 2, 3-3, 4-9
11	$((p \rightarrow q) \rightarrow p) \rightarrow p$	\rightarrow i 1-10

Part I: Propositional Logic

algebraic normal forms, **binary decision diagrams**, conjunctive normal forms, DPLL, Horn formulas, natural deduction, Post's adequacy theorem, **resolution**, SAT, semantics, sorting networks, soundness and **completeness**, syntax, Tseitin's transformation

Part II: Predicate Logic

natural deduction, quantifier equivalences, resolution, semantics, Skolemization, syntax, undecidability, unification

Part III: Model Checking

adequacy, branching-time temporal logic, CTL*, fairness, linear-time temporal logic, model checking algorithms, symbolic model checking

Outline

1. Summary of Previous Lecture
- 2. Completeness**
3. Resolution
4. Intermezzo
5. Binary Decision Diagrams
6. Further Reading

Theorem

natural deduction is **complete**:

$$\varphi_1, \varphi_2, \dots, \varphi_n \models \psi \implies \varphi_1, \varphi_2, \dots, \varphi_n \vdash \psi \text{ is valid}$$

all true statements can be proved

Proof structure

- | | | |
|--|----------------|-------------------|
| ① $\varphi_1, \varphi_2, \dots, \varphi_n \models \psi$ | | assumption |
| ② $\models \varphi_1 \rightarrow (\varphi_2 \rightarrow (\dots (\varphi_n \rightarrow \psi) \dots))$ | ① \implies ② | easy |
| ③ $\vdash \varphi_1 \rightarrow (\varphi_2 \rightarrow (\dots (\varphi_n \rightarrow \psi) \dots))$ is valid | ② \implies ③ | difficult |
| ④ $\varphi_1, \varphi_2, \dots, \varphi_n \vdash \psi$ is valid | ③ \implies ④ | easy |

$$\varphi_1, \varphi_2, \dots, \varphi_n \vDash \psi \quad \implies \quad \vDash \varphi_1 \rightarrow (\varphi_2 \rightarrow (\dots (\varphi_n \rightarrow \psi) \dots))$$

Proof

- ▶ suppose $\vDash \varphi_1 \rightarrow (\varphi_2 \rightarrow (\dots (\varphi_n \rightarrow \psi) \dots))$ does not hold
- ▶ $\bar{v}(\varphi_1) = \dots = \bar{v}(\varphi_n) = \text{T}$ and $\bar{v}(\psi) = \text{F}$ for some valuation v
- ▶ $\varphi_1, \varphi_2, \dots, \varphi_n \vDash \psi$ does not hold

$\vdash \varphi_1 \rightarrow (\varphi_2 \rightarrow (\dots (\varphi_n \rightarrow \psi) \dots))$ is valid $\implies \varphi_1, \varphi_2, \dots, \varphi_n \vdash \psi$ is valid

Proof

- ▶ Π : proof of validity of $\vdash \varphi_1 \rightarrow (\varphi_2 \rightarrow (\dots (\varphi_n \rightarrow \psi) \dots))$
- ▶ proof of validity of $\varphi_1, \varphi_2, \dots, \varphi_n \vdash \psi$:

$\varphi_1 \varphi_2 \dots \varphi_n$	premises
Π	
$\varphi_1 \rightarrow (\varphi_2 \rightarrow (\dots (\varphi_n \rightarrow \psi) \dots))$	
$\varphi_2 \rightarrow (\dots (\varphi_n \rightarrow \psi) \dots)$	$\rightarrow e$
\vdots	\vdots
ψ	$\rightarrow e$

$\models \varphi \implies \vdash \varphi$ is valid

Definition

valuation v , formula φ

$$\langle \varphi \rangle^v = \begin{cases} \varphi & \text{if } \bar{v}(\varphi) = T \\ \neg \varphi & \text{if } \bar{v}(\varphi) = F \end{cases}$$

Main Lemma

p_1, \dots, p_n are all atoms in $\varphi \implies \langle p_1 \rangle^v, \dots, \langle p_n \rangle^v \vdash \langle \varphi \rangle^v$ is valid

every line in truth table corresponds to valid sequent

Main Lemma

p_1, \dots, p_n are all atoms in $\varphi \implies \langle p_1 \rangle^v, \dots, \langle p_n \rangle^v \vdash \langle \varphi \rangle^v$ is valid

Proof (cont'd on subsequent slides)

induction on structure of φ

Example

formula $\varphi = \neg p \vee q$

valuation	p	q	sequent
v_1	T	T	$p, q \vdash \neg p \vee q$
v_2	T	F	$p, \neg q \vdash \neg(\neg p \vee q)$
v_3	F	T	$\neg p, q \vdash \neg p \vee q$
v_4	F	F	$\neg p, \neg q \vdash \neg p \vee q$

Base Cases

$$\varphi = p$$

$$\triangleright v(p) = \text{T}: \quad \langle p \rangle^v = \langle \varphi \rangle^v = p \quad p \vdash p \quad \text{is valid}$$

$$\triangleright v(p) = \text{F}: \quad \langle p \rangle^v = \langle \varphi \rangle^v = \neg p \quad \neg p \vdash \neg p \quad \text{is valid}$$

$$\varphi = \text{T} \quad \vdash \text{T} \quad \text{is valid}$$

$$\varphi = \perp \quad \vdash \neg \perp \quad \text{is valid}$$

Induction Step (4 cases)

$$\text{case 1: } \varphi = \neg \psi$$

$$\text{induction hypothesis: } \langle p_1 \rangle^v, \dots, \langle p_n \rangle^v \vdash \langle \psi \rangle^v \text{ is valid} \quad \text{— } \square$$

$$\triangleright \bar{v}(\varphi) = \text{T}: \quad \langle \varphi \rangle^v = \varphi = \neg \psi = \langle \psi \rangle^v$$

$$\triangleright \bar{v}(\varphi) = \text{F}: \quad \langle \varphi \rangle^v = \neg \varphi = \neg \neg \psi \text{ and } \langle \psi \rangle^v = \psi$$

extend \square with $\neg \neg$ to get proof of $\langle p_1 \rangle^v, \dots, \langle p_n \rangle^v \vdash \langle \varphi \rangle^v$

Induction Step (4 cases)

case 2: $\varphi = \psi_1 \wedge \psi_2$

- ▶ q_1, \dots, q_l : all atoms in ψ_1 r_1, \dots, r_k : all atoms in ψ_2
- ▶ induction hypothesis: $\langle q_1 \rangle^v, \dots, \langle q_l \rangle^v \vdash \langle \psi_1 \rangle^v$ and $\langle r_1 \rangle^v, \dots, \langle r_k \rangle^v \vdash \langle \psi_2 \rangle^v$ are valid
- ▶ $\langle p_1 \rangle^v, \dots, \langle p_n \rangle^v \vdash \langle \psi_1 \rangle^v$ and $\langle p_1 \rangle^v, \dots, \langle p_n \rangle^v \vdash \langle \psi_2 \rangle^v$ are valid
- ▶ $\langle p_1 \rangle^v, \dots, \langle p_n \rangle^v \vdash \langle \psi_1 \rangle^v \wedge \langle \psi_2 \rangle^v$ is valid (using $\wedge i$) — Π
- ▶ to prove: $\langle \psi_1 \rangle^v \wedge \langle \psi_2 \rangle^v \vdash \langle \varphi \rangle^v$ is valid — Π'

$\bar{v}(\psi_1)$	$\bar{v}(\psi_2)$	$\langle \psi_1 \rangle^v \wedge \langle \psi_2 \rangle^v$	$\langle \varphi \rangle^v$
T	T	$\psi_1 \wedge \psi_2$	$\psi_1 \wedge \psi_2$
T	F	$\psi_1 \wedge \neg\psi_2$	$\neg(\psi_1 \wedge \psi_2)$
F	T	$\neg\psi_1 \wedge \psi_2$	$\neg(\psi_1 \wedge \psi_2)$
F	F	$\neg\psi_1 \wedge \neg\psi_2$	$\neg(\psi_1 \wedge \psi_2)$

- ▶ combining Π and Π' yields validity of $\langle p_1 \rangle^v, \dots, \langle p_n \rangle^v \vdash \langle \varphi \rangle^v$

Induction Step (4 cases)

case 3: $\varphi = \psi_1 \vee \psi_2$

- ▶ $\langle p_1 \rangle^v, \dots, \langle p_n \rangle^v \vdash \langle \psi_1 \rangle^v$ and $\langle p_1 \rangle^v, \dots, \langle p_n \rangle^v \vdash \langle \psi_2 \rangle^v$ are valid
- ▶ $\langle p_1 \rangle^v, \dots, \langle p_n \rangle^v \vdash \langle \psi_1 \rangle^v \wedge \langle \psi_2 \rangle^v$ is valid (using $\wedge i$) — Π
- ▶ to prove: $\langle \psi_1 \rangle^v \wedge \langle \psi_2 \rangle^v \vdash \langle \varphi \rangle^v$ is valid — Π'

$\bar{v}(\psi_1)$	$\bar{v}(\psi_2)$	$\langle \psi_1 \rangle^v \wedge \langle \psi_2 \rangle^v$	$\langle \varphi \rangle^v$
T	T	$\psi_1 \wedge \psi_2$	$\psi_1 \vee \psi_2$
T	F	$\psi_1 \wedge \neg \psi_2$	$\psi_1 \vee \psi_2$
F	T	$\neg \psi_1 \wedge \psi_2$	$\psi_1 \vee \psi_2$
F	F	$\neg \psi_1 \wedge \neg \psi_2$	$\neg(\psi_1 \vee \psi_2)$

- ▶ combining Π and Π' yields validity of $\langle p_1 \rangle^v, \dots, \langle p_n \rangle^v \vdash \langle \varphi \rangle^v$

Induction Step (4 cases)

case 4: $\varphi = \psi_1 \rightarrow \psi_2$

- ▶ $\langle p_1 \rangle^v, \dots, \langle p_n \rangle^v \vdash \langle \psi_1 \rangle^v$ and $\langle p_1 \rangle^v, \dots, \langle p_n \rangle^v \vdash \langle \psi_2 \rangle^v$ are valid
- ▶ $\langle p_1 \rangle^v, \dots, \langle p_n \rangle^v \vdash \langle \psi_1 \rangle^v \wedge \langle \psi_2 \rangle^v$ is valid (using \wedge i) — Π
- ▶ to prove: $\langle \psi_1 \rangle^v \wedge \langle \psi_2 \rangle^v \vdash \langle \varphi \rangle^v$ is valid — Π'

$\bar{v}(\psi_1)$	$\bar{v}(\psi_2)$	$\langle \psi_1 \rangle^v \wedge \langle \psi_2 \rangle^v$	$\langle \varphi \rangle^v$
T	T	$\psi_1 \wedge \psi_2$	$\psi_1 \rightarrow \psi_2$
T	F	$\psi_1 \wedge \neg \psi_2$	$\neg(\psi_1 \rightarrow \psi_2)$
F	T	$\neg \psi_1 \wedge \psi_2$	$\psi_1 \rightarrow \psi_2$
F	F	$\neg \psi_1 \wedge \neg \psi_2$	$\psi_1 \rightarrow \psi_2$

- ▶ combining Π and Π' yields validity of $\langle p_1 \rangle^v, \dots, \langle p_n \rangle^v \vdash \langle \varphi \rangle^v$

Theorem

$\models \varphi \implies \vdash \varphi$ is valid

Proof

suppose $\models \varphi$

- ▶ for every valuation v $\langle \varphi \rangle^v = \varphi$
- ▶ for every valuation v $\langle p_1 \rangle^v, \dots, \langle p_n \rangle^v \vdash \varphi$ is valid sequent
- ▶ combine all proofs of these sequents into proof of validity of

$\vdash \varphi$

by $2^n - 1$ applications of LEM and $\forall e$

Example

$\vdash p \wedge q \rightarrow q$ is valid

valuation	p	q	sequent	proof
v_1	T	T	$p, q \vdash p \wedge q \rightarrow q$	Π_1
v_2	T	F	$p, \neg q \vdash p \wedge q \rightarrow q$	Π_2
v_3	F	T	$\neg p, q \vdash p \wedge q \rightarrow q$	Π_3
v_4	F	F	$\neg p, \neg q \vdash p \wedge q \rightarrow q$	Π_4

$p \vee \neg p$ LEM

p assumption	$\neg p$ assumption												
$q \vee \neg q$ LEM	$q \vee \neg q$ LEM												
<table border="1"> <tr> <td>q ass</td> <td>$\neg q$ ass</td> </tr> <tr> <td>$\dots \Pi_1 \dots$</td> <td>$\dots \Pi_2 \dots$</td> </tr> <tr> <td>$p \wedge q \rightarrow q$</td> <td>$p \wedge q \rightarrow q$</td> </tr> </table>	q ass	$\neg q$ ass	$\dots \Pi_1 \dots$	$\dots \Pi_2 \dots$	$p \wedge q \rightarrow q$	$p \wedge q \rightarrow q$	<table border="1"> <tr> <td>q ass</td> <td>$\neg q$ ass</td> </tr> <tr> <td>$\dots \Pi_3 \dots$</td> <td>$\dots \Pi_4 \dots$</td> </tr> <tr> <td>$p \wedge q \rightarrow q$</td> <td>$p \wedge q \rightarrow q$</td> </tr> </table>	q ass	$\neg q$ ass	$\dots \Pi_3 \dots$	$\dots \Pi_4 \dots$	$p \wedge q \rightarrow q$	$p \wedge q \rightarrow q$
q ass	$\neg q$ ass												
$\dots \Pi_1 \dots$	$\dots \Pi_2 \dots$												
$p \wedge q \rightarrow q$	$p \wedge q \rightarrow q$												
q ass	$\neg q$ ass												
$\dots \Pi_3 \dots$	$\dots \Pi_4 \dots$												
$p \wedge q \rightarrow q$	$p \wedge q \rightarrow q$												
$p \wedge q \rightarrow q \vee e$	$p \wedge q \rightarrow q \vee e$												
$p \wedge q \rightarrow q \vee e$													

Natural Deduction Tool

by Andreas Schnabl (2005)

Outline

1. Summary of Previous Lecture
2. Completeness
- 3. Resolution**
4. Intermezzo
5. Binary Decision Diagrams
6. Further Reading

Definitions

- ▶ **clause** is set of literals $\{l_1, \dots, l_n\}$ representing formula

$$\begin{cases} (l_1 \vee \dots \vee l_n) & \text{if } n \geq 1 \\ \perp & \text{if } n = 0 \end{cases}$$

- ▶ \square denotes **empty clause** \emptyset
- ▶ **clausal form** is set of clauses $\{C_1, \dots, C_m\}$ representing formula

$$\begin{cases} C_1 \wedge \dots \wedge C_m & \text{if } m \geq 1 \\ \top & \text{if } m = 0 \end{cases}$$

Remark

every CNF can be written in clausal form

Example

CNF

clausal form

$(\neg p) \wedge (\neg q \vee \neg p) \wedge (\neg p \vee \neg r)$	$\{\{\neg p\}, \{\neg q, \neg p\}, \{\neg p, \neg r\}\}$
$(\neg p \vee q) \wedge (q \vee \neg r) \wedge (p \vee q \vee \neg r)$	$\{\{\neg p, q\}, \{q, \neg r\}, \{p, q, \neg r\}\}$
$(\neg p \vee \neg p) \wedge (q \vee r) \wedge (r \vee q)$	$\{\{\neg p\}, \{q, r\}\}$

Definition

literals l_1 and l_2 are **complementary** if $\underbrace{l_1 = \neg l_2 \text{ or } \neg l_1 = l_2}_{l_1 = l_2^c}$

Notation

if l is literal then $l^c = \begin{cases} \neg p & \text{if } l = p \\ p & \text{if } l = \neg p \end{cases}$

Definition

- ▶ clauses C_1 and C_2 **clash** on literal l if $l \in C_1$ and $l^c \in C_2$
- ▶ **resolvent** of clauses C_1 and C_2 clashing on literal l is clause

$$(C_1 \setminus \{l\}) \cup (C_2 \setminus \{l^c\})$$

- ▶ C_1 and C_2 are **parent clauses** of resolvent

Examples

- ▶ clauses $\{\neg p, \neg q, r\}$ and $\{p, r\}$ clash with resolvent $\{\neg q, r\}$
- ▶ clauses $\{p, \neg p, q\}$ and $\{q, r\}$ do not clash
- ▶ clauses $\{\neg p, \neg q\}$ and $\{p, q\}$ clash with resolvents $\{\neg q, q\}$ and $\{\neg p, p\}$
- ▶ clauses $\{p, \neg p, q\}$ and $\{p, \neg p, q\}$ clash with resolvent $\{p, \neg p, q\}$

Resolution

input: clausal form S

output: yes if S is satisfiable

no if S is unsatisfiable

- ① repeatedly add (new) resolvents of clashing clauses in S
- ② return no as soon as empty clause is derived
- ③ return yes if all clashing clauses have been resolved

Theorem

resolution is terminating

Definition

refutation of S is resolution derivation of \square from S

Theorem

resolution is sound and complete:

S admits refutation \iff clausal form S is unsatisfiable

Definition

- ▶ resolvent of clauses C_1 and C_2 clashing on literal l is clause $(C_1 \setminus \{l\}) \cup (C_2 \setminus \{l^c\})$
- ▶ special case (**unit resolution**): $C_1 = \{l\}$ with resolvent $C_2 \setminus \{l^c\}$

Example 1

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

1 $\{\neg p, \neg q, r\}$

2 $\{p, r\}$

3 $\{q, r\}$

4 $\{\neg r\}$

5 $\{\neg q, r\}$ resolve 1, 2, p

6 $\{r\}$ resolve 3, 5, q

7 \square resolve 4, 6, r

unsatisfiable

Example 2

$$p \wedge (p \rightarrow ((q \vee r) \wedge \neg(q \wedge r))) \wedge (p \rightarrow ((s \vee t) \wedge \neg(s \wedge t))) \wedge (s \rightarrow q) \wedge (\neg r \rightarrow t) \wedge (t \rightarrow s)$$

1	{ p }		
2	{ $\neg p, q, r$ }		
3	{ $\neg p, \neg q, \neg r$ }		
4	{ $\neg p, s, t$ }		
5	{ $\neg p, \neg s, \neg t$ }		
6	{ $\neg s, q$ }		
7	{ r, t }		
8	{ $\neg t, s$ }		
9	{ $\neg q, \neg r$ }	resolve 1, 3, p	
10	{ s, t }	resolve 1, 4, p	
11	{ $\neg s, \neg t$ }	resolve 1, 5, p	
12	{ $\neg s, \neg r$ }	resolve 6, 9, q	
13	{ s }	resolve 8, 10, t	
14	{ $\neg t$ }	resolve 11, 13, s	
15	{ r }	resolve 7, 14, t	
16	{ $\neg r$ }	resolve 12, 13, s	
17	\square	resolve 15, 16, r	

unsatisfiable

Example 3

$$(p \vee q) \wedge (\neg p \vee \neg r) \wedge (\neg q \vee s) \wedge (p \vee \neg r) \wedge (r \vee \neg s)$$

1	$\{p, q\}$		12	$\{\neg p, \neg q\}$	resolve 3, 9, s	23	$\{q, \neg q\}$	resolve 6, 11, r
2	$\{\neg p, \neg r\}$		13	$\{p, \neg s\}$	resolve 4, 5, r	24	$\{p, \neg p\}$	resolve 7, 9, s
3	$\{\neg q, s\}$		14	$\{\neg r, \neg s\}$	resolve 4, 9, p	25	$\{p\}$	resolve 7, 19, s
4	$\{p, \neg r\}$		15	$\{p, \neg q\}$	resolve 4, 11, r	26	$\{\neg q\}$	resolve 8, 11, r
5	$\{r, \neg s\}$		16	$\{\neg q, \neg r\}$	resolve 4, 12, p			
6	$\{q, \neg r\}$	resolve 1, 2, p	17	$\{q, \neg s\}$	resolve 5, 6, r			no further resolvents
7	$\{p, s\}$	resolve 1, 3, q	18	$\{p, r\}$	resolve 5, 7, s			\implies
8	$\{\neg r\}$	resolve 2, 4, p	19	$\{\neg s\}$	resolve 5, 8, r			satisfiable
9	$\{\neg p, \neg s\}$	resolve 2, 5, r	20	$\{s, \neg s\}$	resolve 5, 10, r			
10	$\{\neg r, s\}$	resolve 2, 7, p	21	$\{r, \neg r\}$	resolve 5, 10, s			
11	$\{\neg q, r\}$	resolve 3, 5, s	22	$\{\neg q, \neg s\}$	resolve 5, 16, r			

Example 4

$$\neg \left((p \vee q) \wedge (\neg s \vee \neg u) \wedge (\neg r \vee \neg w) \wedge (t \vee u) \wedge (\neg p \vee \neg r) \wedge (\neg q \vee \neg s) \right. \\ \left. \wedge (\neg p \vee \neg t) \wedge (\neg q \vee \neg u) \wedge (r \vee s) \wedge (\neg t \vee \neg v) \wedge (v \vee w) \right)$$

1	$\{p, q\}$	10	$\{\neg t, \neg v\}$	19	$\{\neg w\}$	resolve 3, 18, r
2	$\{\neg s, \neg u\}$	11	$\{v, w\}$	20	$\{v\}$	resolve 11, 19, w
3	$\{\neg r, \neg w\}$	12	$\{\neg s, t\}$	21	$\{\neg t\}$	resolve 10, 20, v
4	$\{t, u\}$	13	$\{\neg p, s\}$	22	$\{u\}$	resolve 4, 21, t
5	$\{\neg p, \neg r\}$	14	$\{\neg p, \neg s\}$	23	$\{\neg q\}$	resolve 8, 22, u
6	$\{\neg q, \neg s\}$	15	$\{\neg p\}$	24	\square	resolve 16, 23, q
7	$\{\neg p, \neg t\}$	16	$\{q\}$			valid
8	$\{\neg q, \neg u\}$	17	$\{\neg s\}$			
9	$\{r, s\}$	18	$\{r\}$			

Outline

1. Summary of Previous Lecture
2. Completeness
3. Resolution
- 4. Intermezzo**
5. Binary Decision Diagrams
6. Further Reading

Question

Which of these clauses **cannot** be obtained in a single resolution step from the following clausal form ?

$$\{\{p, q, r\}, \{\neg r, q, \neg p\}, \{\neg q, \neg s\}, \{q, \neg p, r\}\}$$

- A $\{p, q, \neg p\}$
- B $\{\neg r, \neg p, \neg s\}$
- C $\{q\}$
- D $\{q, \neg p\}$
- E $\{q, r\}$
- F $\{\neg p, r, \neg s\}$



Outline

1. Summary of Previous Lecture
2. Completeness
3. Resolution
4. Intermezzo
- 5. Binary Decision Diagrams**
6. Further Reading

Definitions

- ▶ **boolean function** f of n arguments is mapping from $\{0, 1\}^n$ to $\{0, 1\}$
- ▶ four basic functions

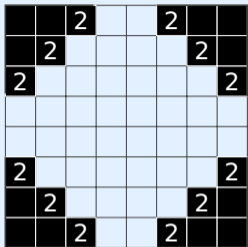
complement	$-$	x	\bar{x}	x	y	xy	$x+y$	$x \oplus y$
product	\cdot	0	1	0	0	0	0	0
sum	$+$	1	0	0	1	0	1	1
exclusive or	\oplus			1	0	0	1	1
				1	1	1	1	0

(xy denotes $x \cdot y$)

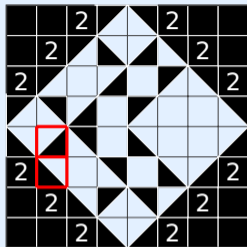
Remarks

- ▶ every boolean function can be expressed in terms of basic functions
- ▶ propositional formulas and truth tables are different **representations** of boolean functions

Exclusive Or or Or



Shakashaka



```
(assert (=> (= x0y3 SW) (and
  (= x0y4 NW)
  (xor (= x1y2 SW) (= x1y3 SE))
  (or (= x1y4 W) (= x1y4 NE))
)))
```

Lemma

▶ $x \oplus y = y \oplus x$

▶ $x \oplus x = 0$

▶ $x \oplus y = \bar{x}y + x\bar{y}$

▶ $(x \oplus y) \oplus z = x \oplus (y \oplus z)$

▶ $(x \oplus y)z = xz \oplus yz$

▶ $x \oplus y = (x + y)(\bar{x} + \bar{y})$

Representations of Boolean Functions

representation	test for			boolean operation		
	compact?	satisfiability	validity	product	sum	complement
propositional formulas	often	hard	hard	easy	easy	easy
truth tables	never	hard	hard	hard	hard	hard
CNFs	sometimes	hard	easy	easy	hard	hard
DNFs	sometimes	easy	hard	hard	easy	hard
?	often	easy	easy	medium	medium	easy

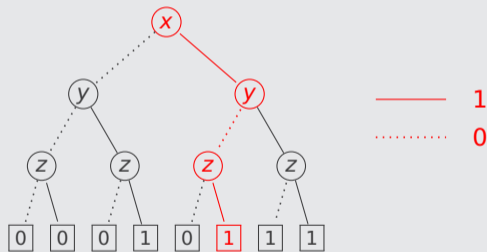
? = reduced ordered binary decision diagrams

Example

- majority function

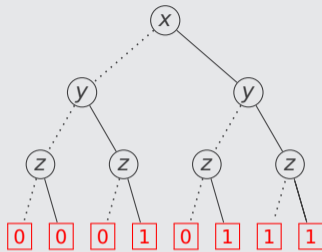
$$f(x,y,z) = \begin{cases} 1 & \text{if } x + y + z > 1 \\ 0 & \text{otherwise} \end{cases}$$

- binary decision tree for f

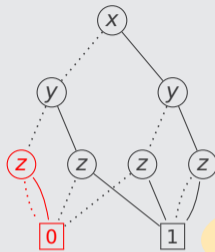


- $f(1, 0, 1) = 1$

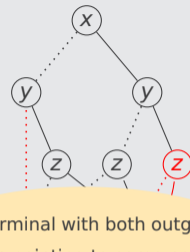
Example (Binary Decision Diagram)



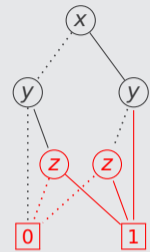
\Rightarrow
C1



\Rightarrow
C2

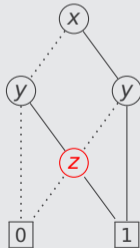


\Rightarrow
C2



non-terminal with both outgoing edges pointing to same node

\Rightarrow
C3



Optimisation Rules

- C1 remove duplicate terminals
- C2 remove redundant tests
- C3 remove duplicate non-terminals

Remark

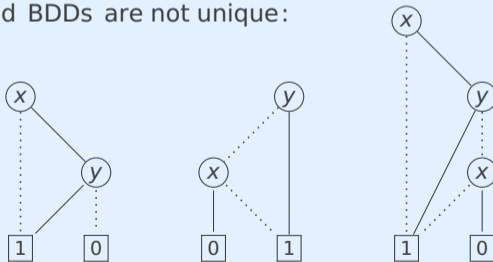
binary decision diagram (BDD) is directed acyclic graph (dag)

Definition

BDD is **reduced** if C1, C2, C3 are not applicable

Remark


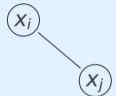
reduced BDDs are not unique:



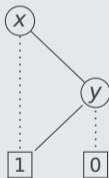
represent boolean function $\bar{x} + y$

Definition

BDD B is **ordered** if there exists order $[x_1, \dots, x_n]$ of variables in B such that

$i < j$ for all edges  and  in B

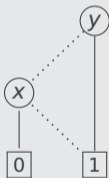
Examples



OBDD

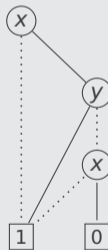
$[x, y]$

$[z, x, y]$ $[x, z, y]$ $[x, y, z]$



OBDD

$[y, x]$



not ordered

Definition

orders o_1 and o_2 are **compatible** if o_1 and o_2 are subsequences of some order o

Example

four variable orders

$$o_1 = [x, y, z]$$

$$o_2 = [x, v]$$

$$o_3 = [z, x]$$

$$o_4 = [v, z, w]$$

- ▶ o_1 and o_2 are compatible (e.g. $o = [x, y, v, z]$)
- ▶ o_1 and o_3 are not compatible
- ▶ o_2 and o_4 are compatible

Lemma

reductions C1, C2, C3 preserve order

Theorem

reduced OBDD representation of boolean function for given order is **unique**

Corollary

checking

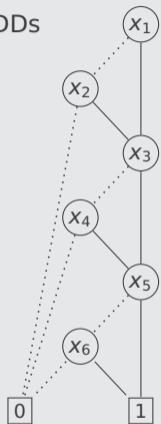
- ▶ satisfiability
- ▶ validity
- ▶ equivalence

is **trivial** for reduced OBDDs (with compatible variable orders)

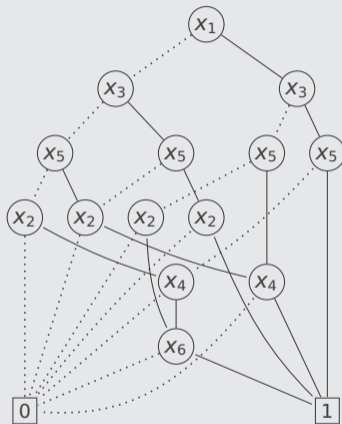
Example

$$f(x_1, x_2, x_3, x_4, x_5, x_6) = (x_1 + x_2) \cdot (x_3 + x_4) \cdot (x_5 + x_6)$$

different reduced OBDDs



$[x_1, x_2, x_3, x_4, x_5, x_6]$



$[x_1, x_3, x_5, x_2, x_4, x_6]$

Outline

1. Summary of Previous Lecture
2. Completeness
3. Resolution
4. Intermezzo
5. Binary Decision Diagrams
- 6. Further Reading**

Huth and Ryan

- ▶ Section 1.4.4
- ▶ Section 6.1

Resolution

- ▶ Wikipedia

[accessed December 27, 2024]

Important Concepts

- ▶ binary decision diagram
- ▶ binary decision tree
- ▶ boolean function
- ▶ complementary literals
- ▶ completeness
- ▶ clashing
- ▶ clausal form
- ▶ clause
- ▶ compatible variable order
- ▶ empty clause
- ▶ exclusive or
- ▶ ordered BDD
- ▶ parent clauses
- ▶ reduced BDD
- ▶ refutation
- ▶ resolution
- ▶ resolvent
- ▶ variable order

homework for April 16