Aussagenlogik

- Syntax
- Semantik
- **Formeln, Modelle, Tautologien und Anwendungen**

Modelle und Gegenbeispiele

Eine für eine Formel φ ausreichende **Belegung** *bel*
ist Modell von φ :⇔ $wert_{bel}(\varphi) = \mathrm{W}$.

Man sagt/schreibt dafür auch: *bel* **erfüllt** φ
bzw. φ **gilt unter** *bel*
bzw. $bel \models \varphi$.

Beispiel: Jede Belegung *bel* mit $bel(A) := \mathrm{F}$, $bel(B) := \mathrm{W}$ ist Modell von $A \rightarrow B$.

bel ist **Gegenbeispiel** zu φ :⇔ $wert_{bel}(\varphi) = F$
Man sagt/schreibt auch: *bel* **widerlegt** φ bzw. $bel \not\models \varphi$.

Beispiel: Jede Belegung *bel* mit $bel(A) := \mathrm{W}$, $bel(B) := \mathrm{F}$ ist ein Gegenbeispiel für $A \rightarrow B$.

Die **Zeilen** der Wahrheitstafel, in denen unter der Formel **W** (bzw. **F**) steht, enthalten unter den Aussagevariablen die **Modelle** (bzw. **Gegenbeispiele**).

Der Wahrheitstafelalgorithmus liefert alle Modelle und Gegenbeispiele einer Formel.

Ggf. Belegung fortsetzen!

Semantische Kategorien von Formeln (1)

Existenz von Modellen und Gegenbeispielen

Eine Formel φ nennt man **erfüllbar**, wenn sie ein Modell besitzt, und **unerfüllbar** (oder **widersprüchlich**) wenn nicht.

Beispiele: $A \rightarrow B$ ist erfüllbar;
$A \wedge \neg A$ ist unerfüllbar.

Eine Formel φ nennt man **kontingent**, wenn sie Modell(e) und Gegenbeispiel(e) besitzt.

Beispiel: $A \wedge B$ ist kontingent.

Eine Formel nennt man **allgemeingültig** (oder eine **Tautologie**) und schreibt dann $\models \varphi$, wenn sie unter jeder für sie ausreichenden Belegung wahr ist, und **widerlegbar** (oder **falsifizierbar,** $\not\models \varphi$) wenn nicht.

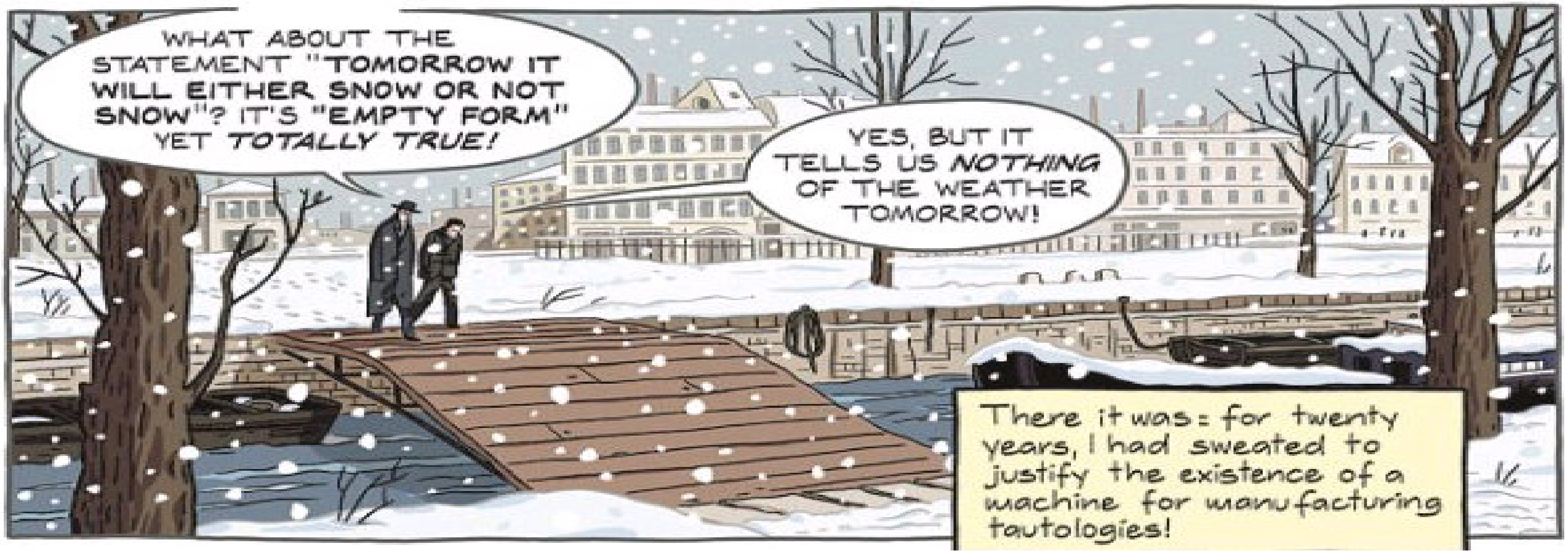
Beispiele: $(A \rightarrow B) \vee (B \rightarrow A)$ ist allgemeingültig und erfüllbar;
$(A \rightarrow B) \wedge (B \rightarrow A)$ ist widerlegbar und erfüllbar
und daher kontingent.

Ü16

Semantische Kategorien von Formeln (2)

Übersicht → Wahrheitstafel

$$\varphi \text{ ist } \left\{\begin{array}{c} \text{erfüllbar}\\ \text{unerfüllbar}\\ \text{kontingent}\\ \text{allgemeingültig}\\ \text{widerlegbar} \end{array}\right\} \Leftrightarrow \text{In } \varphi\text{-Spalte d. W.-tafel kommt } \left\{\begin{array}{c} \text{W}\\ \text{nur F}\\ \text{W und F}\\ \text{nur W}\\ \text{F} \end{array}\right\} \text{ vor.}$$



Doxiadis/Papadimitriou/Papadatis: Logicomix: An Epic Search for Truth. Bloomsbury, 2009

Wichtige Tautologien (1): Implikationen

Ex-Falso-Prinzip	$(P \land \neg P) \to Q$
Peirce's Gesetz	$\big((P \to Q) \to P\big) \to P$
Kontraposition	$(P \to Q) \leftrightarrow (\neg Q \to \neg P)$
Kettenschluss	$(P \to Q) \to \big((Q \to R) \to (P \to R)\big)$
Prämissenverbindung	$\big((P \land Q) \to R\big) \leftrightarrow \big(P \to (Q \to R)\big)$
Prämissenvertauschung	$\big(P \to (Q \to R)\big) \leftrightarrow \big(Q \to (P \to R)\big)$
Selbstdistributivität	$((P \to (Q \to R)) \leftrightarrow ((P \to Q) \to (P \to R))$

Implikationen, Beweisbeispiel: Kontraposition

mit normaler und modifizierter Wahrheitstafel:

P	Q	$P \to Q$	$\neg Q$	$\neg P$	$\neg Q \to \neg P$	$(P \to Q) \leftrightarrow (\neg Q \to \neg P)$
W	W	W	F	F	W	**W**
W	F	F	W	F	F	**W**
F	W	W	F	W	W	**W**
F	F	W	W	W	W	**W**

$(P$	$\to$	$Q)$	$\leftrightarrow$	$(\neg$	Q	$\to$	$\neg$	$P)$
W	W	W	**W**	F	W	W	F	W
W	F	F	**W**	W	F	F	F	W
F	W	W	**W**	F	W	W	W	F
F	W	F	**W**	W	F	W	W	F

Wichtige Tautologien (2): Äquivalenzen/Biimplikationen

Regeln für Konjunktion und Disjunktion

$(P \wedge P) \leftrightarrow P$ $\quad$ $(P \vee P) \leftrightarrow P$ ***Idempotenz***

$(P \wedge Q) \leftrightarrow (Q \wedge P)$ $\quad$ $(P \vee Q) \leftrightarrow (Q \vee P)$ ***Kommutativität***

$((P \wedge Q) \wedge R) \leftrightarrow (P \wedge (Q \wedge R))$ ***Assoziativität***

$((P \vee Q) \vee R) \leftrightarrow (P \vee (Q \vee R))$ ***Assoziativität***

$((P \wedge (Q \vee R)) \leftrightarrow ((P \wedge Q) \vee (P \wedge R))$ ***Distributivität***

$((P \vee (Q \wedge R)) \leftrightarrow ((P \vee Q) \wedge (P \vee R))$ **Ü17a** ***Distributivität***

Negationsregeln

$\neg\neg P \leftrightarrow P$ ***doppelte Negation***

$\neg(P \wedge Q) \leftrightarrow (\neg P \vee \neg Q)$ $\quad$ $\neg(P \vee Q) \leftrightarrow (\neg P \wedge \neg Q)$ ***Anti-Distributivität***

Wichtige Tautologien (3): weitere Äquivalenzen/Biimplikationen

Tautologie- und Kontradiktions-(Absorptions-)Regeln

$(P \wedge (Q \vee \neg Q)) \leftrightarrow P$	$(P \vee (Q \vee \neg Q)) \leftrightarrow (Q \vee \neg Q)$	***Tautologie-Absorption***
$(P \vee (Q \wedge \neg Q)) \leftrightarrow P$	$(P \wedge (Q \wedge \neg Q)) \leftrightarrow (Q \wedge \neg Q)$	***Kontradiktion-Absorption***

Implikations- und Äquivalenzauflösungsregeln

$(P \leftrightarrow Q) \leftrightarrow ((P \rightarrow Q) \wedge (Q \rightarrow P))$ ***Äquivalenzauflösung***

$(P \rightarrow Q) \leftrightarrow (\neg P \vee Q)$ **Ü17b** ***Implikationsauflösung***

Formelmengen

Eine Belegung *bel* ist **ausreichend** (oder **passend**) für eine ***Menge*** Φ von Formeln,
wenn sie alle Aussagevariablen aller Formel $\varphi \in \Phi$ belegt, d.h.

für alle $\varphi \in \Phi$: $Vars(\varphi) \subseteq \mathrm{Def}_{bel}$.

Man nennt eine für eine Menge Φ von Formeln ausreichende Belegung, bei der **jede** Formel in Φ wahr ist, ein **Modell von** Φ.

Beispiel:

Sei Φ die Menge $\{ (A \to B_i) \mid i = 1,2,... \}$.
Dann ist jede ausreichende Belegung *bel* von $A, B_1, B_2, ...$
mit $bel(A) = \mathrm{F}$ ein Modell von Φ. (Welche Modelle gibt es noch?)

Eine Menge von Formeln nennt man **erfüllbar,** wenn sie ein Modell besitzt, und **unerfüllbar** (oder **widersprüchlich**) wenn nicht.

Beispiel:

$\{ (A \to B), \neg(C \vee B \vee \neg A) \}$ ist unerfüllbar. (→ Wahrheitstafel(n)!)

Die „Logifizierung“ kombinatorischer Aufgaben – Ein Kurzkrimi –

Der Kriminalinspektor sucht den oder die Täter unter vier Verdächtigen. Bei der Vernehmung werden folgende Aussagen gemacht:

- Ede: Jimmy hat’s getan
- Jimmy: Carlo war’s.
- Bomber: Ich war es jedenfalls nicht.
- Carlo: Jimmy log, als er sagte ich hätt’s getan.

Der Inspektor weiß von der zuverlässigen Räuber-Jenny, dass

- genau eine der Aussagen stimmt und dass
- die vier stets nur einzeln agieren.

Aussagevariablen:

B, C, E, J: „Bomber/ Carlo/ Ede/ Jimmy war der Täter.“

Problem: **Formulierung der Wissensbasis** =
Formelmenge, deren (ausrechenbare!) **Modelle** den/die möglichen Täter benennen – und damit die **Aufgabe lösen**

Wie rechnet der Inspektor?

Die vier Aussagen der Verdächtigen sind J, C, $\neg B$ und $\neg C$.

Sicher sind nur Jennys Tipps:

P: Entweder stimmt genau die erste Aussage oder genau die zweite usw.:

$$(J \wedge \neg C \wedge B \wedge C) \vee (\neg J \wedge C \wedge B \wedge C) \vee (\neg J \wedge \neg C \wedge \neg B \wedge C) \vee (\neg J \wedge \neg C \wedge B \wedge \neg C)$$

Q: Es kommen keine zwei Täter in Frage:

$$(B \wedge \neg C \wedge \neg E \wedge \neg J) \vee (\neg B \wedge C \wedge \neg E \wedge \neg J) \vee$$
$$(\neg B \wedge \neg C \wedge E \wedge \neg J) \vee (\neg B \wedge \neg C \wedge \neg E \wedge J)$$

Wissensbasis = $\{P,Q\}$ bzw. $\{P \wedge Q\}$;

Modell(e) sind zumindest bestimmbar mit 16-zeiliger Wahrheitstafel, mit „Tricks“ auch unaufwändiger.

Als gut gestellte Aufgabe hat sie genau 1 Lösung (= Modelle für B, C, E, J).

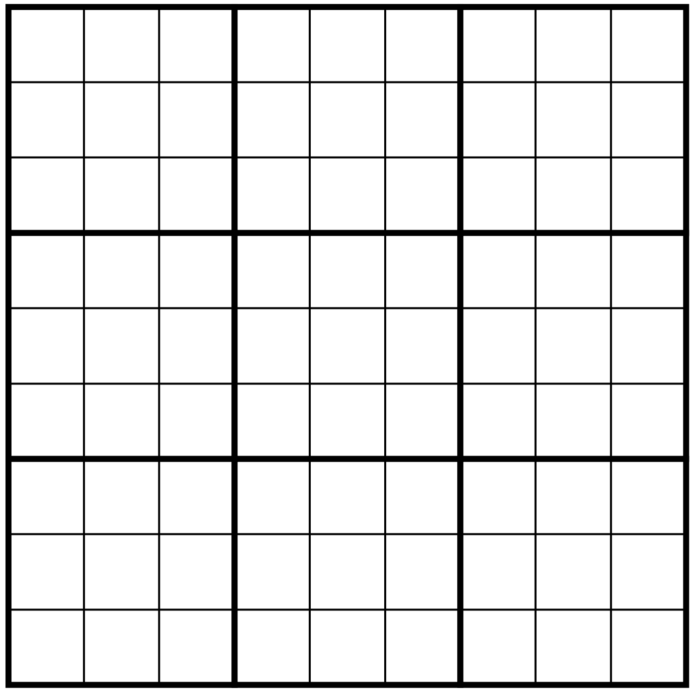
Die „Logifizierung“ kombinatorischer Aufgaben – Sudoku –

Sudoku-Regel (1)
Ein Sudoku-Feld ist eine 9×9-Matrix M, d.h. mit 9 Zeilen und 9 Spalten. Man unterscheidet darin 9 disjunkte Regionen, jede bestehend aus 3×3-Feldern.

Die „Logifizierung“ kombinatorischer Aufgaben
– Sudoku –

Sudoku-Regel (1)
Ein Sudoku-Feld ist eine 9×9-Matrix M, d.h. mit 9 Zeilen und 9 Spalten. Man unterscheidet darin 9 disjunkte Regionen, jede bestehend aus 3×3-Feldern.

Sudoku-Regel (2)
Die möglichen Matrixeinträge sind {leer,1, . . . , 9}. Zu Beginn sind nur einige dieser Felder mit Zahlen gefüllt, also etliche Positionen leer.



Die „Logifizierung“ kombinatorischer Aufgaben – Sudoku –

Sudoku-Regel (1)
Ein Sudoku-Feld ist eine 9×9-Matrix M, d.h. mit 9 Zeilen und 9 Spalten. Man unterscheidet darin 9 disjunkte Regionen, jede bestehend aus 3×3-Feldern.

8				2	9	5		7
	5						8	
9		4			5	2		
1		3						
4				3				8
						7		6
		5	6			8		3
	7						9	
3		1	4	7				2

Sudoku-Regel (2)
Die möglichen Matrixeinträge sind {leer,1, . . . , 9}. Zu Beginn sind nur einige dieser Felder mit Zahlen gefüllt, also etliche Positionen leer.

Sudoku-Regel (3)
Die Aufgabe besteht darin, das Feld vollständig so auszufüllen, d.h. die M_{ik}=leer, so zu verändern, dass in jeder Zeile, jeder Spalte und jeder Region jeweils jede Zahl zwischen 1 und 9 genau einmal vorkommt.

Die „Logifizierung“ kombinatorischer Aufgaben – Sudoku –

Sudoku-Regel (1)
Ein Sudoku-Feld ist eine 9×9-Matrix M, d.h. mit 9 Zeilen und 9 Spalten. Man unterscheidet darin 9 disjunkte Regionen, jede bestehend aus 3×3 Feldern.

8	1	6	3	2	9	5	4	7
7	5	2	1	4	6	3	8	9
9	3	4	7	8	5	2	6	1
1	8	3	9	6	7	4	2	5
4	6	7	5	3	2	9	1	8
5	2	9	8	1	4	7	3	6
2	4	5	6	9	1	8	7	3
6	7	8	2	5	3	1	9	4
3	9	1	4	7	8	6	5	2

Sudoku-Regel (2)
Die möglichen Matrixeinträge sind {leer,1, . . . , 9}. Zu Beginn sind nur einige dieser Felder mit Zahlen gefüllt, also etliche Positionen leer.

Sudoku-Regel (3)
Die Aufgabe besteht darin, das Feld vollständig so auszufüllen, d.h. die M_{ik}=leer, so zu verändern, dass in jeder Zeile, jeder Spalte und jeder Region jeweils jede Zahl zwischen 1 und 9 genau einmal vorkommt.

Sudoku als Logik-Problem (1)

Wir verwenden für die Beschreibung der Anforderungen an die Lösung 729 atomaren Aussagevariablen A_{ijk}, $i, j, k = 1, \ldots, 9$, und eine Belegung *belLsg* mit

$$belLsg(A_{ijk}) = W \quad \Leftrightarrow \quad M_{ij} = k \quad ,$$

d.h. A_{ijk} besagt: « Auf Feld (i , j) steht die Zahl k. »

Nun können wir die Lösungs-Matrix-Eigenschaften und Sudoku-Spielregeln mit Hilfe der Aussagenlogik **formalisieren**:

(i) Auf jedem Feld befindet sich eine Zahl:

z.B. für ij =11: $\text{ZahlDrin}_{11} :\Leftrightarrow A_{111} \vee A_{112} \vee \cdots \vee A_{119}$

also 81 Formeln: $\text{ZahlDrin}_{11} \wedge \text{ZahlDrin}_{12} \wedge \ldots \wedge \text{ZahlDrin}_{99}$

(ii) Auf keinem Feld befinden sich zwei Zahlen:

z.B. für ij=11: $\text{NieZwei}_{11} :\Leftrightarrow \neg(A_{111} \wedge A_{112}) \wedge \neg(A_{111} \wedge A_{113}) \wedge \cdots \wedge \neg(A_{118} \wedge A_{119})$

also : $\text{NieZwei}_{11} \wedge \text{NieZwei}_{12} \wedge \ldots \wedge \text{NieZwei}_{99}$

Sudoku als Logik-Problem

Ähnlich:
(iii) In keiner Zeile, Spalte, Region kommt eine Zahl doppelt vor.

Schließlich **(iv) die Festlegung der Anfangszahlen**, im Beispiel also

$$A_{118} \wedge A_{152} \wedge A_{169} \wedge \cdots \wedge A_{957} \wedge A_{992}$$

8				2	9	5		7
	5						8	
9		4			5	2		
1		3						
4				3				8
						7		6
		5	6			8		3
	7						9	
3		1	4	7				2

Dann „einfach" in einen logischen Problemlöser eintippen ☹ und nach einem **Modell**, d.h. nach den restlichen gültigen A_{ijk}, fragen.
Die kommen dann schnell. ☺

Zum Glück geht es auch anders …

Ein gutes Sudoku hat **genau ein Modell**, d.h. genau eine Lösung.

Manche Denksportaufgaben haben null oder mehrere Lösungen oder sind sogar anderweitig problematisch.

Ü18

Logische Äquivalenz

Zwei Formeln φ und ψ heißen **semantisch** (oder **logisch**) **äquivalent**,

$\varphi \equiv \psi$ bzw. $\varphi \models\mid \psi$,

wenn sie unter allen (für beide ausreichenden) Belegungen den gleichen Wahrheitswert haben

$\Leftrightarrow$ φ und ψ haben in der Wahrheitstafel **identische Wahrheitswerteverläufe**

Äquivalenzsatz

Zwei Formeln sind genau dann semantisch äquivalent, wenn $\varphi \leftrightarrow \psi$ allgemeingültig ist:

$$\varphi \equiv \psi \;\Leftrightarrow\; \models \varphi \leftrightarrow \psi .$$

Begründung: nächste Folie

Tautologien mit $\leftrightarrow$ „oben“ ergeben daher Äquivalenzen.

Zwei (fast identische) Arten der Äquivalenzprüfung

$$\models \overset{\neg A \vee B}{\varphi} \leftrightarrow \overset{A \rightarrow B}{\psi}$$

A	B	$\neg A$	$\neg A \vee B$	$A \rightarrow B$	$(\neg A \vee B) \leftrightarrow (A \rightarrow B)$
W	W	F	W	W	**W**
W	F	F	F	F	**W**
F	W	W	W	W	**W**
F	F	W	W	W	**W**

$$\varphi \equiv \psi$$

A	B	$\neg A$	$\neg A \vee B$
W	W	F	**W**
W	F	F	**F**
F	W	W	**W**
F	F	W	**W**

A	B	$A \rightarrow B$
W	W	**W**
W	F	**F**
F	W	**W**
F	F	**W**

Der ⊤⊥-Dialekt (1)

Die Sprache der AL-Formeln **mit den Konstanten** *top* und *bottom* wird durch folgende Grammatik erzeugt:

- Nichtterminalsymbol-Menge $\{S\}$ und Startsymbol S wie gehabt.
- $T = \{ \top, \bot, A_1, A_2, \ldots, \neg, \wedge, \vee, \rightarrow, \leftrightarrow, (,) \}$ (Terminalsymbole erweitert)
- $R = \{S \rightarrow \top \mid \bot \mid A_1 \mid A_2 \mid \ldots,$ (Regeln: Verwendung wie Variablen)
  $S \rightarrow \neg S \mid (S \wedge S) \mid (S \vee S) \mid (S \rightarrow S) \mid (S \leftrightarrow S) \}$

Der **Wert** ist auf den Konstanten **unabhängig von der Belegung** festgelegt:

$wert_{bel}(\top) := \text{W}$

$wert_{bel}(\bot) := \text{F}$

⊤ und ⊥ kommen in der natürlichen Sprache nicht vor.

Punktueller Ersatz: „... oder ich fress' einen Besen“, für $\ldots \vee \bot$
„Ach Quatsch“, für $\ldots \wedge \bot$
„... weil es nun einmal ist wie es ist“, für $\ldots \wedge \top$

Der $\top\bot$-Dialekt (2)

Im $\top\bot$-Dialekt gelten zusätzliche **Tautologien**, z.B. ...

$\top$		Wahr
	$\neg\bot$	Numquam falsum
$(P \vee \neg P) \leftrightarrow \top$	$(P \wedge \neg P) \leftrightarrow \bot$	Tertium non datur
$(P \wedge \top) \leftrightarrow P$	$(P \wedge \bot) \leftrightarrow \bot$	Absorptionsgesetze
$(P \vee \top) \leftrightarrow \top$	$(P \vee \bot) \leftrightarrow P$	
$(\top \rightarrow P) \leftrightarrow P$		Ex vero nonnisi verum
	$\bot \rightarrow P$	Ex falso quodlibet

Assoziativität und Notation

Assoziativgesetze $\Rightarrow$

Ketten von (d.h. „benachbarte“) Konjunktionen sowie Ketten von Disjunktionen schreibt man meist **ohne** innere **Klammern**,
da **alle Klammerungen äquivalente** Terme ergeben.[1]

(*Doppeltes*) **Beispiel**: $\big((A \wedge (B \wedge C)) \vee D\big) \vee E \quad \Longrightarrow \quad (A \wedge B \wedge C) \vee D \vee E\,.$

Wo man eine *eindeutig* bestimmte Formel benötigt, genügt eine Standardinterpretation der „Sammelformel“, z.B. „von links nach rechts“:

$$\varphi_1 \wedge \varphi_2 \wedge \ldots \wedge \varphi_n \quad \Longrightarrow \quad ((\ldots((\varphi_1 \wedge \varphi_2) \wedge \varphi_2) \wedge \ldots \wedge \varphi_{-1}) \wedge \varphi_n)$$

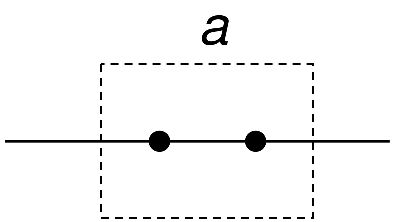
Andere **Schreibweise**: $\varphi_1 \wedge \varphi_2 \wedge \ldots \wedge \varphi_n \quad \Longleftrightarrow \quad \bigwedge_{k=1}^{n} \varphi_k$

(analog $\vee$ u. $\bigvee$)

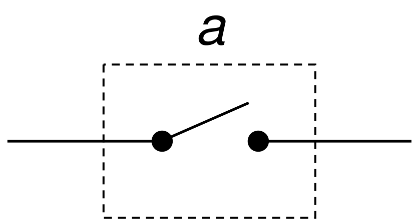
[1]) Entgegen Gerüchten ist der vollständige **Beweis nicht trivial.** ☹

Boolesche Schaltwerke (1)

Ein **Boolesches Schaltwerk** ist aus einer Anzahl von Elementen der Art **Arbeitskontakt** oder **Ruhekontakt** zusammengesetzt:



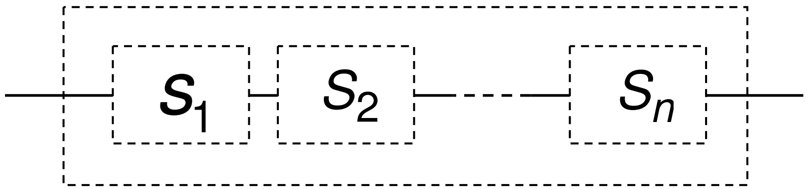
„Ruhe“kontakt $\neg a$

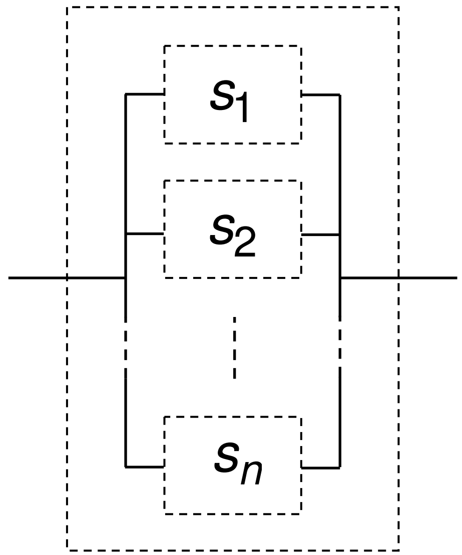


Arbeitskontakt a

(Schalter a steuert alle Kontakte namens a und steht in beiden Bildern auf AUS)

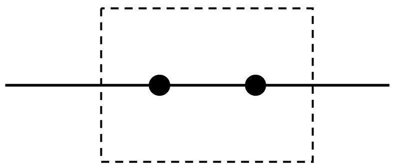
Die **Zusammensetzung** dieser Kontakte bzw. bereits zusammengesetzter Kontaktegruppen (aha: math. Induktion!) erfolgt **parallel** oder **seriell**:





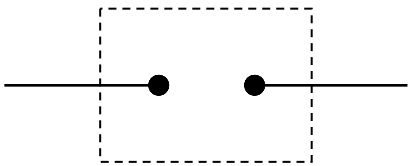
Boolesche Schaltwerke (2)

(T)



feste Verbindung

(⊥)



dauernde Unterbrechung

Heutige Schaltwerke enthalten natürlich auch komplexere Elemente (Gatter).

Vereinbarungen

- Verschiedene Kontakte können den gleichen Namen tragen. Sie haben dann stets die gleiche Stellung.
- a- und $\neg a$-Kontakte haben stets entgegengesetzte Stellung.
- Alle Kontakte mit Anschrift a oder $\neg a$ werden vom gleichen Schalter a gesteuert.
- Die Kontakte sind in Stellung <u>Schalter auf AUS</u> gezeichnet.

Formeldarstellung Boolescher Schaltwerke

1. Zuordnung *Z*: $\boxed{\text{Boolesche } \textbf{Schaltwerke}} \Rightarrow \boxed{\text{(gewisse) AL-}\textbf{Formeln}}$

Feste Verbindung	$\mapsto$	T
Dauernde Unterbrechung	$\mapsto$	$\perp$
Arbeitskontakt mit Schalter *a*	$\mapsto$	a
Ruhekontakt mit Schalter *a*	$\mapsto$	$\neg a$
Reihenschaltung von $S_1, ..., S_n$	$\mapsto$	$Z(S_1) \wedge ... \wedge Z(S_n)$
Parallelschaltung von $S_1, ..., S_n$	$\mapsto$	$Z(S_1) \vee ... \vee Z(S_n)$

2. Zuordnung: $\boxed{\textbf{Schalterstellungskombinationen}} \Rightarrow \boxed{\textbf{Belegungen}}$

Komb	$\mapsto$	Belegung bel_{Komb} mit
$bel_{Komb} : a \mapsto W$	$:\Leftrightarrow$	Schalter *a* unter *Komb* auf EIN.

> S verbindet leitend bei Schalterstellung *Komb* $\Leftrightarrow$
> $wert_{bel_{Komb}}(Z(S)) = W$

Ü19a Ü19b

Formeln Einsetzen – Substitution (1)

Substitutionssatz

Werden in einer Tautologie (bzw. in einer unerfüllbaren Formel)
für jede Aussagevariable A_i, i=1,2,...,
alle Vorkommen von A_i
jeweils durch die gleiche Formel ψ_i ersetzt,
so ist die entstehende Formel
ebenfalls eine Tautologie (bzw. unerfüllbar).

Substitution

Man schreibt meist nur die Ersetzungen mit $A_i \neq \psi_i$
und auch nur die für vorkommende A_i hin, also nur endlich viele.

Beispiel

$$A \vee (A \to B) \quad \xrightarrow[B \mapsto A \wedge B]{A \mapsto B} \quad B \vee (B \to (A \wedge B))$$

Wozu Substitutionssatz?

Beweisen Sie $\boxed{(A \wedge B \wedge C \wedge D) \to (A \wedge B \wedge C \wedge D)}$
a) mit Wahrheitstafel b) mit Substitutionssatz

Ü20

Formeln Einsetzen – Substitution (2)

Rekursive Definition der (**gleichzeitigen**) **Substitution** $\varphi_{[X_1, \ldots, X_n / \psi_1, \ldots, \psi_n]}$

von (paarweise verschiedenen Aussagevariablen) $P_1, \ldots, P_n$

durch (nicht unbedingt verschiedene) Formeln $\psi_1, \ldots, \psi_n$, :

Für $P \in AV$: $\quad P_{[P_1, \ldots, P_n / \psi_1, \ldots, \psi_n]} := \begin{cases} \psi_i & \text{für } P = P_i \\ P & \text{sonst} \end{cases}$,

Ferner $\quad (\neg\varphi)_{[P_1, \ldots, P_n / \psi_1, \ldots, \psi_n]} := \neg(\varphi_{[P_1, \ldots, P_n / \psi_1, \ldots, \psi_n]})$,

Für $\otimes \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$

ist $\quad (\varphi \otimes \rho)_{[P_1, \ldots, P_n / \psi_1, \ldots, \psi_n]} := \varphi_{[P_1, \ldots, P_n / \psi_1, \ldots, \psi_n]} \otimes \rho_{[P_1, \ldots, P_n / \psi_1, \ldots, \psi_n]}$.

Achtung: Reihenfolge-Effekte bei **nicht gleichzeitiger** Substitution:
Oft sind $\varphi_{[A,B / \psi, \rho]}$, $\varphi_{[A / \psi][B / \rho]}$ und $\varphi_{[B / \rho][A / \psi]}$
alle verschieden und nicht äquivalent.

Achtung: „Umkehrung“ des Substitutionssatzes falsch: Durch Substitution kann **Tautologie** evtl. erst **entstehen**.

Suchen Sie einfache Beispiele! **Ü21**

Ü22

Formeln Einsetzen – Ersetzung

Ersetzungssatz (Satz über die äquivalente Ersetzung)
Werden in einer Formel φ
ein oder mehrere Vorkommen einer Teilformel ψ
durch eine zu ψ äquivalente Formel ρ ersetzt,
so ist die entstehende Formel zu φ äquivalent.

Alle Möglichkeiten der 0-, ein- oder mehrfachen **Ersetzung**, **rekursiv**

$$
\begin{array}{lll}
\multicolumn{3}{l}{Ers_{[\psi/\rho]}(\varphi) := \{\varphi\} \cup} \\
\text{[if} & \varphi = \psi & \text{then } \{\rho\} \\
\text{else if} & \varphi = \neg\sigma & \text{then } \{\neg\tau \mid \tau \in Ers_{[\psi/\rho]}(\sigma)\} \\
\text{else if} & \varphi = \varphi_1 \otimes \varphi_2 & \text{then } \{\sigma \otimes \tau \mid \sigma \in Ers_{[\psi/\rho]}(\varphi_1), \tau \in Ers_{[\psi/\rho]}(\varphi_2)\} \\
\text{else } \varnothing]
\end{array}
$$

wann $\varnothing$? – wenn φ Aussagevariable $\neq \psi$

Anwendungsbeispiel des Ersetzungssatzes:

$$\underbrace{(B \to C)}_{\substack{||| \\ \neg C \to \neg B}} \vee \underbrace{A}_{\substack{||| \\ \neg\neg A}} \equiv (\neg C \to \neg B) \vee \neg\neg A, \quad \text{denn}$$

Ü23

Substitution vs. Ersetzung

Substitution **im Syntaxbaum**:
alle gleichnamigen Blätter durch je denselben Ast ersetzen.

Ersetzung **im Syntaxbaum**:
einen Ast (oder 0 oder mehrere gleiche Äste) durch (je) einen äquivalenten Ast ersetzen.

Transformation:	einfache Substitution	einfache Ersetzung
Was wird ersetzt?	Aussagevariable	Teilformel
Welche Vorkommen?	alle	0, 1, mehrere, alle
Wodurch?	beliebige Formel	äquivalente Formel
Ergebnis äquivalent?	JA, falls Tautologie oder Widerspruch	JA

Junktorenbasen (1)

Mit den Junktoren einer Junktorenbasis können wir
alle möglichen Spaltenbelegungen in Wahrheitstafeln erzeugen.

Genauer gesagt ist eine **Junktorenbasis *JuBa***

eine Junktorenmenge der Art, dass wir

zu jeder endlichen Menge *VarSet* von Aussagevariablen

jeden gewünschte Wahrheitswerteverlauf
(über alle Belegungen von *VarSet*)

mit einer geeigneten **Formel „über** *JuBa* **und** *VarSet*“ erzielen können.

Junktorenbasen (2)

Beispiel: $\boxed{\{\neg, \wedge, \vee\} \text{ ist Junktorenbasis}}$

Beweisidee:
Der gewünschten 2^n-zeiligen Wahrheitstafel entspricht eine **Disjunktion von** 2^n **Konjunktionen** aus je n **Literalen** (= Variable oder $\neg$Variable)

A	B	C	Spalte
⋮	⋮	⋮	⋮
F	**W**	**F**	**W**
⋮	⋮	⋮	⋮
F	**F**	**W**	**F**
⋮	⋮	⋮	⋮

F W F W $\Rightarrow$ $\boxed{\ldots \vee \ (\neg A \wedge B \wedge \neg C) \ \vee \ldots}$

F F W F $\Rightarrow$ $\boxed{\text{(nichts)}}$

Junktorenbasen (3)

Satz: **Von Junktorenbasen zu Junktorenbasen**

- Seien M und N Junktorenmengen und M eine Junktorenbasis. Genau dann ist auch N eine Junktorenbasis, wenn zu jedem, sagen wir n-stelligen, Junktor $j \in M$ eine zu $j(A_1, ..., A_n)$ äquivalente Formel über N und $\{A_1, ..., A_n\}$ existiert.
- Jede Junktoren-Obermenge einer Junktorenbasis ist selbst eine Junktorenbasis.

Anwendungen: $\{\neg, \wedge\}$ ist eine Junktorenbasis, denn $A \vee B \equiv \neg(\neg A \wedge \neg B)$.
$\{\neg, \vee\}$ ist eine Junktorenbasis, denn $A \wedge B \equiv \neg(\neg A \vee \neg B)$.
(mit Junktorenbasen-, Substitutions- und Ersetzungssatz)

Ü24a Ü24b

Junktorenbasen (4)

Es geht sogar noch sparsamer!

- Die Sheffer-Verknüpfung **alleine**, d.h. $\{\uparrow\}$, ist eine Junktorenbasis.
- Die Peirce-Verknüpfung **alleine**, d.h. $\{\downarrow\}$, ist eine Junktorenbasis.

Beweistip:
Suche Formeln über der jeweiligen Basis für $\neg\varphi$, $\varphi \wedge \psi$ und/oder $\varphi \vee \psi$.

- If-then-else alleine, d.h. $\{\rightarrow/\}$, ist eine Junktorenbasis **im** T$\perp$**-Dialekt**

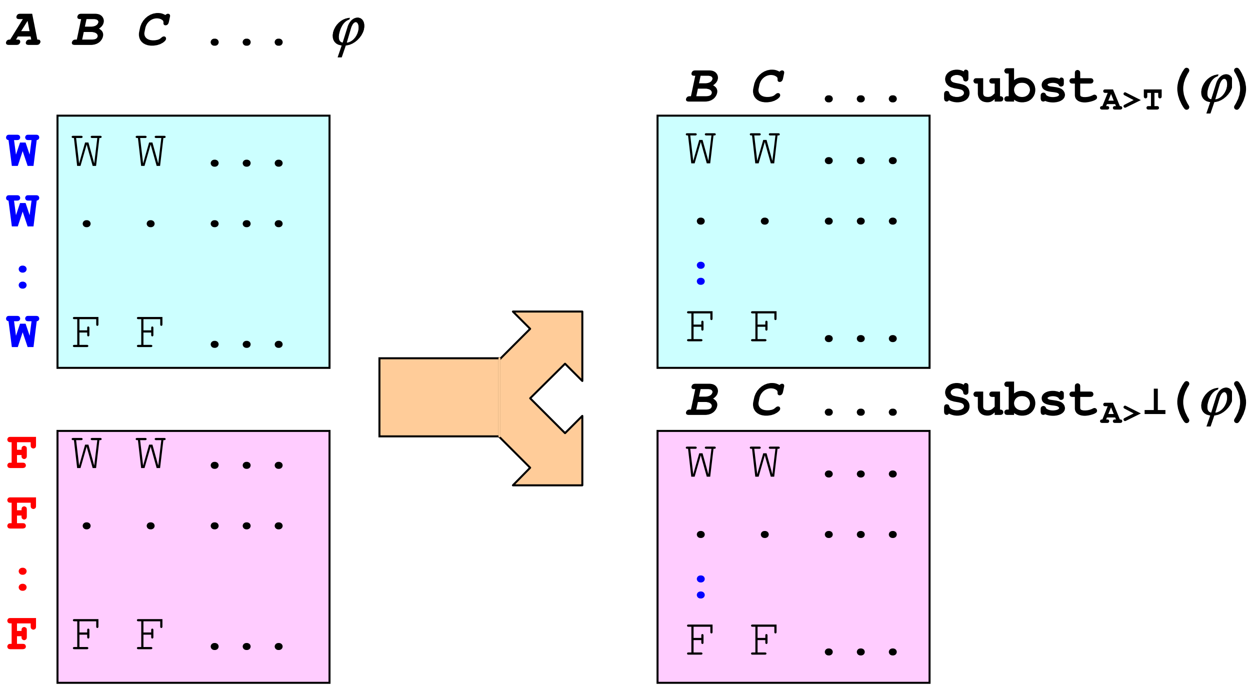
Beweistip:
analog oder ...

Shannon-Umformung (1)

Grundlage: Für jede AL-Formel φ und Aussagevariable z.B. A gilt:

$$\boxed{\varphi \equiv A \rightarrow \mathrm{Subst}_{A\rightarrow\mathrm{T}}(\varphi)\,/\,\mathrm{Subst}_{A\rightarrow\perp}(\varphi)}$$ **Shannon-Expansion**

Der Satz ist leicht einzusehen, wenn Sie sich die Wahrheitstafel in üblicher Schreibweise mit A in der ersten Spalte vorstellen.



Was wenn A nicht in φ vorkommt?

Shannon-Umformung (2)

2 Beispiele für Expansionsschritt

- $\neg A \equiv A \to \neg \mathrm{T} / \neg \bot$
- $A \vee B \equiv A \to \mathrm{T} \vee B / \bot \vee B$

Algorithmus ITE: **Äquivalente Umformung in ITE-Form**

Beginne mit der Ausgangsformel φ. Solange möglich:

- wähle eine noch nicht verarbeitete vorkommende Variable *P*,
- wende die Shannon-Expansion bzgl. *P* auf die zuletzt erhaltene Formel an
- und markiere *P* als verarbeitet.

Vereinfache schließlich die konstanten Teilterme zu T bzw. $\bot$.

Korrektheitssatz:

- ITE terminiert stets.
- Die durch Umformung erhaltene Formel ist in ITE-Form und zu φ äquivalent.

Vergleiche mit **Wahrheitstafel** in dieser Reihenfolge!

2 Beispiele für Vereinfachung

$\neg A \equiv A \to \neg \mathrm{T} / \neg \bot \equiv A \to \bot / \mathrm{T}$

$A \vee B \equiv A \to \mathrm{T} \vee B / \bot \vee B \equiv A \to \mathrm{T} / B$a

Ü25ab Ü25cd