Beispiel: Absorption aus den anderen Axiomen:

$$
\begin{aligned}
(a \sqcap b) \sqcup a &= (a \sqcap b) \sqcup (a \sqcap 1) \\
&= (a \sqcap b) \sqcup (a \sqcap (b \sqcup b^c)) \\
&= \underbrace{(a \sqcap b) \sqcup ((a \sqcap b)}_{} \sqcup (a \sqcap b^c)) \\
&= (a \sqcap b) \sqcup (a \sqcap b^c) \\
&= a \sqcap (b \sqcup b^c) \\
&= a \sqcap 1 \\
&= a
\end{aligned}
$$

$$(a \sqcap b) \sqcup a \overset{!}{=} a$$

alternative Def Boolescher Algebren:

partielle Ordnung mit Suprema, Infima, kleinstes & größtes Element, Komplement

Boolesche Algebren sind <u>Verbände</u> (Hasse-Diagramm)

Beispiel: M Menge

$$\mathrm{Pot}(M) = \{ X \mid X \subseteq M \} = \text{Menge der Teilmengen von } M$$

$(\mathrm{Pot}(M), \cap, \cup, \underset{X^C}{M \setminus \ldots}, M, \emptyset)$ ist Boole'sche Algebra

(„Potenzmengenalgebra von M")

partielle geordnet „$\subseteq$"

z.B. $M = \{a, b\}$

$$\begin{tikzcd}[column sep=small] & \{a,b\} \arrow[dl, dash] \arrow[dr, dash] & \\ \{a\} \arrow[dr, dash] & & \{b\} \arrow[dl, dash] \\ & \emptyset & \end{tikzcd}$$

Beispiel

$\mathcal{J}_0$

$$\begin{tikzcd} T/\sim \arrow[d, dash] \\ \bot/\sim \end{tikzcd}$$

$M = \emptyset$

$\mathrm{Pot}(\emptyset) = \{\emptyset\}$

$$\begin{tikzcd} \{\emptyset\} \arrow[d, dash] \\ \emptyset \end{tikzcd}$$

2 Elemente

Satz von Stone (ohne Beweis)

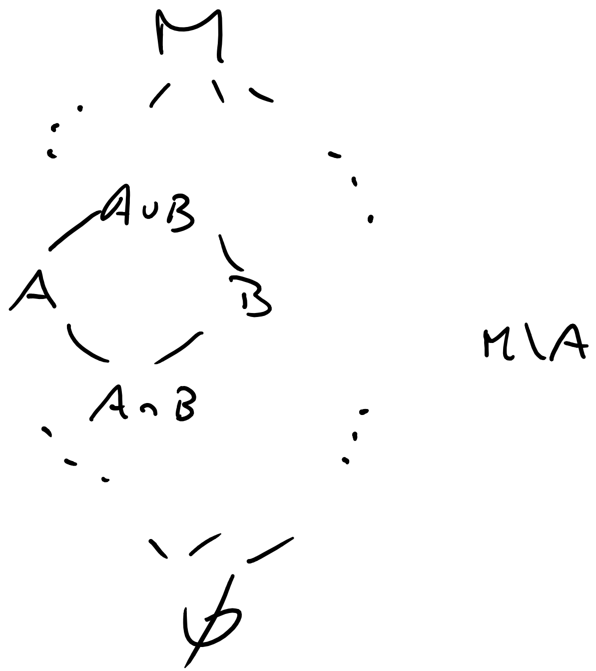
Jede Boole'sche Algebra ist Unteralgebra einer Potenzmengenalgebra

Bsp einer „echten" Unteralgebra:

$M = \mathbb{N}$

$B = \{ X \subseteq \mathbb{N} \mid X \text{ endlich oder } M \setminus X \text{ endlich} \}$

Pot(M)

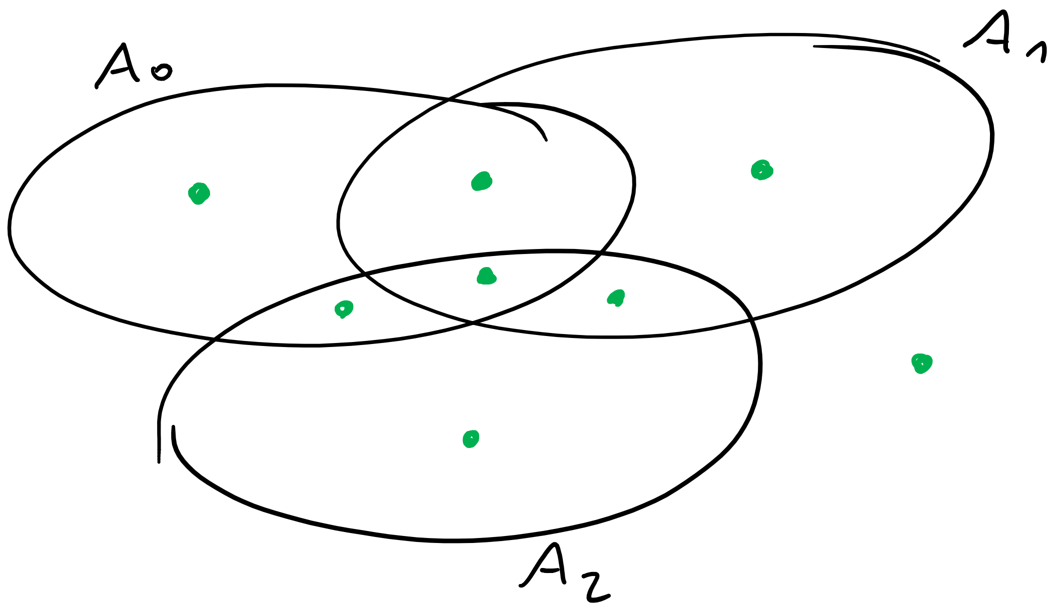


Beweisidee

$$B \hookrightarrow \mathrm{Pot}(\mathrm{Hom}(B, \{0,1\}))$$
$$b \longmapsto \{ h \mid h(b) = 1 \}$$

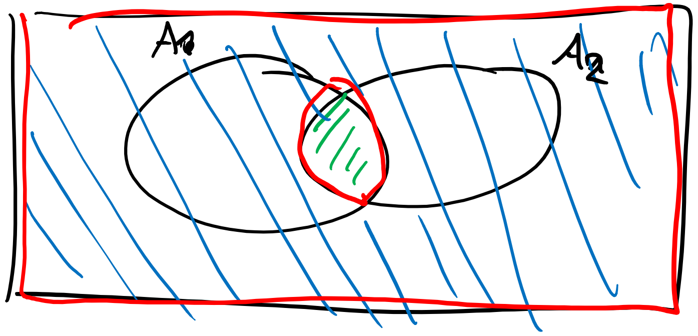
A_0, A_1, A_2

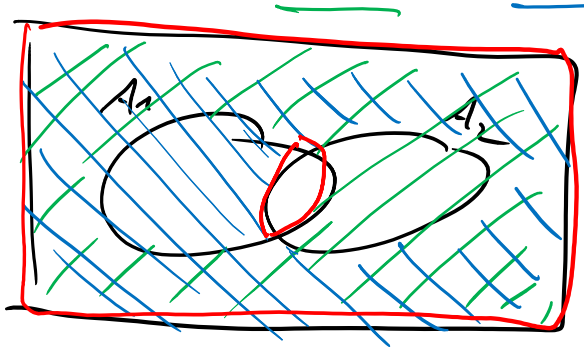
$\mathcal{F}_3 \hookrightarrow \text{Pot}$ (8 Belegungen)



Venn-Diagramm

de Morgan $(A_1 \cap A_2)^c = (A_1^c \cup A_2^c)$





$\sqcup \quad \cup \quad \vee$
$\sqcap \quad \cap \quad \wedge$

Dualität

Wenn $\mathcal{B} = (B, \sqcap, \sqcup, {}^c, 1, 0)$ Boole'sche Algebra,

dann ist $\mathcal{B}^* = (B, \sqcup, \sqcap, {}^c, 0, 1)$ auch Boole'sche Algebra

(die „duale Algebra")

Als partielle Ordnung $\mathcal{B} = (B, \leq)$, $\mathcal{B}^* = (B, \geq)$

$i: \mathcal{B} \longrightarrow \mathcal{B}^*$, $b \mapsto b^c$, ist Isomorphismus Boole'scher Algebren

(denn $(b \sqcap b')^c = b^c \sqcup b'^c$ (d.i. $i(b \sqcap b') = i(b) \sqcup i(b')$)

$\vdots$

$0^c = 1$ (d.i. $i(0) = 1$)

$b^{cc} = b^{cc}$ (d.i. $i(b)^c = i(b^c)$))

$\vdots$

das ist (≙ das heißt) id est i.e.

bijektiv: wegen $b^{cc} = b$ ist i.B. die Abb. surjektiv

F aus. Formel ohne $\rightarrow, \leftrightarrow$

F^* ist die Formel, die aus F entsteht, indem man

$\wedge$ und $\vee$

$\top$ und $\bot$ vertauscht.

Dann gilt: $F \sim G \Rightarrow F^* \sim G^*$

$F \vdash G \Rightarrow G^* \vdash F^*$

$\boxed{\text{Forum:}}$

Fakultät f. Mathematik & Physik

↳ Mathematik

↳ Lehrexport

2.4 Erfüllbarkeit

zwei algorithmische Probleme

gegeben: ausl. Formel F

(1) ist F erfüllbar?

(2) falls ja: finde erfüllende Belegung

Erinnerung: $\vdash F \iff \neg F$ nicht erfüllbar

$F \sim G \iff \neg(F \leftrightarrow G)$ nicht erfüllbar

(A) Wahrheitstafeln

$2^{\text{Anzahl der Aussagenvariablen}}$ viele Belegungen

$2^{100} \approx 10^{30}$ Jahr $\leq 10^{8}$ Sekunden

ExaFlops $\approx 10^{18}$ Operationen / Sekunde

Vorsicht: es ist unklar, ob es prinzipiell bessere Verfahren gibt ($P = NP$)

(B) Formel in DNF $\bigvee_i \bigwedge_j L_{ij}$

o.E. $\bigwedge_j L_{ij}$ „redundant", d.h. nicht A_l und $\neg A_l$ gleichzeitig in $\bigwedge$

redundantes $\bigwedge_j L_{ij}$ ist erfüllbar durch: $\beta(A_i) = 1$ falls A_i eines der Literale

$\beta(A_i) = 0$ falls $\neg A_i$ — " —

andere beliebig

„Formel in redundanter DNF" erfüllbar $\iff$ nicht $\perp$

Problem: Umformen einer Formel in DNF braucht i.a. exponentielle Zeit

(c) Formel in konjunktiver Normalform $\bigwedge_i \bigvee_j L_{ij}$

„Klausel"

$$(A_{i_1} \vee \dots \vee A_{i_l} \vee \neg A_{j_1} \vee \dots \vee \neg A_{j_k})$$

$$(\sim ((A_{j_1} \wedge \dots \wedge A_{j_k}) \to (A_{i_1} \vee \dots \vee A_{i_l})))$$

Klausel = Disjunktion von Literalen

wird gerne mit der Menge der Literale $\{A_{i_1}, \dots, A_{i_l}, \neg A_{j_1}, \dots \neg A_{j_k}\}$ identifiziert

$$(A_0 \vee \neg A_1 \vee A_0) \rightsquigarrow \{A_0, \neg A_1\}$$

Belegung β erfüllt eine Klausel, falls sie (die zugehörigen Formeln) wahr macht

β eine Klauselmenge, wenn jede Klausel in der Menge erfüllt wird gegeben Klauseln $C_1 = \{L_1, \ldots, L_k, A_i\}$

$C_2 = \{L_1', \ldots, L_m', \neg A_i\}$

$R = \{L_1, \ldots, L_k, L_1', \ldots, L_m'\}$ ist eine <u>Resolvente</u> von C_1 und C_2

R entsteht durch Resolution aus C_1 und C_2

<u>Satz</u>: Eine Klauselmenge ist genau dann erfüllbar, wenn sich die leere Klausel $\emptyset$ nicht durch sukzessive Resolution ergibt.

entspricht $\perp$

$$\left.\begin{aligned} C_1 &= \{A_1\} \\ C_2 &= \{\neg A_1\} \end{aligned}\right\} \text{Resolution} \quad R = \{\}$$

$C_1 = \{A_1, \neg A_2, A_3\}$

$C_2 = \{\neg A_1, A_2, \neg A_3\}$