

# A Wahrscheinlichkeit und Information

## A.1 Die Wahrscheinlichkeit von Formeln

Üblicherweise führt man den Wahrscheinlichkeitsbegriff mit Hilfe von Wahrscheinlichkeitsmaßen über Wahrscheinlichkeitsräumen – im Wesentlichen also über Mengenalgebren – ein. Wir werden hier nur spezielle Mengenalgebren betrachten, nämlich Potenzmengen (= Mengen aller Teilmengen) einer Menge  $\Omega$ . Diese sind trivialerweise bzgl. Vereinigung, Durchschnitt und Komplementbildung abgeschlossen, so dass die Frage der *Messbarkeit* einer Menge (formal lediglich ihre Zugehörigkeit zur betrachteten Mengenalgebra) nicht ausdrücklich diskutiert werden muss: Alle Teilmengen von  $\Omega$  sind Elemente der Potenzmenge  $2^\Omega$  und daher messbar im wahrscheinlichkeitstheoretischen Sinne.

Überdies werden wir voraussetzen, dass die betrachtete Menge  $\Omega$  endlich ist, so dass wir insgesamt hier nur einen kleinen, für Anwendungen aber durchaus ausreichenden Ausschnitt der Wahrscheinlichkeitstheorie präsentieren.

**Definition A.1 (Wahrscheinlichkeitsfunktion)** Eine *Wahrscheinlichkeitsfunktion* über einer (endlichen) Menge  $\Omega$  ist eine Abbildung

$$P : 2^\Omega \rightarrow [0, 1]$$

die den folgenden Eigenschaften genügt:

(P1)  $P(\Omega) = 1$

(P2) Sind  $M_1$  und  $M_2$  disjunkte Mengen, so gilt für ihre Vereinigung

$$P(M_1 \cup M_2) = P(M_1) + P(M_2), \quad \text{wenn } M_1 \cap M_2 = \emptyset$$

□

Die obige Eigenschaft (P2) verlangt nur die endliche Additivität einer Wahrscheinlichkeitsfunktion. Dies ist sicherlich im Hinblick auf die vorausgesetzte Endlichkeit von  $\Omega$  ausreichend. Für Wahrscheinlichkeitsmaße im Allgemeinen wird die sog.  $\sigma$ -*Additivität* gefordert, d.h. die Additivität über abzählbar unendliche Vereinigungen. Für den Bereich der Wissensrepräsentation benötigt man im Allgemeinen nur die endliche Additivität, und deswegen wollen wir es hier dabei belassen.

**Definition A.2 (Elementarereignis, Ereignis)** Sei  $\Omega$  eine Menge, auf der eine Wahrscheinlichkeitsfunktion gegeben ist.

Ein Element  $\omega \in \Omega$  wird als *Elementarereignis* bezeichnet, eine Teilmenge  $M \subseteq \Omega$  als *Ereignis*. □

Eine wichtige und oft benutzte Eigenschaft von Wahrscheinlichkeitsfunktionen ist die, dass jede Wahrscheinlichkeitsfunktion eindeutig bestimmt ist durch ihre Werte auf den Elementarereignissen  $\omega \in \Omega$ . Alle diese Elementarereignisse sind paarweise disjunkt, d.h. es ist  $\{\omega\} \cap \{\omega'\} = \emptyset$  für  $\omega \neq \omega'$ . Also folgt mit der obigen Eigenschaft (P2) für jede Teilmenge  $M$  von  $\Omega$ :

$$P(M) = \sum_{\omega \in M} P(\omega) \quad (\text{A.1})$$

wobei der Einfachheit halber  $P(\omega) := P(\{\omega\})$  für  $\omega \in \Omega$  gesetzt wird.

In der Wissensrepräsentation betrachtet man allerdings im Allgemeinen keine Mengen, sondern Aussagen, Eigenschaften, Merkmale, Prädikate o.Ä. Mit Hilfe des Wahrscheinlichkeitsbegriffes werden wir im Folgenden eine *probabilistische Logik* definieren, bei der nicht Mengen, sondern aussagenlogischen Formeln Wahrscheinlichkeiten zugewiesen werden.

Wir betrachten eine aussagenlogische Sprache  $\mathcal{L}$  über einer Signatur  $\Sigma = \{N_1, N_2, N_3, \dots\}$ . Jeder Name  $N_i$  von  $\Sigma$  repräsentiert eine Aussage, die wahr oder falsch sein kann. Es bezeichne *Form* die Menge aller aussagenlogischen Formeln von  $\mathcal{L}$ . Jede Formel stellt wieder eine Aussage dar, nämlich entweder eine *atomare Formel* (also eine Aussage aus  $\Sigma$ ) oder eine zusammengesetzte Formel; ein *Literal* ist eine atomare Formel entweder in positiver oder in negierter Form (vgl. Definition 3.22). Mit  $\top$  werde eine beliebige tautologische Formel bezeichnet (z.B.  $A \vee \neg A$ ) und mit  $\perp$  eine beliebige widersprüchliche Formel (z.B.  $A \wedge \neg A$ ).

*Vollkonjunktionen* sind Konjunktionen, die jeden Buchstaben des Alphabets von  $\mathcal{L}$  in positiver oder negierter Form enthalten. Sie dienen der vollständigen Klassifizierung von Objekten, stellen also Elementarereignisse im wahrscheinlichkeitstheoretischen Sinne dar.

Im Folgenden werde mit  $\Omega$  immer die Menge aller Vollkonjunktionen einer logischen Sprache  $\mathcal{L}$  bezeichnet. Eine Wahrscheinlichkeitsfunktion  $P$  über  $\Omega$  nennen wir auch *Wahrscheinlichkeitsfunktion über  $\mathcal{L}$* .

**Beispiel A.3 (Vollkonjunktionen 1)** Die Signatur einer Sprache bestehe aus den Buchstaben  $\{R, W, G\}$ , die beispielsweise die folgenden Bedeutungen haben können:

$$\begin{array}{ll} R & : \text{ rund} \\ W & : \text{ weiß} \\ G & : \text{ groß} \end{array}$$

und damit zur Klassifikation von Gegenständen benutzt werden können. Es gibt hier  $2^3 = 8$  Vollkonjunktionen:

$$\begin{array}{ll} R \wedge W \wedge G & \neg R \wedge W \wedge G \\ R \wedge W \wedge \neg G & \neg R \wedge W \wedge \neg G \\ R \wedge \neg W \wedge G & \neg R \wedge \neg W \wedge G \\ R \wedge \neg W \wedge \neg G & \neg R \wedge \neg W \wedge \neg G \end{array} \quad \square$$

Umfasst das Vokabular der Sprache  $n$  (zweiwertige) Aussagen, so gibt es  $2^n$  Vollkonjunktionen.

Die kanonische disjunktive Normalform einer Formel  $A$  besteht aus genau denjenigen Vollkonjunktionen  $\omega$ , die Implikanten von  $A$  sind, d.h. für die die Formel  $\omega \Rightarrow A$  allgemeingültig ist; man schreibt dafür auch  $A(\omega) = 1$ :

$$A \equiv \bigvee_{\omega:A(\omega)=1} \omega \quad (\text{A.2})$$

Die Schreibweise

$$A(\omega) = 1 \quad \text{für} \quad \models \omega \Rightarrow A$$

entstammt dem Ansatz, Formeln als sog. logische Funktionen und Vollkonjunktionen als Tupel atomarer Wahrheitswerte zu betrachten.

**Beispiel A.4 (Vollkonjunktionen 2)** Wir setzen das Beispiel A.3 fort. Die Formel  $R$  hat hier die kanonische disjunktive Normalform

$$R \equiv (R \wedge W \wedge G) \vee (R \wedge W \wedge \neg G) \vee (R \wedge \neg W \wedge G) \vee (R \wedge \neg W \wedge \neg G) \quad \square$$

Eine Formel  $A$  repräsentiert also eine Menge von Vollkonjunktionen

$$\Omega_A := \{\omega \in \Omega \mid A(\omega) = 1\} \quad (\text{A.3})$$

und damit ein wahrscheinlichkeitstheoretisches Ereignis, und wir setzen

$$P(A) := P(\Omega_A)$$

Die Mengenoperationen Vereinigung und Schnitt entsprechen nun den logischen Operationen Disjunktion und Konjunktion, und aus der Komplementbildung wird die Negation:

$$\begin{aligned} A \vee B & \text{ entspricht } \Omega_A \cup \Omega_B \\ A \wedge B & \text{ entspricht } \Omega_A \cap \Omega_B \\ \neg A & \text{ entspricht } \Omega - \Omega_A \end{aligned}$$

Aus der obigen Formel (A.2) erhalten wir also das aussagenlogische Gegenstück zu Gleichung (A.1):

$$P(A) = \sum_{\omega:A(\omega)=1} P(\omega) \quad (\text{A.4})$$

Ausgehend von einer Wahrscheinlichkeitsfunktion  $P$  auf  $\Omega$  kann auf diese Weise jeder Formel  $A \in \text{Form}$  eine Wahrscheinlichkeit  $P(A)$  zugeordnet werden.  $P(A)$  symbolisiert den Grad der Gewissheit, mit der  $A$  – in einer realen oder fiktiven Population – wahr ist.

Die Wahrscheinlichkeit einer Formel wird folglich eindeutig bestimmt durch die Wahrscheinlichkeit ihrer zugehörigen Vollkonjunktionen. Manchmal werden Wahrscheinlichkeitsfunktionen daher auch in der Form

$$\{P(\omega) \mid \omega \in \Omega\} \quad (\text{A.5})$$

angegeben, wobei jedes  $P(\omega) \in [0, 1]$  und  $\sum_{\omega \in \Omega} P(\omega) = 1$  ist. Umgekehrt definiert jede solche Zuweisung mittels der Gleichung (A.4) eine Wahrscheinlichkeitsfunktion. Da in (A.5) angegeben wird, wie sich die Wahrscheinlichkeitsmasse von 1 auf die Vollkonjunktionen verteilt, spricht man auch von einer *Wahrscheinlichkeitsverteilung* (*probability distribution*).

**Definition A.5 (Gleichverteilung)** Eine Wahrscheinlichkeitsverteilung, die jeder Vollkonjunktion  $\omega$  die gleiche Wahrscheinlichkeit  $P(\omega)$  zuordnet, heißt *Gleichverteilung* und wird mit  $P_0$  bezeichnet. Wegen  $P(\Omega) = 1$  und der Gleichung (A.1) gilt  $P_0(\omega) = \frac{1}{n}$  für  $|\Omega| = n$ .  $\square$

**Definition A.6 (Exklusivität)** Zwei Formeln  $A, B \in \text{Form}$  heißen *exklusiv* (*exclusive*), wenn ihre Konjunktion widersprüchlich ist, wenn also  $A \wedge B \equiv \perp$  ist.  $\square$

Je zwei verschiedene Vollkonjunktionen sind exklusiv. Allgemein lassen sich viele interessante Eigenschaften und Beziehungen von Formeln durch die mit ihnen assoziierten Mengen von Vollkonjunktionen beschreiben:

**Proposition A.7** Zu einer Formel  $A \in \text{Form}$  sei die Menge  $\Omega_A$  wie in der Formel (A.3) definiert.

1. Zwei Formeln  $A, B \in \text{Form}$  sind genau dann exklusiv, wenn die mit ihnen assoziierten Mengen von Vollkonjunktionen disjunkt sind, d.h.

$$A, B \text{ exklusiv} \quad \text{gdw.} \quad \Omega_A \cap \Omega_B = \emptyset$$

2. Eine Formel  $A$  ist genau dann tautologisch, wenn  $\Omega_A = \Omega$  ist.
3. Zwei Formeln  $A, B \in \text{Form}$  sind genau dann logisch äquivalent, wenn die mit ihnen assoziierten Mengen von Vollkonjunktionen gleich sind, d.h.

$$A \equiv B \quad \text{gdw.} \quad \Omega_A = \Omega_B$$

4. Eine Formel  $A$  impliziert genau dann logisch eine Formel  $B$ , wenn  $\Omega_A \subseteq \Omega_B$  gilt:

$$\models A \Rightarrow B \quad \text{gdw.} \quad \Omega_A \subseteq \Omega_B$$

Damit lässt sich eine Wahrscheinlichkeitsfunktion  $P$  über  $\mathcal{L}$  wie folgt charakterisieren:

**Proposition A.8** Eine Abbildung  $P : \text{Form} \rightarrow [0, 1]$  ist genau dann eine Wahrscheinlichkeitsfunktion über  $\mathcal{L}$ , wenn gilt:

(P0) Sind  $A$  und  $B$  logisch äquivalente Formeln,  $A \equiv B$ , so gilt  $P(A) = P(B)$ .

(P1)  $P(\top) = 1$  für jede tautologische Formel  $\top$ .

(P2) Sind  $A, B \in \text{Form}$  zwei exklusive Formeln, so gilt für ihre Disjunktion

$$P(A \vee B) = P(A) + P(B)$$

Aus diesem Satz lassen sich einige wichtige und grundlegende Eigenschaften von Wahrscheinlichkeitsfunktionen ableiten:

**Proposition A.9** Sei  $P$  eine Wahrscheinlichkeitsfunktion über  $\mathcal{L}$ , seien  $A, B \in \text{Form}$ .

1.  $P(A) = P(A \wedge B) + P(A \wedge \neg B)$ ;
2.  $P(\neg A) = 1 - P(A)$ ;
3.  $P(\perp) = 0$  für jede widersprüchliche Formel  $\perp$ ;
4.  $\models A \Rightarrow B$  impliziert  $P(A) \leq P(B)$ ;
5.  $P(A \vee B) = P(A) + P(B) - P(A \wedge B)$ .

Für die Eigenschaft (P2) in Proposition A.8 ist wichtig, dass die durch  $A$  und  $B$  repräsentierten Aussagen sich gegenseitig ausschließen. Demgegenüber gilt Proposition A.9 (5) für beliebige Formeln  $A$  und  $B$ , schließt aber (P2) mit ein: Sind  $A$  und  $B$  exklusiv, so ist  $A \wedge B \equiv \perp$ , also  $P(A \wedge B) = 0$  wegen Proposition A.9 (3), und man erhält (P2).

Die Aussage in Proposition A.9 (4) besagt, dass die Wahrscheinlichkeit einer Formel  $B$  mindestens so groß ist wie die einer Formel  $A$ , wenn  $B$  logisch aus  $A$  folgt. Dem entspricht in der mengentheoretischen Formulierung gerade die Monotonie einer Wahrscheinlichkeitsfunktion (vgl. Proposition A.7(4)).

**Selbsttestaufgabe A.10 (Wahrscheinlichkeitsfunktion)** Beweisen Sie die Aussagen der Proposition A.9. ■

Statistische Häufigkeitsverteilungen sind spezielle, in der Praxis aber oft auftretende Wahrscheinlichkeitsverteilungen. Aus ihnen erhält man daher leicht Wahrscheinlichkeitsfunktionen, wie das folgende Beispiel zeigt.

**Beispiel A.11 (Wahrscheinlichkeitsverteilung)** Wir nehmen an, der Zusammenhang zwischen einer Krankheit  $D$  (= *Diagnose*) und zwei Symptomen  $S_1, S_2$  sei bei hundert Patienten statistisch erfasst und in der folgenden Häufigkeitsverteilung festgehalten worden:

$D$	$S_1$	$S_2$	abs. Häufigkeit	rel. Häufigkeit
0	0	0	19	0.19
0	0	1	8	0.08
0	1	0	11	0.11
0	1	1	2	0.02
1	0	0	15	0.15
1	0	1	14	0.14
1	1	0	20	0.20
1	1	1	11	0.11

Dabei liest sich z.B. die dritte Zeile wie folgt: Bei 11 Patienten (also 11 %) wurde zwar Symptom  $S_1$ , nicht aber Symptom  $S_2$  festgestellt; die Diagnose war negativ. Jeder Zeile entspricht eine Vollkonjunktion (der dritten Zeile also die Vollkonjunktion  $\neg D \wedge S_1 \wedge \neg S_2$ ), dessen Wahrscheinlichkeit durch die zugehörige relative Häufigkeit gegeben wird:

$$P(\neg D \wedge S_1 \wedge \neg S_2) = 0.11$$

Die letzte Spalte der obigen Tabelle zeigt die Wahrscheinlichkeiten aller Vollkonjunktionen an. Damit lassen sich nun alle anderen Wahrscheinlichkeiten berechnen; wir erhalten z. B.

$$\begin{aligned} P(S_1) &= 0.11 + 0.02 + 0.20 + 0.11 = 0.44 \\ P(S_2) &= 0.08 + 0.02 + 0.14 + 0.11 = 0.35 \\ P(D) &= 0.15 + 0.14 + 0.20 + 0.11 = 0.60 \\ P(D \wedge S_1) &= 0.20 + 0.11 = 0.31 \end{aligned} \quad \square$$

## A.2 Randverteilungen

Durch das Aufsummieren elementarer Wahrscheinlichkeiten lassen sich auch Verteilungen über Teilmengen der betrachteten Aussagen aus einer gegebenen Verteilung ableiten. Das ist immer dann sinnvoll, wenn man die Zusammenhänge nur zwischen bestimmten, ausgewählten Aussagen untersuchen will, gewisse Aussagen also gezielt ausblenden möchte.

**Definition A.12 (Randverteilung)** Sei  $\Sigma' \subseteq \Sigma$  eine Teilmenge der Signatur der Sprache  $\mathcal{L}$ , und sei  $\mathcal{L}'$  die durch  $\Sigma'$  definierte Sprache mit Menge  $\Omega'$  der Vollkonjunktionen und Formelmenge  $Form'$ . Sei  $P$  eine Wahrscheinlichkeitsverteilung über  $\Sigma$ . Jede Vollkonjunktion  $\omega' \in \Omega'$  lässt sich sowohl als eine Formel über  $\Sigma$  als auch als eine Formel über  $\Sigma'$  auffassen. Durch

$$P'(\omega') := P(\omega') = \sum_{\omega: \omega'(\omega)=1} P(\omega)$$

wird eine Wahrscheinlichkeitsverteilung über  $\Sigma'$  definiert.  $P'$  wird *Randverteilung* oder *marginale Verteilung* (*marginal distribution*) von  $P$  (über  $\Sigma'$ ) genannt, und den Vorgang ihrer Berechnung nennt man *Marginalisieren*.  $\square$

Die Randverteilung  $P'$  stimmt also in allen Formeln aus  $Form \cap Form'$  mit  $P$  überein.

**Beispiel A.13 (Marginalisieren, Randverteilung)** Wir setzen das obige Beispiel A.11 fort und berechnen eine Randverteilung  $P'$  über  $D$  und  $S_1$ . Dazu müssen entsprechende Wahrscheinlichkeiten aus Beispiel A.11 aufsummiert werden. Für die Vollkonjunktion  $D \wedge S_1$  (über der Menge  $\{D, S_1\}$ !) errechneten wir bereits oben

$$P'(D \wedge S_1) = P(D \wedge S_1) = 0.31$$

Die folgende Tabelle enthält die komplette Randverteilung  $P'$ .

$D$	$S_1$	$P'$
0	0	0.27
0	1	0.13
1	0	0.29
1	1	0.31

$\square$

**Selbsttestaufgabe A.14 (Marginalisieren, Randverteilung)** Berechnen Sie zu der in Beispiel A.11 gegebenen Verteilung die Randverteilung  $P''$  über  $D$  und  $S_2$ . ■

### A.3 Bedingte Wahrscheinlichkeiten

Ein wichtiges Konzept für Wahrscheinlichkeitstheorie und Wissensrepräsentation gleichermaßen ist die *bedingte Wahrscheinlichkeit*:

**Definition A.15 (bedingte Wahrscheinlichkeit)** Sei  $P$  eine Wahrscheinlichkeitsfunktion auf  $\mathcal{L}$ , und seien  $A, B \in \text{Form}$  zwei Formeln, wobei  $P(A)$  als positiv vorausgesetzt wird:  $P(A) > 0$ . Die *bedingte Wahrscheinlichkeit von B gegeben A* wird definiert durch

$$P(B | A) = \frac{P(A \wedge B)}{P(A)}$$

□

$P(B | A)$  gibt die Wahrscheinlichkeit an, dass, wenn  $A$  erfüllt ist, auch  $B$  wahr ist.

**Beispiel A.16 (bedingte Wahrscheinlichkeit)** Für Beispiel A.11 berechnen wir bedingte Wahrscheinlichkeiten:

$$P(D | S_1) = \frac{P(D \wedge S_1)}{P(S_1)} = \frac{0.31}{0.44} = 0.705$$

$$P(D | S_2) = \frac{P(D \wedge S_2)}{P(S_2)} = \frac{0.14 + 0.11}{0.35} = 0.714$$

Die Wahrscheinlichkeit, dass die Krankheit  $D$  vorliegt, wenn Symptom  $S_1$  auftritt, beträgt also ungefähr 70 %. Die entsprechende Wahrscheinlichkeit für Symptom  $S_2$  ist nur unwesentlich höher, nämlich ca. 71 %. □

**Selbsttestaufgabe A.17 (bedingte Wahrscheinlichkeit)** Berechnen Sie in Beispiel A.11 die bedingten Wahrscheinlichkeiten  $P(S_1 | D)$  und  $P(S_2 | D)$  und interpretieren Sie diese. ■

**Proposition A.18** Sei  $P$  eine Wahrscheinlichkeitsfunktion auf  $\mathcal{L}$ , sei  $A \in \text{Form}$  mit  $P(A) > 0$ . Dann erfüllt die Funktion

$$P_A : \text{Form} \rightarrow [0, 1], \quad P_A(B) := P(B | A)$$

die Eigenschaften (P0), (P1), (P2) der Proposition A.8.  $P_A$  ist daher eine Wahrscheinlichkeitsfunktion auf  $\mathcal{L}$ .

**Selbsttestaufgabe A.19 (bedingte Wahrscheinlichkeitsfunktion)**

Berechnen Sie in Beispiel A.11 die bedingte Wahrscheinlichkeitsfunktion  $P_{S_2}$ . ■

Es gibt also zwei Möglichkeiten, um aus einer Wahrscheinlichkeitsfunktion  $P$  neue Wahrscheinlichkeitsfunktionen zu gewinnen: die Marginalisierung und die *Konditionalisierung*, also die Bildung einer bedingten Wahrscheinlichkeitsfunktion. Von beiden wird sehr häufig Gebrauch gemacht. Obwohl in beiden Methoden Aussagen scheinbar eliminiert werden, dürfen sie nicht verwechselt werden: Wenn über eine Aussagenvariable  $A$  marginalisiert wird, so bedeutet dies, dass sie für die folgenden Betrachtungen keine Rolle spielt. So wurde in Beispiel A.13 beim Übergang zu  $P'$  über die Variable  $S_2$  marginalisiert;  $P'$  lässt nur noch die Untersuchung der Beziehungen zwischen  $D$  und  $S_1$  zu. Eine Konditionalisierung nach der Variablen  $A$  hingegen konzentriert die Betrachtungen auf die Fälle, in denen  $A$  wahr ist und lässt alle anderen außer Acht. Die Wahrscheinlichkeitsfunktion  $P_{S_2}$  in Aufgabe A.19 gibt die Beziehungen zwischen  $D$  und  $S_1$  unter der Annahme, dass  $S_2$  wahr ist, wieder.

Mit Hilfe des Konzeptes der bedingten Wahrscheinlichkeit lassen sich einige wichtige Resultate ableiten:

**Theorem A.20 (Satz von der totalen Wahrscheinlichkeit)** *Es seien  $A, B_1, \dots, B_n$  Formeln. Es sei weiterhin vorausgesetzt, dass  $B_1, \dots, B_n$  paarweise exklusiv und überdies ausschöpfend sind, d.h., es gilt  $B_i \wedge B_j \equiv \perp$  für  $i \neq j$  und  $B_1 \vee \dots \vee B_n \equiv \top$ . Außerdem sei  $P(B_i) > 0$  für alle  $i \in \{1, \dots, n\}$ . Dann gilt*

$$P(A) = \sum_{i=1}^n P(A | B_i) \cdot P(B_i) \quad (\text{A.6})$$

**Beweis:** Es ist

$$\begin{aligned} \sum_{i=1}^n P(A|B_i) \cdot P(B_i) &= \sum_{i=1}^n \frac{P(A \wedge B_i)}{P(B_i)} \cdot P(B_i) = \sum_{i=1}^n P(A \wedge B_i) = \\ P\left(\bigvee_{i=1}^n (A \wedge B_i)\right) &= P\left(A \wedge \left(\bigvee_{i=1}^n B_i\right)\right) = P(A \wedge \top) = P(A) \quad \blacksquare \end{aligned}$$

Häufig wird der Satz von der totalen Wahrscheinlichkeit für den Fall  $n = 2$  angewendet. Eine Aussage  $B$  und ihre Negation  $\neg B$  sind exklusiv und ausschöpfend, und Formel (A.6) lautet in diesem Fall

$$P(A) = P(A | B) \cdot P(B) + P(A | \neg B) \cdot P(\neg B)$$

Beachten Sie, dass in der obigen Formel die beiden bedingten Wahrscheinlichkeiten noch mit den absoluten Wahrscheinlichkeiten von  $B$  bzw.  $\neg B$  multipliziert werden müssen, um eine korrekte wahrscheinlichkeitstheoretische Beziehung zu liefern.

Die nächste Formel wird oft auch als *Kettenregel* bezeichnet:

**Proposition A.21 (Kettenregel)** *Sei  $P$  eine Wahrscheinlichkeitsfunktion über  $\mathcal{L}$ , und es seien  $A_1, \dots, A_n$  Formeln, so dass  $P(\bigwedge_{i=1}^{n-1} A_i) > 0$  ist. Dann gilt:*

$$P\left(\bigwedge_{i=1}^n A_i\right) = P(A_1) \cdot P(A_2|A_1) \cdot P(A_3|A_1 \wedge A_2) \cdot \dots \cdot P(A_n | \bigwedge_{i=1}^{n-1} A_i)$$

**Beweisidee:** Der Beweis ist eine einfache Multiplikationsaufgabe.  $\blacksquare$



## A.4 Der Satz von Bayes

Bedingte Wahrscheinlichkeiten spielen für wissensbasierte Systeme eine besonders wichtige Rolle, da sie es erlauben, einer Regel in konsistenter Weise ein Maß der (Un)Sicherheit zuzuordnen. Regeln in Wenn-dann-Form gehören zu den elementarsten, gleichzeitig aber auch wichtigsten und populärsten Instrumenten, um Wissen zu formalisieren. Als sichere “Produktionsregeln” werden sie in deterministischen Systemen verwendet. Doch erst die Möglichkeit, mit Unsicherheit behaftete Regeln zu formulieren und zu verarbeiten, erlaubt eine realistische Nähe zum menschlichen Schließen.

Betrachten wir hierzu die folgende Problemstellung aus dem Bereich der medizinischen Diagnose: Hier geht es grundsätzlich darum, aus einem Symptom  $S$  auf das Vorhandensein einer Krankheit (Diagnose)  $D$  zu schließen. Gesucht ist also die bedingte Wahrscheinlichkeit  $P(D | S)$ . Ist sie “groß” genug, wird man die entsprechende Krankheit als mögliche Diagnose in Betracht ziehen. Für einen medizinischen Experten ist es jedoch oft sehr viel leichter, die “umgekehrte” bedingte Wahrscheinlichkeit  $P(S | D)$  anzugeben, also die Wahrscheinlichkeit, dass ein Patient mit der Krankheit  $D$  das Symptom  $S$  zeigt.

Die folgende, unter dem Namen “Bayessche Regel” bekannte und berühmte Formel erlaubt es, die eine bedingte Wahrscheinlichkeit aus der anderen zu berechnen:

**Theorem A.22 (Satz von Bayes)** *Sei  $P$  eine Wahrscheinlichkeitsfunktion über  $\mathcal{L}$ , und seien  $A, B$  zwei Formeln mit  $P(A), P(B) > 0$ . Dann gilt*

$$P(B | A) = \frac{P(A | B)P(B)}{P(A)} \quad (\text{A.7})$$

**Beweis:** Es ist

$$\frac{P(A | B)P(B)}{P(A)} = \frac{\frac{P(A \wedge B)}{P(B)}P(B)}{P(A)} = \frac{P(A \wedge B)}{P(A)} = P(B | A) \quad \blacksquare$$

Der Beweis dieser Regel ist also sehr einfach, fast trivial. Dennoch hat die Bayessche Regel weitreichende Konsequenzen für die praktische Anwendung probabilistischen Schließens, wie das Beispiel A.24 zeigen wird. Zuvor wollen wir die Bayes’sche Regel noch etwas verallgemeinern, denn in der obigen Form ist sie noch zu einfach, um für reale Anwendungen von Bedeutung zu sein. In der Realität hat man es mit mehreren möglichen Diagnosen und mit einer ganzen Reihe von Symptomen zu tun. Außerdem wirft die obige Formel noch ein Problem auf: Während die Bestimmung der Wahrscheinlichkeit  $P(D)$  einer Krankheit für Mediziner häufig noch akzeptabel ist, macht die Prognose der Wahrscheinlichkeit des Auftretens eines Symptoms  $P(S)$  (oder einer Symptomkombination) in der Regel große Schwierigkeiten. Nimmt man an, dass die Diagnosemöglichkeiten exklusiv und ausschöpfend gewählt sind, so lassen sich mit Hilfe des Satzes von der totalen Wahrscheinlichkeit A.6 die Symptomwahrscheinlichkeiten eliminieren, und man erhält die Bayessche Formel in ihrer allgemeinen Form:

**Theorem A.23 (Satz von Bayes (allgemein))**

Es sei  $\{B_i \mid i \in \{1, \dots, n\}\}$  eine Menge exklusiver und ausschöpfender Aussagen in Form, und für alle  $i \in \{1, \dots, n\}$  gelte  $P(B_i) > 0$ . Ferner sei  $A$  eine Aussage mit  $P(A) > 0$ . Dann gilt für alle  $i \in \{1, \dots, n\}$

$$P(B_i \mid A) = \frac{P(A \mid B_i)P(B_i)}{\sum_{j=1}^n (P(A \mid B_j)P(B_j))} \quad (\text{A.8})$$

**Beweisidee:** Formel (A.8) ergibt sich leicht aus Formel (A.7) durch Anwendung des Satzes A.20 von der totalen Wahrscheinlichkeit. ■

**Beispiel A.24 (medizinische Diagnose)** Wir betrachten wieder zwei Symptome  $S_1, S_2$ , die zur Diagnose einer Krankheit  $D$  beitragen können. Ein Arzt schätze die folgenden bedingten Wahrscheinlichkeiten:

$$\begin{aligned} P(D) = 0.3 \quad P(S_1 \mid D) = 0.6 \quad P(S_1 \wedge S_2 \mid D) = 0.4 \\ P(S_1 \mid \neg D) = 0.2 \quad P(S_1 \wedge S_2 \mid \neg D) = 0.1 \end{aligned}$$

Aus dem Satz A.20 von der totalen Wahrscheinlichkeit können wir nun die Wahrscheinlichkeiten von  $S_1$  und  $S_1 \wedge S_2$  berechnen:

$$\begin{aligned} P(S_1) &= P(S_1 \mid D)P(D) + P(S_1 \mid \neg D)P(\neg D) \\ &= 0.6 \cdot 0.3 + 0.2 \cdot 0.7 = 0.32 \end{aligned}$$

$$\begin{aligned} P(S_1 \wedge S_2) &= P(S_1 \wedge S_2 \mid D)P(D) + P(S_1 \wedge S_2 \mid \neg D)P(\neg D) \\ &= 0.4 \cdot 0.3 + 0.1 \cdot 0.7 = 0.19 \end{aligned}$$

Mit dem Satz von Bayes können wir jetzt auf die Wahrscheinlichkeit für die Diagnose  $D$  schließen, wenn das Symptom  $S_1$  oder die Symptomkombination  $S_1 \wedge S_2$  festgestellt werden:

$$\begin{aligned} P(D \mid S_1) &= \frac{P(S_1 \mid D)P(D)}{P(S_1)} \\ &= \frac{0.6 \cdot 0.3}{0.32} \approx 0.563 \end{aligned}$$

$$\begin{aligned} P(D \mid S_1 \wedge S_2) &= \frac{P(S_1 \wedge S_2 \mid D)P(D)}{P(S_1 \wedge S_2)} \\ &= \frac{0.4 \cdot 0.3}{0.19} \approx 0.632 \quad \square \end{aligned}$$

An dieser Stelle sei allerdings kritisch bemerkt, dass die Annahme einer exklusiven und ausschöpfenden Menge von Diagnosemöglichkeiten nicht unproblematisch ist. Lässt sich die ausschöpfende Eigenschaft noch relativ leicht durch die Aufnahme der Diagnose “*Diagnose = andere*” herstellen, so kann die geforderte Ausschließlichkeit der Diagnosen ein echtes Problem darstellen: Natürlich können konkurrierende Krankheiten gemeinsam auftreten, auch wenn dies nur selten vorkommen mag.

So gehört die Ausschließlichkeitsannahme zu jenen *vereinfachenden Annahmen*, die man trifft, um überhaupt (meist in vertretbarer Weise) probabilistisch schließen zu können.

## A.5 Mehrwertige Aussagenvariable

Bisher haben wir Aussagenvariable immer als zweiwertig vorausgesetzt, da eine Aussage in der klassischen Logik entweder wahr oder falsch sein kann. Eine Aussagenvariable stand im Wesentlichen also für *eine Aussage*. Eine *mehrwertige Aussagenvariable* hingegen fasst mehrere Aussagen zusammen, von denen jede wie gewohnt wahr oder falsch sein kann, die aber in einem inneren Zusammenhang stehen: Sie sind alternative Ausprägungen eines Merkmals, das von der Aussagenvariablen repräsentiert wird.

**Beispiel A.25 (Aussagenvariable)** Beispiele für Aussagenvariable sind

$$\begin{aligned} \text{Geschlecht} &= \{\text{Mann}, \text{Frau}\} \\ \text{Fieber} &= \{\text{niedrig}, \text{hoch}, \text{kein}\} \end{aligned} \quad \square$$

**Definition A.26 ((mehrwertige) Aussagenvariable)** Eine *(mehrwertige) Aussagenvariable*  $A$  ist eine Funktion

$$A : \Omega \rightarrow \mathcal{A} \subset \text{Form}$$

der Vollkonjunktionen  $\omega$  in eine endliche Menge  $\mathcal{A}$  exklusiver und ausschöpfender Formeln.  $\mathcal{A}$  wird der *Wertebereich* der Variablen  $A$  genannt.  $\square$

Eine Aussagenvariable liefert also eine Zerlegung der Menge  $\Omega$ . Bevor wir diesen Begriff an einem Beispiel näher erläutern, führen wir noch einige nützliche Notationen ein.

Üblicherweise bezeichnet man Aussagenvariable mit großen Buchstaben und ihre Werte mit kleinen Buchstaben und schreibt dann auch

$$A = \{a^{(1)}, \dots, a^{(n)}\}$$

wobei die  $a^{(i)}$  exklusive Formeln sind. Einen beliebigen Wert der Aussagenvariablen  $A$  bezeichnen wir mit  $a$  oder manchmal auch mit  $\dot{a}$  (insbesondere bei zweiwertigen Aussagenvariablen):

$$a, \dot{a} \in \mathcal{A} = \{a^{(1)}, \dots, a^{(n)}\}$$

Mehrwertige Aussagenvariable können auch zweiwertig sein, wie im obigen Beispiel die Variable *Geschlecht*. Oft wird der Wertebereich einer zweiwertigen Variablen  $A$  in der Form  $\{a, \bar{a}\}$  angegeben, wobei

$$\bar{a} = \neg a$$

ist. Ist die Wertigkeit einer Aussagenvariablen  $A$  allerdings größer als 2, so entspricht die Negation eines ihrer Werte  $a$  der Disjunktion der übrigen Werte.

Wir werden im Folgenden annehmen, dass die Buchstaben  $A_1, A_2, \dots, A_n$  unseres logischen Vokabulars mehrwertige Aussagenvariable mit entsprechenden Wertebereichen  $\mathcal{A}_1, \mathcal{A}_2, \dots, \mathcal{A}_n$  repräsentieren. Die Vollkonjunktionen haben dann die Form

$$(A_1 = a_1) \wedge (A_2 = a_2) \wedge \dots \wedge (A_n = a_n), \quad a_i \in \mathcal{A}_i$$

und werden einfacher mit

$$a_1 a_2 \dots a_n$$

bezeichnet, das Konjunktionszeichen  $\wedge$  wird also in der Regel weggelassen. Wenn wir auf die Aussagenvariablen selbst und nicht nur auf bestimmte Ausprägungen Bezug nehmen, dann verwenden wir weiterhin Großbuchstaben. So bezeichnet

$$A_1 A_2 \dots A_n := A_1 \wedge A_2 \wedge \dots \wedge A_n := \{a_1 a_2 \dots a_n \mid a_i \in \mathcal{A}_i, i = 1, \dots, n\}$$

die Menge aller Vollkonjunktionen über  $A_1, A_2, \dots, A_n$ .

**Beispiel A.27 (Kartenspiel)** Die 52 Karten eines handelsüblichen, vollständigen Kartenspiels (ohne Joker) können durch die beiden Aussagenvariablen

$$\begin{aligned} F &: \text{Spielkartenfarbe} &= &\{Herz, Karo, Pik, Kreuz\} \\ W &: \text{Spielkartenwert} &= &\{2, 3, \dots, 10, Bube, Dame, König, As\} \end{aligned}$$

beschrieben werden. Die Menge der Vollkonjunktionen  $\Omega$  ist dann

$$\Omega = FW = \{f\dot{w} \mid f \in \mathcal{F}, w \in \mathcal{W}\}$$

Dem Wert *Herz* von *Spielkartenfarbe* entspricht dann z.B. die Formel

$$\text{Herz} \equiv (\text{Herz}2) \vee (\text{Herz}3) \vee \dots \vee (\text{Herz}As)$$

und *Spielkartenfarbe* ordnet jeder Vollkonjunktion (also jeder Karte)  $f\dot{w}$  diejenige der vier möglichen Formeln *Herz*, *Karo*, *Pik* oder *Kreuz* zu, die aus ihr logisch folgt.

Eine weitere Aussagenvariable ist in diesem Beispiel auch die Variable *Farbe* =  $\{\text{rot}, \text{schwarz}\}$ , wobei z.B. *rot* hier der Formel

$$\text{rot} \equiv \text{Herz} \vee \text{Karo}$$

entspricht. □

Eine entsprechende Notation verwenden wir für Wahrscheinlichkeiten. Eine Wahrscheinlichkeitsfunktion  $P$  über  $\mathbf{A} = \{A_1 A_2 \dots A_n\}$  nennt man auch eine *gemeinsame Verteilung (joint probability distribution)* über  $A_1, A_2, \dots, A_n$ ; wir bezeichnen sie mit  $P(A_1, A_2, \dots, A_n)$  oder  $P(\mathbf{A})$ .  $\mathbf{a}$  repräsentiert eine (beliebige) Vollkonjunktion über  $\mathbf{A}$ .

$P(A_i \mid A_1, \dots, A_{(i-1)}, A_{(i+1)}, \dots, A_n)$  ist die Menge aller bedingten Wahrscheinlichkeiten der genannten Aussagenvariablen:

$$\begin{aligned} P(A_i \mid A_1, \dots, A_{(i-1)}, A_{(i+1)}, \dots, A_n) &:= \\ &\{P(a_i \mid a_1 \dots a_{(i-1)} a_{(i+1)} \dots a_n) \mid a_1 \in \mathcal{A}_1, \dots, a_n \in \mathcal{A}_n\} \end{aligned}$$

und mittels

$$\begin{aligned} P(a_1 \dots a_{(i-1)} a_{(i+1)} \dots a_n) &= \sum_{A_i} P(a_1 \dots a_{(i-1)} A_i a_{(i+1)} \dots a_n) := \\ &\sum_{a_i \in \mathcal{A}_i} P(a_1 \dots a_{(i-1)} a_i a_{(i+1)} \dots a_n) \end{aligned}$$

wird über eine (oder mehrere) Variable marginalisiert.

Diese vereinfachenden Notationen werden im Folgenden von Nutzen sein. So steht z.B. mit den obigen Bezeichnungen

$$P(\mathbf{A}) > 0$$

dafür, dass für alle Vollkonjunktionen  $a_1 \dots a_n$  über  $\mathbf{A}$   $P(a_1 \dots a_n) > 0$  gilt. Weiterhin ist z.B.  $P(A_1)$  die Randverteilung von  $P$  über (der Variablen)  $A_1$ . Allgemeiner wird für eine Teilmenge  $\mathbf{A}' \subseteq \mathbf{A}$  die durch Aufsummieren über alle Variablen in  $\mathbf{A} - \mathbf{A}'$  entstehende Randverteilung von  $P$  über den in  $\mathbf{A}'$  auftretenden Variablen mit  $P(\mathbf{A}')$  bezeichnet:

$$P(\mathbf{A}') = \sum_{\mathbf{A}-\mathbf{A}'} P(\mathbf{A})$$

Sind  $\mathbf{A}', \mathbf{A}''$  zwei (disjunkte) Teilmengen von  $\mathbf{A}$ , so schreiben wir oft  $P(\mathbf{A}', \mathbf{A}'')$  für  $P(\mathbf{A}' \cup \mathbf{A}'')$ ;  $\mathbf{a} = \mathbf{a}' \mathbf{a}''$  ist dann eine Vollkonjunktion über alle Variablen in  $\mathbf{A}' \cup \mathbf{A}''$ . Sind  $\mathbf{A}, \mathbf{B}$  Mengen von Variablen, so steht  $P(\mathbf{A} | \mathbf{B})$  stellvertretend für beliebige bedingte Wahrscheinlichkeiten  $P(\mathbf{a} | \mathbf{b})$ . Insbesondere sei  $P(\mathbf{A} | \emptyset) = P(\mathbf{A})$ .

## A.6 Abhängigkeiten und Unabhängigkeiten

Zu den wichtigsten und angenehmsten Eigenschaften, die Symptome, Krankheiten, Aussagen oder allgemeiner Aussagenvariable vom probabilistischen Standpunkt aus haben können, und die ebenfalls oft als vereinfachende Annahmen in das Design eines Systems eingehen, gehören *probabilistische Unabhängigkeiten*.

**Definition A.28 ((statistische) Unabhängigkeit)** Seien  $\mathbf{A}, \mathbf{B}$  Mengen von Aussagenvariablen aus  $\mathcal{L}$ .  $\mathbf{A}$  und  $\mathbf{B}$  heißen (*statistisch*) *unabhängig*, wenn gilt

$$P(\mathbf{A}, \mathbf{B}) = P(\mathbf{A}) \cdot P(\mathbf{B})$$

d.h., wenn

$$P(\mathbf{ab}) = P(\mathbf{a})P(\mathbf{b})$$

für alle Vollkonjunktionen  $\mathbf{a}, \mathbf{b}$  über  $\mathbf{A}$  bzw.  $\mathbf{B}$ . Gilt diese Beziehung nicht, so heißen  $\mathbf{A}$  und  $\mathbf{B}$  *abhängig*.  $\square$

$\mathbf{A}$  und  $\mathbf{B}$  sind also (statistisch) unabhängig, wenn die Randverteilung über (den Variablen in)  $\mathbf{A}$  und  $\mathbf{B}$  sich als Produkt der Randverteilungen über  $\mathbf{A}$  und  $\mathbf{B}$  ergibt.

**Proposition A.29** Seien  $\mathbf{A}, \mathbf{B}$  Mengen von Aussagenvariablen aus  $\mathcal{L}$ , und es sei  $P(\mathbf{B}) > 0$ .  $\mathbf{A}$  und  $\mathbf{B}$  sind genau dann unabhängig, wenn gilt

$$P(\mathbf{A} | \mathbf{B}) = P(\mathbf{A})$$

**Selbsttestaufgabe A.30 (Unabhängigkeit)** Beweisen Sie Proposition A.29.  $\blacksquare$

Die statistische Unabhängigkeit von Aussagen oder Aussagenvariablen lässt sich leider nur in relativ wenigen Fällen zeigen bzw. annehmen. Sehr viel realistischer erscheinen hingegen meistens Unabhängigkeiten “unter gewissen Annahmen”.

**Definition A.31 (bedingte Unabhängigkeit)** Es seien  $\mathbf{A}, \mathbf{B}, \mathbf{C}$  Mengen von Aussagenvariablen, und es sei  $P(\mathbf{C}) > 0$ .  $\mathbf{A}$  und  $\mathbf{B}$  heißen *bedingt unabhängig bei gegebenem  $\mathbf{C}$* , in Zeichen

$$\mathbf{A} \perp\!\!\!\perp_P \mathbf{B} \mid \mathbf{C}$$

wenn gilt

$$P(\mathbf{A}, \mathbf{B} \mid \mathbf{C}) = P(\mathbf{A} \mid \mathbf{C}) \cdot P(\mathbf{B} \mid \mathbf{C}) \quad (\text{A.9})$$

d.h.

$$P(\mathbf{ab} \mid \mathbf{c}) = P(\mathbf{a} \mid \mathbf{c}) \cdot P(\mathbf{b} \mid \mathbf{c}), \quad \text{wenn } P(\mathbf{c}) > 0$$

für alle entsprechenden Vollkonjunktionen  $\mathbf{a}, \mathbf{b}, \mathbf{c}$ . Sind diese Gleichungen nicht erfüllt, so heißen  $\mathbf{A}$  und  $\mathbf{B}$  *bedingt abhängig bei gegebenem  $\mathbf{C}$* .  $\square$

Aus der Definition ergibt sich sofort, dass die Relation der bedingten Unabhängigkeit symmetrisch ist: Ist  $\mathbf{A}$  bedingt unabhängig von  $\mathbf{B}$  bei gegebenem  $\mathbf{C}$ , so ist auch  $\mathbf{B}$  bedingt unabhängig von  $\mathbf{A}$  bei gegebenem  $\mathbf{C}$ :

$$\mathbf{A} \perp\!\!\!\perp_P \mathbf{B} \mid \mathbf{C} \quad \Rightarrow \quad \mathbf{B} \perp\!\!\!\perp_P \mathbf{A} \mid \mathbf{C}$$

Es sei allerdings schon hier darauf hingewiesen, dass (statistische) Unabhängigkeit und bedingte Unabhängigkeit einander nicht implizieren; siehe hierzu auch Beispiel A.33.

Bevor wir den wichtigen Begriff der bedingten Unabhängigkeit an einem Beispiel erläutern, wollen wir eine äquivalente und sehr gebräuchliche Charakterisierung der bedingten Unabhängigkeit angeben:

**Proposition A.32** *Es seien  $\mathbf{A}, \mathbf{B}, \mathbf{C}$  Mengen von Aussagenvariablen, und es sei weiterhin  $P(\mathbf{C}, \mathbf{B}) > 0$ .  $\mathbf{A}$  und  $\mathbf{B}$  sind genau dann bedingt unabhängig bei gegebenem  $\mathbf{C}$ , wenn gilt*

$$P(\mathbf{A} \mid \mathbf{C}, \mathbf{B}) = P(\mathbf{A} \mid \mathbf{C}) \quad (\text{A.10})$$

$\mathbf{A}$  und  $\mathbf{B}$  sind also genau dann bedingt unabhängig bei gegebenem  $\mathbf{C}$ , wenn bei festliegenden Werten der Variablen aus  $\mathbf{C}$  die Variablen aus  $\mathbf{B}$  keinen Einfluss mehr auf die Variablen in  $\mathbf{A}$  haben.

**Beweis:**[von Proposition A.32]  $\mathbf{A}$  und  $\mathbf{B}$  sind genau dann bedingt unabhängig bei gegebenem  $\mathbf{C}$ , wenn für alle Vollkonjunktionen  $\mathbf{a}, \mathbf{b}, \mathbf{c}$  der Variablen aus  $\mathbf{A}, \mathbf{B}, \mathbf{C}$  gilt

$$\begin{aligned} P(\mathbf{ab} \mid \mathbf{c}) &= P(\mathbf{a} \mid \mathbf{c}) \cdot P(\mathbf{b} \mid \mathbf{c}) \\ \Leftrightarrow \frac{P(\mathbf{abc})}{P(\mathbf{c})} &= \frac{P(\mathbf{ac})}{P(\mathbf{c})} \cdot \frac{P(\mathbf{bc})}{P(\mathbf{c})} \\ \Leftrightarrow \frac{P(\mathbf{ac})}{P(\mathbf{c})} &= \frac{P(\mathbf{abc})P(\mathbf{c})}{P(\mathbf{c})P(\mathbf{bc})} = \frac{P(\mathbf{abc})}{P(\mathbf{bc})} \\ \Leftrightarrow P(\mathbf{a} \mid \mathbf{c}) &= P(\mathbf{a} \mid \mathbf{cb}) \end{aligned}$$

also genau dann, wenn die Gleichung (A.10) gilt. ■

**Beispiel A.33 (gemischte Population)** Wir betrachten eine Population erwachsener Personen unter folgenden Gesichtspunkten:

$$\begin{aligned} G &= \{f, m\} && \text{Geschlecht (} f = \text{Frau, } m = \text{Mann)} \\ R &= \{r, \bar{r}\} && \text{Raucher sein} \\ H &= \{h, \bar{h}\} && \text{verheiratet sein} \\ S &= \{s, \bar{s}\} && \text{schwanger sein} \end{aligned}$$

Die folgende Tabelle enthalte die Wahrscheinlichkeiten der  $2^4 = 16$  Vollkonjunktionen:

		$m$		$f$	
		$r$	$\bar{r}$	$r$	$\bar{r}$
$h$	$s$	0.00	0.00	0.01	0.05
	$\bar{s}$	0.04	0.16	0.02	0.12
$\bar{h}$	$s$	0.00	0.00	0.01	0.01
	$\bar{s}$	0.10	0.20	0.07	0.21

Aus dieser gemeinsamen Verteilung errechnen wir:

$$\begin{aligned} P(f) = 0.5 &= P(m) && P(r) = 0.25 \\ P(s) = 0.08 &&& P(h) = 0.4 \end{aligned}$$

Die Population enthält also gleich viele Frauen wie Männer, ein Viertel der Personen sind Raucher und 40 % sind verheiratet; die Quote der schwangeren Personen in der Gesamtpopulation beträgt 8 %.

Schauen wir uns nun bedingte Wahrscheinlichkeiten an: Wie erwartet ist

$$P(s | m) = 0$$

(nur Frauen können schwanger sein); die Schwangerschaftswahrscheinlichkeit bei den Frauen berechnet sich zu

$$P(s | f) = \frac{P(sf)}{P(f)} = \frac{0.01 + 0.05 + 0.01 + 0.01}{0.5} = 0.16$$

Weiterhin ist

$$P(f | r) = \frac{P(rf)}{P(r)} = \frac{0.01 + 0.02 + 0.01 + 0.07}{0.25} = 0.44$$

d.h. 44 % der Raucher sind Frauen. Damit ist

$$P(f) \neq P(f | r)$$

Die Variablen  $G$  und  $R$  sind also nicht unabhängig (vgl. Proposition A.29).

Bei den Variablen  $G$  und  $H$  hingegen errechnen sich die folgenden marginalen Wahrscheinlichkeiten:

$\dot{g}$	$\dot{h}$	$P(GH)$	$P(G)$	$P(H)$	$P(G)P(H)$
$f$	$\bar{h}$	0.2	0.5	0.4	0.2
$f$	$\bar{h}$	0.3	0.5	0.6	0.3
$m$	$\bar{h}$	0.2	0.5	0.4	0.2
$m$	$\bar{h}$	0.3	0.5	0.6	0.3

Es ist also  $P(\dot{g}\dot{h}) = P(\dot{g})P(\dot{h})$  für alle Ausprägungen der beiden Merkmale, und damit sind die Variablen  $G$  und  $H$  unabhängig. Wegen

$$P(fh | \bar{s}) = \frac{P(fh\bar{s})}{P(\bar{s})} = \frac{0.02 + 0.12}{0.92} \approx 0.152$$

und

$$\begin{aligned} P(f | \bar{s}) \cdot P(h | \bar{s}) &= \frac{P(f\bar{s})}{P(\bar{s})} \cdot \frac{P(h\bar{s})}{P(\bar{s})} \\ &= \frac{0.02 + 0.12 + 0.07 + 0.21}{0.92} \cdot \frac{0.04 + 0.16 + 0.02 + 0.12}{0.92} \\ &= \frac{0.42}{0.92} \cdot \frac{0.34}{0.92} \approx 0.169 \end{aligned}$$

gilt aber

$$P(fh | \bar{s}) \neq P(f | \bar{s}) \cdot P(h | \bar{s})$$

Damit sind  $G$  und  $H$  bedingt abhängig bei gegebenem  $S$ , obwohl  $G$  und  $H$  (statisch) unabhängige Variable sind. □

**Selbsttestaufgabe A.34 (Satz von Bayes)** Berechnen Sie mit dem Satz von Bayes im Beispiel A.33 die bedingte Wahrscheinlichkeit  $P(r | f)$ . ■

**Selbsttestaufgabe A.35 (bedingte Unabhängigkeit)** Sind im Beispiel A.33 die Variablen  $G$  und  $R$  bedingt unabhängig bei gegebenem  $S$ , d.h. gilt  $G \perp\!\!\!\perp R | S$ ? ■

**Selbsttestaufgabe A.36 (Wahrscheinlichkeitsverteilungen)** Es sei die in folgender Tabelle aufgelistete Wahrscheinlichkeitsverteilung über die 3 (binären) Ausgavevariablen  $X, Y, Z$  gegeben:

$x$	$y$	$z$	$P(x, y, z)$	$x$	$y$	$z$	$P(x, y, z)$
0	0	0	$\frac{1}{26}$	1	0	0	$\frac{4}{13}$
0	0	1	$\frac{1}{52}$	1	0	1	$\frac{1}{13}$
0	1	0	$\frac{1}{52}$	1	1	0	$\frac{2}{13}$
0	1	1	$\frac{1}{13}$	1	1	1	$\frac{4}{13}$

1. Berechnen Sie die Randverteilung über  $X$  und  $Y$ ,  $P(X, Y)$ .
2. Berechnen Sie die bedingten Wahrscheinlichkeiten  $P(X | Y)$ .
3. Zeigen Sie, dass  $X$  und  $Y$  bedingt unabhängig sind bei gegebenem  $Z$ . ■



## A.7 Der Begriff der Information

*Information* ist einer der modernen Schlüsselbegriffe. Um richtige Entscheidungen zu treffen und Fehler zu vermeiden, ist es entscheidend, über “gute” Informationen zu verfügen. Verglichen mit dem eher statischen Begriff des Wissens hat Information eine starke dynamische Komponente – sie ist die *Neuigkeit*, die wir (aus Büchern, aus dem Internet oder von Freunden) erfahren und an andere weitergeben. Dementsprechend handelt dann auch die mathematische Informationstheorie von informationsemittierenden Quellen und informationstransportierenden Kanälen (vgl. z. B. [86]). Information wird dort allerdings in einem ganz formalen Rahmen gesehen. Im Folgenden wollen wir diesen abstrakten Informationsbegriff entwickeln und zeigen, wie sich mit Hilfe von Wahrscheinlichkeiten Information quantitativ messen lässt.

Am Anfang steht die Präzisierung des Begriffes: Es ist wichtig, zwischen einer *Nachricht* und dem mit ihr assoziierten *Informationsgehalt* zu unterscheiden. Wenn wir Nachrichten formal behandeln, sie also jeglicher semantischer Bedeutung entledigen, so stellt sich die Frage nach einem Kriterium, das uns Nachrichten interessant erscheinen lässt. Nun ist es so, dass wir überraschende Nachrichten im Allgemeinen als informativer empfinden als die Bestätigung offensichtlicher Vermutungen<sup>1</sup>. Unsere Überraschung ist dabei eng gekoppelt mit der Wahrscheinlichkeit, die wir dieser Nachricht (bzw. dem mit ihr verbundenen Ereignis) zuweisen. Es liegt also nahe, den Informationsgehalt einer Nachricht in Abhängigkeit von ihrer Wahrscheinlichkeit zu definieren:

$$\text{Information}(\text{Nachricht}) = \text{Inf}(P(\text{Nachricht}))$$

wobei

$$\text{Inf} : [0, 1] \rightarrow \mathbb{R}^{\geq 0} \cup \{\infty\}$$

eine stetige, monoton fallende Funktion ist. Der unmöglichen Nachricht wird dabei ein unendlicher Informationsgehalt zugewiesen, während bekannte Nachrichten den Informationsgehalt 0 haben. Zu Normierungszwecken legt man außerdem noch fest, welche Wahrscheinlichkeit mit dem Wert 1 assoziiert werden soll. Im Allgemeinen fixiert man

$$\text{Inf}(0.5) = 1 \tag{A.11}$$

Eine weitere, entscheidende Forderung an *Inf* ist die folgende: Der Informationsgehalt voneinander unabhängiger Nachrichten soll die Summe der einzelnen Informationsgehalte sein. Da die gemeinsame Wahrscheinlichkeit unabhängiger Nachrichten gerade das Produkt der einzelnen Wahrscheinlichkeiten ist, bedeutet dies

$$\text{Inf}(x_1 x_2) = \text{Inf}(x_1) + \text{Inf}(x_2)$$

Diese Funktionalgleichung bestimmt *Inf* nun eindeutig: Es ist

$$\text{Inf}(x) := -\log_2 x \tag{A.12}$$

---

<sup>1</sup> Die Betonung liegt hier auf “offensichtlich” – sicherlich war der Nachweis der berühmten *Fermat’schen Vermutung* ein Meilenstein in der modernen Mathematik und überaus informativ!

also der negative duale Logarithmus<sup>2</sup>. Die Basis 2 wird durch die Normierung (A.11) bedingt. Der duale Logarithmus kann nach der folgenden Formel aus einem Logarithmus zur Basis  $b$  (z. B.  $b = 10$  oder  $b = e$ ) berechnet werden:

$$\log_2 x = \frac{\log_b x}{\log_b 2}$$

Die Maßeinheit der Information ist das *bit*. Die Fixierung der Basis spielt allerdings oft nur eine sekundäre Rolle, wir werden daher im Folgenden oft nur  $\log$  statt  $\log_2$  schreiben.

Der in *bit* gemessene Informationsgehalt einer Nachricht gibt die Anzahl der Ja/Nein-Antworten an, die bei einer optimalen Fragestrategie nötig sind, um diese Nachricht zu isolieren. 1 *bit* entspricht dabei dem Informationsgehalt einer Ja/Nein-Antwort (s. auch [216]).

Die Gleichung (A.12) liefert einen überaus griffigen Maßbegriff für Information. Bei dieser Quantifizierung der Information bleibt allerdings nicht nur ihr Inhalt unberücksichtigt, es fließen auch keinerlei Bewertungen oder Nutzensvorstellungen ein. Das heißt jedoch nicht, dass die formale Information einer Nachricht ein absolut objektiver Begriff ist (was unserer Intuition widerspräche). Durch die Zuordnung von Wahrscheinlichkeiten zu Nachrichten kann eine subjektive Komponente ins Spiel kommen. Im Allgemeinen werden wir es aber eher mit statistischen Wahrscheinlichkeiten zu tun haben, wie auch im folgenden Beispiel.

**Beispiel A.37** In einer Urne befinden sich insgesamt 8 Kugeln, 4 weiße, 2 rote und je 1 blaue und grüne Kugel. In einem Zug werde jeweils 1 Kugel (mit Zurücklegen) gezogen. Die Wahrscheinlichkeit, eine weiße (rote, blaue, grüne) Kugel zu ziehen, beträgt dann  $\frac{1}{2}$  ( $\frac{1}{4}$ ,  $\frac{1}{8}$ ,  $\frac{1}{8}$ ). Folglich lässt sich die mit dem Ausgang eines Zuges verbundene Information berechnen als

$$\text{Information}(weiss) = \text{Inf}(\frac{1}{2}) = 1\text{bit}$$

$$\text{Information}(rot) = \text{Inf}(\frac{1}{4}) = 2\text{bit}$$

$$\text{Information}(blau) = \text{Inf}(\frac{1}{8}) = 3\text{bit}$$

$$\text{Information}(grün) = \text{Inf}(\frac{1}{8}) = 3\text{bit} \quad \square$$

## A.8 Entropie

Führt man die gedankliche Verbindung zwischen Wahrscheinlichkeiten und Information weiter, so ist nicht nur die mittels Wahrscheinlichkeitsverteilung  $P$  mit jedem einzelnen Elementarereignis  $\omega$  assoziierte Information von Bedeutung. Um einen Gesamteindruck von der zu erwartenden Information zu bekommen, berechnet man ferner den *mittleren Informationsgehalt*

<sup>2</sup> Die Lösung solcher Funktionalgleichungen ist mathematischer Standard. Für einen Beweis im Rahmen der Informationstheorie vgl. [86], S. 110f.

$$H(P) = - \sum_{\omega} P(\omega) \log_2 P(\omega)$$

wobei man  $0 \log_2 0 = 0$  setzt.  $H(P)$  heißt die *Entropie* der Verteilung  $P$ . Sie misst die *Unsicherheit* bezüglich eines zu erwartenden Elementarereignisses, die man in Kenntnis der Verteilung  $P$  empfindet.

Der Begriff “Entropie” wurde zuerst in der statistischen Mechanik in Verbindung mit thermodynamischen Systemen benutzt. Sie war diejenige physikalische Größe, die man maximierte, um eine möglichst indifferente Ausgangsverteilung zur Beschreibung des Systems zu erhalten (s. [93]). Shannon [194] erkannte später, dass die Entropie nichts anderes war als die Unbestimmtheit, die man dem System bei gegebenen Randbedingungen zubilligen musste, und benutzte Entropie als grundlegenden Begriff in der Informationstheorie.

An dieser Stelle ist es wichtig, sich noch einmal bewusst zu machen, dass unsere Information umso größer ist, je unsicherer wir über das Eintreffen eines bestimmten Ereignisses sind. Wenn wir bereits im Voraus hundertprozentig wissen, mit welchem Elementarereignis  $\omega_0$  wir zu rechnen haben, d. h. ist  $P(\omega_0) = 1$  und  $P(\omega) = 0$  für alle anderen  $\omega \neq \omega_0$ , so ist  $H(P) = 0$ . Ist hingegen  $P$  eine Gleichverteilung über den betrachteten Variablen, so haben wir offensichtlich überhaupt keine Vorstellung, was passieren wird. In diesem Fall ist  $H(P)$  – ebenso wie unsere “mittlere Überraschung” – maximal, wie der folgende Satz zeigt:

**Proposition A.38** *Bezeichne  $P_0$  die Gleichverteilung über  $\Omega$ , d. h.  $P_0(\omega) = \frac{1}{n}$  für  $\omega \in \Omega$ , wobei  $|\Omega| = n$ .*

1. *Es ist  $H(P_0) = \log n$ .*
2. *Für jede beliebige Wahrscheinlichkeitsverteilung  $P$  über  $\Omega$  gilt  $H(P) \leq H(P_0)$ .*

**Selbsttestaufgabe A.39 (Maximale Entropie)** Beweisen Sie Proposition A.38. *Hinweis:* Verwenden Sie für den Nachweis des zweiten Teils die Ungleichung

$$\sum_{i=1}^m x_i \log \frac{x_i}{y_i} \geq 0, \tag{A.13}$$

wobei  $(x_1, \dots, x_m), (y_1, \dots, y_m)$  Tupel nichtnegativer reeller Zahlen mit  $\sum_{i=1}^m x_i = \sum_{i=1}^m y_i$  sind. Für einen Beweis dieser Ungleichung siehe z. B. [86], S. 111. ■

Die Entropie ist ein Maß, das auf Wahrscheinlichkeitsverteilungen angewendet wird. Dabei spielt es keine Rolle, ob es sich um die gemeinsame Verteilung mehrerer Aussagenvariablen handelt oder um die (marginale) Verteilung über eine bestimmte Aussagenvariable  $A = \{a^{(1)}, \dots, a^{(n)}\}$ . In letzterem Fall bezeichnet man die Entropie der entsprechenden Verteilung auch als die Entropie von  $A$ , also

$$H(A) = - \sum_{i=1}^n P(a^{(i)}) \log P(a^{(i)}) \tag{A.14}$$

wobei  $P$  als gegeben angenommen wird.  $H(A)$  drückt unsere mittlere Unsicherheit darüber aus, welchen Wert die Variable  $A$  annehmen wird.

Betrachten wir zwei Aussagenvariable  $A = \{a^{(1)}, \dots, a^{(n)}\}$ ,  $B = \{b^{(1)}, \dots, b^{(m)}\}$  und eine gemeinsame Verteilung  $P$  über  $A$  und  $B$ . Dann lässt sich oft aus dem Wert der einen Variablen Information über den Wert der anderen gewinnen. Der *bedingte Informationsgehalt*, den  $B = b^{(j)}$  für  $A = a^{(i)}$  besitzt, wird gemessen durch  $-\log P(a^{(i)}|b^{(j)})$ . Summiert man über alle Werte auf und berücksichtigt die entsprechenden bedingten Wahrscheinlichkeiten, so erhält man die *erwartete Unbestimmtheit von  $A$  nach der Beobachtung von  $B$*  als

$$H(A|B) = - \sum_{i,j} P(b^{(j)})P(a^{(i)}|b^{(j)}) \log P(a^{(i)}|b^{(j)})$$

$H(A|B)$  wird kurz als die *bedingte Entropie von  $A$  bezüglich  $B$*  bezeichnet. Die Unbestimmtheit bezüglich  $A$  und  $B$  wird als *Verbundentropie* bezeichnet und mit  $H(A, B)$  bezeichnet:

$$H(A, B) = - \sum_{i,j} P(a^{(i)}b^{(j)}) \log P(a^{(i)}b^{(j)})$$

Zwischen Verbund- und bedingter Entropie besteht der folgende Zusammenhang:

$$H(A, B) = H(B) + H(A|B) = H(A) + H(B|A) \quad (\text{A.15})$$

(A.15) entspricht der Vorstellung, dass Information grundsätzlich additiv ist, und lässt sich leicht nachrechnen. Bereinigt man nun die Entropie von  $A$  um die bedingte Entropie von  $A$  bezüglich  $B$ , so erhält man ein Maß für die Information, die  $B$  für  $A$  bereithält:

$$\begin{aligned} \text{Inf}(A \parallel B) &= H(A) - H(A|B) && (\text{A.16}) \\ &= - \sum_i P(a^{(i)}) \log P(a^{(i)}) - \left( - \sum_{i,j} P(b^{(j)})P(a^{(i)}|b^{(j)}) \log P(a^{(i)}|b^{(j)}) \right) \\ &= \sum_{i,j} P(a^{(i)}b^{(j)}) \log \frac{P(a^{(i)}b^{(j)})}{P(b^{(j)})} - \sum_i \left( \sum_j P(a^{(i)}b^{(j)}) \right) \log P(a^{(i)}) \\ &= \sum_{i,j} P(a^{(i)}b^{(j)}) \log \frac{P(a^{(i)}b^{(j)})}{P(b^{(j)})} - \sum_{i,j} P(a^{(i)}b^{(j)}) \log P(a^{(i)}) \\ &= \sum_{i,j} P(a^{(i)}b^{(j)}) \left( \log \frac{P(a^{(i)}b^{(j)})}{P(b^{(j)})} - \log P(a^{(i)}) \right) \\ &= \sum_{i,j} P(a^{(i)}b^{(j)}) \log \frac{P(a^{(i)}b^{(j)})}{P(a^{(i)})P(b^{(j)})} \end{aligned}$$

$\text{Inf}(A \parallel B)$  wird *gegenseitige Information (mutual information)* oder *eigentliche Information (information proper)* genannt. Sie ist symmetrisch in  $A$  und  $B$ . Wenn  $A$  und  $B$  unabhängig sind, d. h.  $P(a^{(i)}b^{(j)}) = P(a^{(i)})P(b^{(j)})$  für alle  $i, j$ , so ist  $\text{Inf}(A \parallel B) = 0$  – in diesem Fall liefert keine der beiden Variablen irgendwelche Information über die andere.

Schließlich wollen wir uns mit dynamischen Veränderungen von Wahrscheinlichkeitsverteilungen, wie sie z. B. in Zeitreihenanalysen vorkommen, beschäftigen. Nehmen wir einmal an, die Verhältnisse, die zur Erstellung der Verteilung  $P$  geführt haben, ändern sich, und die der neuen Situation angemessene Verteilung sei  $Q$ . Es soll der Informationsgewinn berechnet werden, der sich daraus ergibt, dass man diese Änderung bemerkt hat. In diesem Fall benutzt man zur Bestimmung der Entropie korrekterweise die neuen Informationswerte  $-\log Q(\omega)$ . Ist einem die Änderung jedoch verborgen geblieben, so rechnet man weiter mit  $-\log P(\omega)$ . Die Gewichts-faktoren werden in jedem Fall von der aktuellen Verteilung  $Q$  bestimmt, so dass sich der gesuchte Informationsgewinn als Differenz

$$\begin{aligned} R(Q, P) &:= -\sum_{\omega} Q(\omega) \log P(\omega) - \left( -\sum_{\omega} Q(\omega) \log Q(\omega) \right) \\ &= \sum_{\omega} Q(\omega) \log \frac{Q(\omega)}{P(\omega)} \end{aligned} \quad (\text{A.17})$$

berechnet.  $R(Q, P)$  wird als *relative Entropie (cross-entropy) von  $Q$  bzgl.  $P$*  bezeichnet. Sie misst den Informationsabstand zwischen  $P$  und  $Q$  auf der Basis von  $Q$ . Sie ist zwar nicht symmetrisch, d. h. im Allgemeinen gilt  $R(Q, P) \neq R(P, Q)$ , jedoch positiv definit, es ist also immer  $R(Q, P) \geq 0$  (wegen der Ungleichung (A.13)), und  $R(Q, P) = 0$  genau dann, wenn  $P = Q$ . Die relative Entropie ist eines der wichtigsten Grundkonzepte in der Informationstheorie, denn sowohl die gegenseitige Information (A.16) als auch die Entropie selbst lassen sich (bis auf eine Konstante) als relative Entropien auffassen. Bei der gegenseitigen Information sieht man das sofort aus der Definition von  $\text{Inf}(A \parallel B)$ ; für die Entropie zeigt dies der folgende Satz:

**Proposition A.40** Sei  $P = (p_1, \dots, p_n)$  eine Wahrscheinlichkeitsverteilung, und sei  $P_0 = (\frac{1}{n}, \dots, \frac{1}{n})$  eine passende Gleichverteilung. Dann ist

$$R(P, P_0) = \log n - H(P)$$

**Selbsttestaufgabe A.41 (Relative Entropie)** Beweisen Sie Proposition A.40. ■

**Selbsttestaufgabe A.42 (Entropie)** Sei  $\Omega = \{\omega_1, \dots, \omega_n\}$  die Menge der Elementarereignisse, und es sei  $P$  eine Wahrscheinlichkeitsverteilung auf  $\Omega$  mit  $P(\omega_i) = p_i$ .  $H(P) = H(p_1, \dots, p_n) = -\sum_{i=1}^n p_i \log p_i$  ist die Entropie von  $P$ .

1. Sei  $2 \leq k \leq n - 1$  ein fester Index, und sei  $s := p_1 + \dots + p_k$ . Zeigen Sie:

$$H(p_1, \dots, p_n) = H(s, p_{k+1}, \dots, p_n) + sH\left(\frac{p_1}{s}, \dots, \frac{p_k}{s}\right)$$

2. Sei  $X \subset \Omega$  ein Ereignis mit  $X \neq \emptyset, X \neq \Omega$ . Sei  $Y := \Omega - X$ . (Wir wählen hier eine mengentheoretische Schreibweise für Ereignisse.) Mit  $P_X$  und  $P_Y$  bezeichnen wir die bedingten Verteilungen nach  $X$  und  $Y$ , d. h.  $P_X = P(\cdot | X)$  und  $P_Y = P(\cdot | Y)$ . Zeigen Sie (unter Verwendung von Teil 1):

$$H(P) = H(P(X), P(Y)) + P(X)H(P_X) + P(Y)H(P_Y) \quad \blacksquare$$

## B Graphentheoretische Grundlagen

Soweit wir sie für die Behandlung probabilistischer Netzwerke benötigen, stellen wir hier die graphentheoretischen Grundlagen vor. Neben den grundlegenden Definitionen von gerichteten und ungerichteten Graphen sind dies insbesondere die Konzepte moraler und triangulierter Graphen, die sog. *running intersection property* (fortlaufende Schnitteigenschaft) sowie Hypergraphen.

### B.1 Graphen und Cliques

**Definition B.1 (gerichteter Graph)** Ein *gerichteter Graph* ist ein Paar  $\mathcal{G} = \langle \mathbf{V}, \mathcal{E} \rangle$ , wobei  $\mathbf{V}$  eine Menge von *Ecken* (*vertices*) oder *Knoten* (*nodes*) ist und  $\mathcal{E} \subseteq \mathbf{V} \times \mathbf{V}$  eine Menge von Knotenpaaren  $(v, w)$  ist, den *Kanten* (*edges*) von  $\mathcal{G}$ .  $\square$

Wir werden im Folgenden stets annehmen, dass die Menge  $\mathbf{V}$  der Knoten eines Graphen endlich ist. Weiterhin setzen wir voraus, dass  $\mathcal{E}$  keine Schlingen enthält, d.h., für alle  $v \in \mathbf{V}$  ist  $(v, v) \notin \mathcal{E}$ .

Während Kanten in in einem ungerichteten Graphen oft als zweielementige Knotenmengen dargestellt werden, führt die folgende Definition ungerichtete Graphen als Spezialfall gerichteter Graphen ein (wie z.B. auch in [80]), so dass wir für viele Konzepte dieselbe Notation für beide Arten von Graphen verwenden können.

**Definition B.2 (ungerichteter Graph)** Ein *ungerichteter Graph* ist ein gerichteter Graph  $\mathcal{G} = \langle \mathbf{V}, \mathcal{E} \rangle$ , in dem die Relation  $\mathcal{E}$  symmetrisch ist, d.h., für alle  $v, w \in \mathbf{V}$  gilt:

$$(v, w) \in \mathcal{E} \Rightarrow (w, v) \in \mathcal{E} \quad \square$$

Wir fassen also  $\{(v, w), (w, v)\}$  als *eine* Kante des ungerichteten Graphen auf.<sup>1</sup> In Abbildungen werden wir die Kanten in einem gerichteten Graphen durch Pfeile und die Kanten in einem ungerichteten Graphen durch Linien zwischen den beteiligten Knoten darstellen (vgl. Abbildungen B.1 – B.3). Weiterhin treffen wir folgende Vereinbarung: Wenn wir davon sprechen, in einem ungerichteten Graphen eine Kante zwischen den Knoten  $v$  und  $w$  einzufügen, so meinen wir damit immer das Hinzufügen der Kanten  $(v, w)$  und  $(w, v)$  zur Kantenmenge; das Entfernen einer Kante zwischen  $v$  und  $w$  entspricht dem Entfernen der Kanten  $(v, w)$  und  $(w, v)$ .

<sup>1</sup> Diese Sichtweise einer Kante eines ungerichteten Graphen als die Kombination von  $(v, w)$  und  $(w, v)$  entspricht also in eindeutiger Weise der Darstellung  $\{v, w\}$  einer Kante eines ungerichteten Graphen als zweielementige Menge. Formal kann man die Kanten  $(v, w)$  und  $(w, v)$  mit Hilfe der Äquivalenzrelation  $(v, w) \sim (w, v)$  identifizieren.

**Definition B.3 ((einfacher) Weg, Zyklus)** Sei  $\mathcal{G} = \langle \mathbf{V}, \mathcal{E} \rangle$  ein Graph (gerichtet oder ungerichtet). Ein *Weg (path)* der Länge  $n (\geq 1)$  zwischen zwei Knoten  $v, v' \in \mathbf{V}$  ist eine Folge von Knoten

$$v_0, v_1, \dots, v_n$$

so dass  $v = v_0, v' = v_n$  gilt und für jedes  $i \in \{1, \dots, n\}$   $(v_{i-1}, v_i) \in \mathcal{E}$  ist.

Ein Weg  $v_0, v_1, \dots, v_n$  heißt *einfach*, falls alle Knoten paarweise verschieden sind (außer evtl.  $v_0 = v_n$ ), d.h., falls folgende zwei Bedingungen gelten:

- $v_i \neq v_j$  für  $i \neq j$  und  $0 \leq i, j \leq n - 1$  (keine zwei Kanten des Weges haben denselben Anfangsknoten)
- $v_i \neq v_j$  für  $i \neq j$  und  $1 \leq i, j \leq n$  (keine zwei Kanten des Weges haben denselben Endknoten)

Ein *Zyklus (cycle)* der Länge  $n$  ist ein Weg  $v_0, v_1, \dots, v_{n-1}, v_n$  mit  $v_0 = v_n$ , d.h. mit identischem Anfangs- und Endpunkt; für einen ungerichteten Graphen verlangen wir zusätzlich, dass der Weg mindestens drei verschiedene Knoten enthält.<sup>2</sup>

□

**Definition B.4 (Adjazenz, Nachbar)** Zwei Knoten  $u, v \in \mathbf{V}$  eines ungerichteten Graphen  $\mathcal{G} = \langle \mathbf{V}, \mathcal{E} \rangle$  heißen *adjazent (adjacent)* oder *benachbart*, wenn sie durch eine Kante verbunden sind. Die Menge der Nachbarn eines Knoten  $u$  ist

$$nb(u) := \{v \in \mathbf{V} \mid (u, v) \in \mathcal{E}\}$$

□

**Definition B.5 (azyklisch, DAG)** Ein Graph heißt *azyklisch*, wenn er keinen Zyklus enthält. Ein gerichteter, azyklischer Graph wird mit dem Kürzel *DAG (directed acyclic graph)* bezeichnet.

□

**Definition B.6 (Elternknoten, Kindknoten)** Sei  $\mathcal{G} = \langle \mathbf{V}, \mathcal{E} \rangle$  ein DAG, seien  $v, w \in \mathbf{V}$  Knoten.  $w$  heißt *Elternknoten (parent)* von  $v$ , wenn  $(w, v) \in \mathcal{E}$ .  $v$  heißt in diesem Falle *Kindknoten (child)* von  $w$ . Die Menge der Elternknoten eines Knotens  $v$  wird mit  $pa(v)$  bezeichnet:

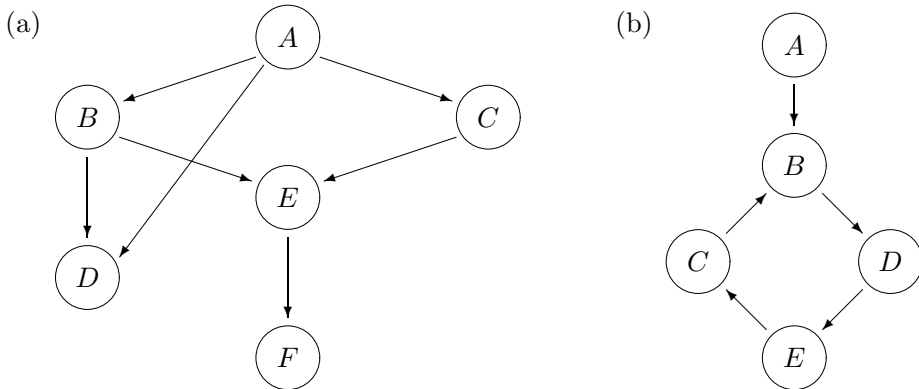
$$pa(v) = \{w \in \mathbf{V} \mid (w, v) \in \mathcal{E}\}$$

□

Der transitive Abschluss dieser Relationen liefert die Begriffe *Vorfahren* und *Nachkommen*:

**Definition B.7 (Vorfahren, Nachkommen)** Sei  $\mathcal{G} = \langle \mathbf{V}, \mathcal{E} \rangle$  ein DAG, seien  $v, w \in \mathbf{V}$  Knoten.  $w$  heißt *Vorfahr (ancestor)* von  $v$  und  $v$  heißt *Nachkomme (descendant)* von  $w$ , wenn es in  $\mathcal{G}$  einen Weg von  $w$  nach  $v$  gibt. Für die Menge der Vorfahren bzw. Nachkommen eines Knoten  $v$  führen wir die folgenden Bezeichnungen ein:

<sup>2</sup> Damit ist sichergestellt, dass in einem ungerichteten Graphen nicht schon eine einzelne ungerichtete Kante ein Zyklus ist.



**Abbildung B.1** DAG mit 6 Knoten (a) und ein gerichteter Graph, der kein DAG ist (b)

$$\begin{aligned}
 an(v) &= \{u \in \mathbf{V} \mid u \text{ ist Vorfahre von } v\} \\
 de(v) &= \{u \in \mathbf{V} \mid u \text{ ist Nachkomme von } v\}
 \end{aligned}$$

Eine *Vorfahrenmenge* (*ancestral set*) ist eine Menge  $\mathbf{A} \subseteq \mathbf{V}$ , die abgeschlossen ist bzgl. der Vorfahrenrelation, d.h.  $an(v) \subseteq \mathbf{A}$  für alle  $v \in \mathbf{A}$ . Für eine Knotenmenge  $\mathbf{W} \subseteq \mathbf{V}$  bezeichnet

$$An(\mathbf{W}) = \mathbf{W} \cup \bigcup_{w \in \mathbf{W}} an(w)$$

die kleinste Vorfahrenmenge, die  $\mathbf{W}$  enthält. Weiterhin bezeichne

$$nd(v) = \mathbf{V} - (de(v) \cup \{v\})$$

die Menge aller Knoten, die von  $v$  verschieden und auch keine Nachkommen von  $v$  sind (*non-descendants*). □

**Beispiel B.8 (gerichtete Graphen, DAG)** Abbildung B.1(a) zeigt einen DAG mit 6 Knoten. Der Graph in Abbildung B.1(b) ist gerichtet, aber kein DAG, da er den Zyklus  $B, D, E, C, B$  enthält. Für den Graphen in Abbildung B.1(a) ist  $pa(B) = \{A\}$  und  $de(B) = \{D, E, F\}$ . □

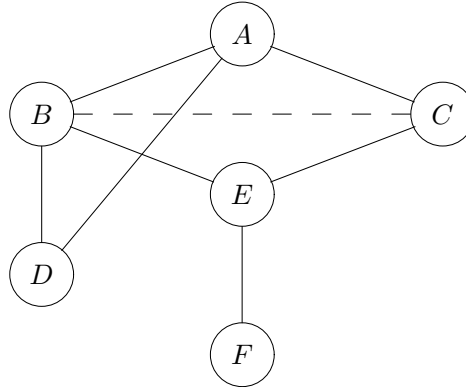
Aus einem gerichteten Graphen entsteht auf einfache Weise ein ungerichteter Graph durch Ignorieren der Richtungen:

**Definition B.9 (ungerichteter Graph eines gerichteten Graphen)** Sei  $\mathcal{G} = \langle \mathbf{V}, \mathcal{E}^d \rangle$  ein gerichteter Graph. Der zu  $\mathcal{G}$  gehörige (*ungerichtete*) Graph  $\mathcal{G}'$  ist der Graph  $\mathcal{G}' = \langle \mathbf{V}, \mathcal{E} \rangle$  mit

$$\mathcal{E} = \mathcal{E}^d \cup \{(w, v) \mid (v, w) \in \mathcal{E}^d\}$$
□

Die folgende Definition dient dazu, beim Übergang von einem gerichteten Graphen zu einem ungerichteten Graphen die Information, die in den Richtungen der Kanten steckt, nicht völlig aufzugeben. Dabei wird nicht einfach der zu einem DAG gehörige ungerichtete Graph als Ausgangspunkt genommen, sondern ein modifizierter Graph.





**Abbildung B.2** Moraler Graph zu Abb. B.1(a); die gestrichelte Linie wurde hinzugefügt

**Definition B.10 (moraler Graph)** Sei  $\mathcal{G} = \langle \mathbf{V}, \mathcal{E} \rangle$  ein DAG. Der *moral Graph* (*moral graph*)  $\mathcal{G}_m$  entsteht aus  $\mathcal{G}$  in zwei Schritten:

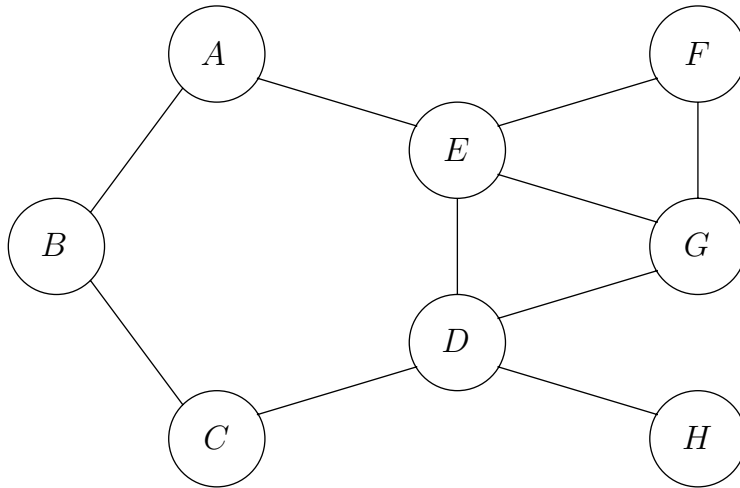
- Sind zwei Knoten  $u, v$  Elternknoten eines gemeinsamen Kindknotens, d.h., gibt es ein  $w \in \mathbf{V}$  mit  $u, v \in pa(w)$ , und sind diese noch nicht durch eine Kante verbunden, so füge eine Kante  $(u, v)$  oder  $(v, u)$  zu  $\mathcal{E}$  hinzu. So entsteht ein (gerichteter) Graph  $\mathcal{G}_m^d$ , in dem alle Eltern eines gemeinsamen Kindes durch eine Kante verbunden sind.
- $\mathcal{G}_m$  ist der zu  $\mathcal{G}_m^d$  gehörige ungerichtete Graph. □

Tatsächlich wäre es korrekt und auch angemessen, das englische “moral” mit “moralisch” zu übersetzen, also vom “moralischen Graphen” zu sprechen. Denn die Idee, die der Konstruktion dieses Graphen zugrunde liegt, ist nichts anderes als diejenige, eine “Heirat” von Elternpaaren gemeinsamer Kinder – graphentheoretisch(!) – zu erzwingen. Um einer sachlicheren Darstellung willen ziehen wir es jedoch vor, das Kunstwort “moraler Graph” zu benutzen. Der kuriose Moralisierungsgedanke darf dennoch als intuitive Beschreibung der Aufgabe dieses Graphen bestehen bleiben.

**Beispiel B.11 (moraler Graph)** Abbildung B.2 zeigt den moralen Graphen zu Abbildung B.1(a). □

**Definition B.12 (vollständiger Graph, leerer Graph)** Ein ungerichteter Graph  $\mathcal{G} = \langle \mathbf{V}, \mathcal{E} \rangle$  heißt *vollständig* (*complete*), wenn je zwei Knoten aus  $\mathbf{V}$  durch eine Kante verbunden sind, d.h., wenn gilt  $(v, w) \in \mathcal{E}$  für alle  $v, w \in \mathbf{V}$ .  $\mathcal{G}$  heißt *leerer Graph*, wenn seine Kantenmenge leer ist:  $\mathcal{E} = \emptyset$ . □

**Definition B.13 (Clique)** Sei  $\mathcal{G} = \langle \mathbf{V}, \mathcal{E} \rangle$  ein ungerichteter Graph. Eine Teilmenge  $\mathbf{C} \subseteq \mathbf{V}$  heißt *Clique* von  $\mathcal{G}$ , wenn  $\mathbf{C}$  eine *maximale vollständige* Menge ist, d.h., wenn jedes Paar verschiedener Knoten aus  $\mathbf{C}$  durch eine Kante aus  $\mathcal{E}$  miteinander



**Abbildung B.3** Ungerichteter Graph mit 8 Knoten

verbunden ist, und wenn  $\mathbf{C}$  bzgl. dieser Eigenschaft maximal unter den Teilmengen von  $\mathbf{V}$  ist, wenn es also keine andere vollständige Teilmenge von  $\mathbf{V}$  gibt, die  $\mathbf{C}$  enthält.  $\square$

**Beispiel B.14 (Cliques)** Abbildung B.3 zeigt einen ungerichteten Graphen mit Knotenmenge  $\mathbf{V} = \{A, B, C, D, E, F, G, H\}$ . Hier gibt es 7 Cliques:

$$\begin{array}{llll} \mathbf{C}_1 = \{A, B\} & \mathbf{C}_2 = \{B, C\} & \mathbf{C}_3 = \{C, D\} & \mathbf{C}_4 = \{D, H\} \\ \mathbf{C}_5 = \{D, E, G\} & \mathbf{C}_6 = \{E, F, G\} & \mathbf{C}_7 = \{A, E\} & \end{array}$$

$\square$

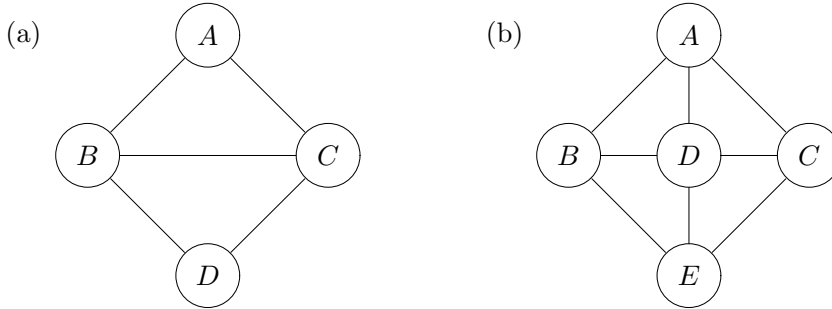
## B.2 Triangulierte Graphen

**Definition B.15 (Sehne)** Sei  $\mathcal{G} = \langle \mathbf{V}, \mathcal{E} \rangle$  ein ungerichteter Graph. Eine *Sehne* eines einfachen Weges oder Zyklus'  $v_0, v_1, \dots, v_n$  in  $\mathcal{G}$  ist eine Kante zwischen zwei nicht aufeinanderfolgenden Knoten  $v_i, v_j, |i - j| > 1$ .  $\square$

**Definition B.16 (triangulierter Graph)** Ein ungerichteter Graph heißt *trianguliert*, wenn jeder einfache Zyklus der Länge  $> 3$  eine Sehne besitzt.  $\square$

**Beispiel B.17 (Sehne eines Zyklus)** Abbildung B.3 zeigt einen ungerichteten Graphen. Die Kante  $\{E, G\}$  ist eine Sehne des Zyklus  $E, F, G, D, E$  der Länge 4. Dennoch ist dieser Graph nicht trianguliert, denn der Zyklus  $A, B, C, D, E, A$  der Länge 5 besitzt keine Sehne.  $\square$

**Selbsttestaufgabe B.18 (triangulierte Graphen)** Entscheiden Sie für jeden der beiden Graphen in Abbildung B.4, ob er trianguliert ist.  $\blacksquare$



**Abbildung B.4** Graphen zu Aufgabe B.18

Um einen beliebigen ungerichteten Graphen  $\mathcal{G} = \langle \mathbf{V}, \mathcal{E} \rangle$  zu triangulieren, werden ihm Kanten hinzugefügt. Diesen Vorgang bzw. die entsprechende Kantenmenge nennt man *Fill-in*. Zu seiner Konstruktion nimmt man eine lineare Ordnung  $\alpha$  auf der Knotenmenge  $\mathbf{V}$  an.

**Definition B.19 (Fill-in, Fill-in-Graph)** Sei  $\mathcal{G} = \langle \mathbf{V}, \mathcal{E} \rangle$  ein ungerichteter Graph, und sei  $\alpha$  eine lineare Ordnung auf den Knoten von  $\mathcal{G}$ .

- Der *Fill-in* von  $\mathcal{G}$  bzgl.  $\alpha$  ist die Kantenmenge  $\mathcal{F}(\alpha)$ , wobei

$(v, w) \in \mathcal{F}(\alpha)$  gdw.  $(v, w) \notin \mathcal{E}$  und es gibt einen Weg zwischen  $v$  und  $w$ , der außer  $v$  und  $w$  nur Knoten enthält, die bzgl. der Ordnung  $\alpha$   $v$  und  $w$  nachgeordnet sind. D.h., ist  $u \notin \{v, w\}$  ein Knoten auf diesem Weg, so gilt  $v < u$  und  $w < u$  bzgl.  $\alpha$ .

- Der *Fill-in-Graph* von  $\mathcal{G}$  bezüglich  $\alpha$  ist der Graph

$$\mathcal{G}(\alpha) = \langle \mathbf{V}, \mathcal{E} \cup \mathcal{F}(\alpha) \rangle \quad \square$$

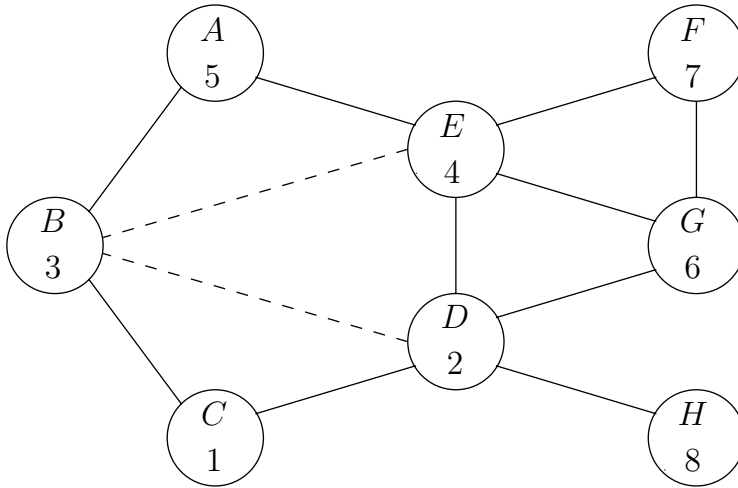
Der Fill-in-Graph wird in der Literatur – ein wenig irreführend – als *Eliminationsgraph* bezeichnet. Diese Namensgebung spielt auf ein Triangulationsverfahren an, das mittels sukzessiver Knotenelimination arbeitet; vgl. z. B. [157].

**Proposition B.20** Sei  $\mathcal{G} = \langle \mathbf{V}, \mathcal{E} \rangle$  ein ungerichteter Graph, und sei  $\alpha$  eine Ordnung auf den Knoten von  $\mathcal{G}$ . Der Fill-in-Graph von  $\mathcal{G}$  bezüglich  $\alpha$ ,  $\mathcal{G}(\alpha)$ , ist trianguliert.

**Beispiel B.21 (Fill-in, Fill-in-Graph)** Wir betrachten wieder den Graphen aus Abbildung B.3 und legen die folgende Ordnung auf den Knoten fest:

$$\alpha : C < D < B < E < A < G < F < H$$

In Abbildung B.5 ist diese Ordnung durch eine Nummerierung der Knoten angezeigt. Wir bestimmen den Fill-in  $\mathcal{F}(\alpha)$ :



**Abbildung B.5** Triangulierter Graph mit nummerierten Knoten

Es ist  $(B, D) \in \mathcal{F}(\alpha)$ , da die Knoten durch den Weg  $B, A, E, D$  miteinander verbunden sind und  $B, D < A, E$  sind. Ebenso ist  $(B, E) \in \mathcal{F}(\alpha)$ . Dies sind die einzigen beiden Kanten in  $\mathcal{F}(\alpha)$ . Abbildung B.5 zeigt den Fill-in-Graphen, wobei die Kanten des Fill-ins gestrichelt eingezeichnet sind. Der Fill-in-Graph ist offensichtlich trianguliert.  $\square$

**Selbsttestaufgabe B.22 (Fill-in)** Warum enthält im obigen Beispiel B.21 die Menge  $\mathcal{F}(\alpha)$  nicht die Kante  $(A, D)$ ?  $\blacksquare$

Fill-in und Fill-in-Graph eines ungerichteten Graphen hängen entscheidend von der gewählten Ordnung  $\alpha$  auf den Knoten ab. Dabei wäre eine Ordnung optimal, die bei einem bereits triangulierten Graphen zu einem leeren Fill-in führt, so dass der Graph mit seinem Fill-in-Graphen übereinstimmt. Eine solche Ordnung kann durch die sog. *maximum cardinality search* (*Maximalzahl-Suche*, *MCS*) gefunden werden: Einem beliebigen Knoten wird die Zahl 1 zugewiesen, und der jeweils nächste Knoten wird so ausgewählt, dass er zu der größtmöglichen Zahl bereits nummerierter Knoten adjazent ist. Gibt es dabei mehrere Möglichkeiten, so wird eine davon ausgewählt.

**Selbsttestaufgabe B.23 (maximum cardinality search)** Handelt es sich bei der Ordnung der Knoten in Abbildung B.5 um eine Ordnung nach dem MCS-Kriterium (bzgl. des nicht-triangulierten Graphen aus Abbildung B.3)?  $\blacksquare$

**Proposition B.24** Sei  $\mathcal{G} = \langle \mathbf{V}, \mathcal{E} \rangle$  ein ungerichteter Graph, und es sei  $\alpha$  eine Ordnung auf den Knoten  $\mathbf{V}$ . Ist  $\alpha$  aus einer MCS entstanden und ist  $\mathcal{G}$  bereits trianguliert, so ist  $\mathcal{G}(\alpha) = \mathcal{G}$ .

Verwendet man also zur Bestimmung einer Ordnung  $\alpha$  die MCS, so lässt sich das obige Triangulationsverfahren auch als Testverfahren benutzen.

Doch nicht nur für Triangulationen ist die *maximum cardinality search* wichtig. Auch im Hinblick auf die Cliques eines Graphen liefert dieses Verfahren nützliche Ordnungen. Jede Knoten-Ordnung induziert nämlich eine Cliques-Ordnung, indem man sich nach dem maximalen Knoten einer jeden Clique richtet.

**Beispiel B.25 (Cliques-Ordnung)** Der triangulierte Graph in Abbildung B.5 besitzt die folgenden Cliques und die angegebene, von der Knoten-Ordnung induzierte Cliques-Ordnung:

Clique	max. Knoten	Knoten-Nr.	Cliques-Ordnung
$\{C, D, B\}$	$B$	(3)	1
$\{B, D, E\}$	$E$	(4)	2
$\{B, E, A\}$	$A$	(5)	3
$\{E, F, G\}$	$F$	(7)	5
$\{E, D, G\}$	$G$	(6)	4
$\{D, H\}$	$H$	(8)	6

□

**Definition B.26 (Separation in ungerichteten Graphen)**  $\mathcal{G} = \langle \mathbf{V}, \mathcal{E} \rangle$  sei ein ungerichteter Graph, und  $\mathbf{A}, \mathbf{B}, \mathbf{C} \subseteq \mathbf{V}$  seien paarweise disjunkte Teilmengen von  $\mathbf{V}$ .  $\mathbf{C}$  separiert  $\mathbf{A}$  und  $\mathbf{B}$  in  $\mathcal{G}$ , geschrieben

$$\mathbf{A} \perp_{\mathcal{G}} \mathbf{B} \mid \mathbf{C}$$

wenn jeder Weg zwischen einem Knoten in  $\mathbf{A}$  und einem Knoten in  $\mathbf{B}$  mindestens einen Knoten aus  $\mathbf{C}$  enthält (vgl. Abbildung B.6). □

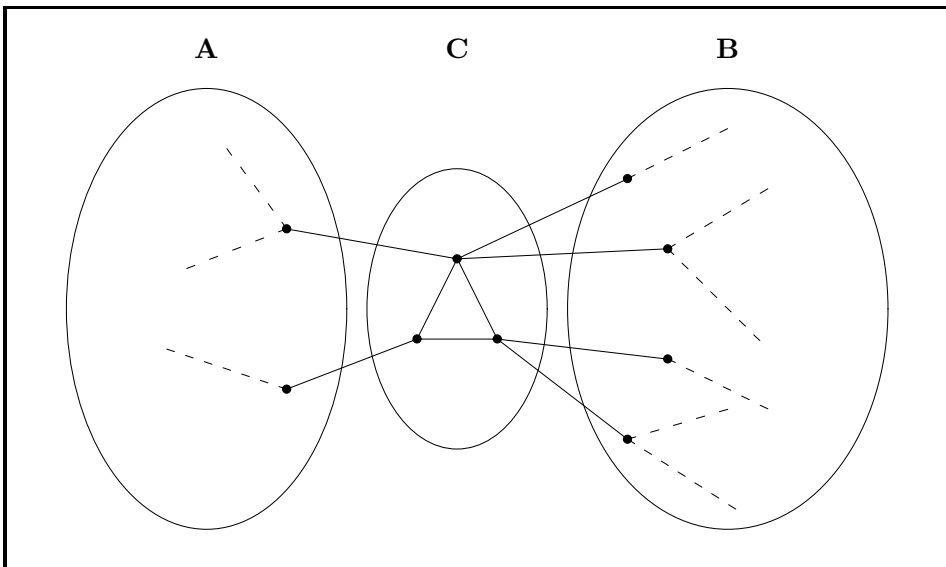


Abbildung B.6  $\mathbf{C}$  separiert  $\mathbf{A}$  und  $\mathbf{B}$

**Definition B.27 (Zerlegung)** Es sei  $\mathcal{G} = \langle \mathbf{V}, \mathcal{E} \rangle$  ein ungerichteter Graph, und sei  $\mathbf{V} = \mathbf{A} \cup \mathbf{B} \cup \mathbf{C}$  mit paarweise disjunkten Teilmengen  $(\mathbf{A}, \mathbf{B}, \mathbf{C})$ . Das Tripel  $(\mathbf{A}, \mathbf{B}, \mathbf{C})$  heißt *Zerlegung (decomposition) von  $\mathcal{G}$* , wenn gilt:

1.  $\mathbf{C}$  separiert  $\mathbf{A}$  und  $\mathbf{B}$  in  $\mathcal{G}$ ;
2. der durch  $\mathbf{C}$  beschriebene Teilgraph von  $\mathcal{G}$  ist vollständig.

Sind  $\mathbf{A}$  und  $\mathbf{B}$  beide nichtleer, so wird die Zerlegung  $(\mathbf{A}, \mathbf{B}, \mathbf{C})$  *eigentlich* genannt.  $\square$

**Definition B.28 (Zerlegbarer Graph)** Ein ungerichteter Graph  $\mathcal{G}$  heißt *zerlegbar (decomposable)*, wenn er entweder vollständig ist, oder wenn er eine eigentliche Zerlegung  $(\mathbf{A}, \mathbf{B}, \mathbf{C})$  besitzt, für die jeder der beiden durch  $\mathbf{A} \cup \mathbf{C}$  und  $\mathbf{B} \cup \mathbf{C}$  aufgespannten Teilgraphen von  $\mathcal{G}$  wieder zerlegbar ist.  $\square$

Diese rekursive Definition eines zerlegbaren Graphen ist wohldefiniert, da jeder der genannten Teilgraphen weniger Knoten hat als  $\mathcal{G}$ .

Zerlegbare Graphen lassen sich leicht charakterisieren:

**Proposition B.29** *Ein ungerichteter Graph  $\mathcal{G}$  ist genau dann zerlegbar, wenn er trianguliert ist.*

Ein Beweis dieses Satzes findet sich z. B. in [40], S. 51.

Auch bei gerichteten Graphen gibt es den Begriff der Separation, hier als *d-Separation* bezeichnet. Die ursprüngliche Definition von Pearl (s. [164], S. 117) baute explizit auf den Richtungen der Kanten auf. Eine äquivalente Definition führt die d-Separation auf die Separation im zugehörigen moralen Graphen zurück und benutzt daher die Richtungen nur implizit (vgl. [40], S. 72).

**Definition B.30 (d-Separation)** Sei  $\mathcal{G}^d = \langle \mathbf{V}, \mathcal{E}^d \rangle$  ein DAG, seien  $\mathbf{A}, \mathbf{B}, \mathbf{C}$  disjunkte Teilmengen von  $\mathbf{V}$ .  $\mathbf{C}$  *d-separiert*  $\mathbf{A}$  und  $\mathbf{B}$  in  $\mathcal{G}^d$ , wenn  $\mathbf{C}$  die beiden Mengen  $\mathbf{A}$  und  $\mathbf{B}$  in dem moralen Graphen, der durch die kleinste Vorfahrenmenge  $An(\mathbf{A} \cup \mathbf{B} \cup \mathbf{C})$  aufgespannt wird, separiert.  $\square$

**Beispiel B.31 (d-Separation)** Wir betrachten den DAG  $\mathcal{G}^d$  in Abbildung B.7(a). Es ist klar, dass  $A$  die einelementigen Mengen  $\{B\}$  und  $\{C\}$  d-separiert: Die kleinste Vorfahrenmenge von  $\{A, B, C\}$  ist  $\{A, B, C\}$  selbst, und der entsprechende morale Graph hat die einfache, in Abbildung B.7(b) gezeigte Form. Hier führt jeder Weg zwischen  $B$  und  $C$  über  $A$ .

Die Knotenmenge  $\{A, D\}$  jedoch d-separiert  $\{B\}$  und  $\{C\}$  nicht. Die kleinste Vorfahrenmenge von  $\{A, B, C, D\}$  ist wieder die Menge selbst. Bei der Moralisierung des Teilgraphen wird jedoch eine Kante zwischen  $B$  und  $C$  eingefügt, die die Separationseigenschaft von  $\{A, D\}$  untergräbt (s. Abbildung B.7(c)).  $\square$

**Selbsttestaufgabe B.32 (d-Separation)** Klären Sie die folgenden beiden Fragen zum DAG in Abbildung B.7(a):

1. d-separiert  $\{B, C\}$   $A$  und  $D$ ?
2. d-separiert  $\{A, E\}$   $B$  und  $C$ ? ■

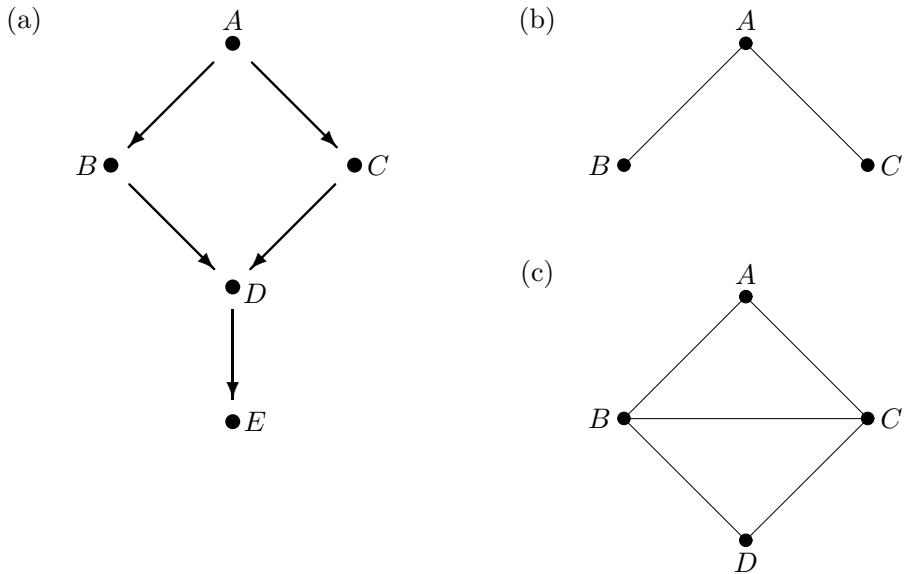


Abbildung B.7 DAG (a) und morale Graphen (b) und (c) zu Beispiel B.31

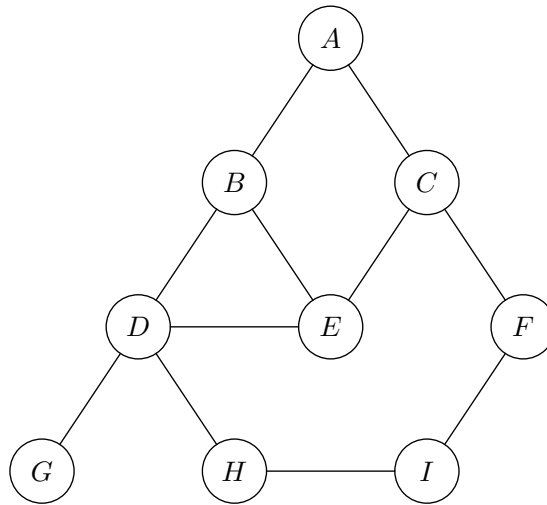


Abbildung B.8 Ausgangsgraph zu Selbsttestaufgabe B.33

**Selbsttestaufgabe B.33 (Triangulation)** Sei  $\mathcal{G}$  wie in Abbildung B.8 gegeben.

1. Geben Sie eine lineare Ordnung  $\alpha$  der Knoten von  $\mathcal{G}$  nach dem Maximalzahl-Kriterium an.
2. Triangulieren Sie  $\mathcal{G}$  durch Berechnen des Fill-ins  $\mathcal{F}(\alpha)$ , wobei  $\alpha$  die in Teil 1 bestimmte MCS-Ordnung ist. ■

**Selbsttestaufgabe B.34 (d-Separation)** Es sei  $P$  eine (positive) Wahrscheinlichkeitsverteilung über den Variablen  $A, B, C$ , in der  $A \perp\!\!\!\perp_P C \mid B$  gilt.

Geben Sie alle (zusammenhängenden) DAG's an, die diese bedingte Unabhängigkeit graphisch repräsentieren, d.h. in denen gilt:  $B$  d-separiert  $A$  und  $C$ . ■

### B.3 Die running intersection property RIP

**Definition B.35 (running intersection property, RIP)** Sei  $\mathcal{G} = \langle \mathbf{V}, \mathcal{E} \rangle$  ein ungerichteter Graph mit  $q$  Cliques. Eine (lineare) Ordnung  $(\mathbf{C}_1, \dots, \mathbf{C}_q)$  dieser Cliques hat die *running intersection property, RIP (fortlaufende Schnitteigenschaft)*, wenn es für jedes  $i \in \{2, \dots, q\}$  ein  $j < i$  gibt, so dass

$$\mathbf{C}_i \cap (\mathbf{C}_1 \cup \dots \cup \mathbf{C}_{i-1}) \subseteq \mathbf{C}_j \quad (\text{B.1})$$

gilt. □

**Selbsttestaufgabe B.36 (running intersection property)** Prüfen Sie, ob die Ordnung  $(\mathbf{C}_1, \mathbf{C}_2, \dots, \mathbf{C}_7)$  der Cliques in Beispiel B.14 die RIP besitzt. ■

Nicht in jedem Fall kann eine Cliques-Ordnung mit der *running intersection property* gefunden werden. Bei triangulierten Graphen ist das jedoch immer möglich, wobei eine passende Cliques-Ordnung durch eine MCS-Ordnung bestimmt werden kann.

**Theorem B.37 (MCS und RIP)** Sei  $\mathcal{G} = \langle \mathbf{V}, \mathcal{E} \rangle$  ein triangulierter ungerichteter Graph. Sei  $\alpha$  eine Ordnung auf  $\mathbf{V}$ , die dem MCS-Kriterium folgt, und die Cliques von  $\mathcal{G}$  seien gemäß ihrer maximalen Knoten geordnet. Dann besitzt diese Cliques-Ordnung die *running intersection property*. □

Ein Beweis dieses Theorems findet sich z. B. in [157].

Die *running intersection property* ermöglicht die Anordnung der Cliques eines Graphen in einer Baumstruktur, dem sog. *Cliques- oder Verbindungsbaum (junction tree)*, dessen Knoten gerade die Cliques sind:

Sei also  $\mathbf{C}_1, \dots, \mathbf{C}_q$  eine RIP-Ordnung der Cliques eines triangulierten Graphen  $\mathcal{G}$ . Für  $i \in \{2, \dots, q\}$  definiere die Menge

$$\mathbf{S}_i := \mathbf{C}_i \cap (\mathbf{C}_1 \cup \dots \cup \mathbf{C}_{i-1})$$

Wegen der *running intersection property* gibt es zu jedem solchen  $i$  ein  $j < i$  so, dass  $\mathbf{S}_i \subseteq \mathbf{C}_j$  ist; gibt es mehrere solcher  $j$ , so wähle man eines,  $j(i)$ , aus.  $\mathbf{C}_{j(i)}$  wird dann als Elternclique zu  $\mathbf{C}_i$  bestimmt. Auf diese Weise entsteht ein Baum mit Knotenmenge  $\{\mathbf{C}_1, \dots, \mathbf{C}_q\}$ . Die Mengen  $\mathbf{S}_i$  sind Separatoren des zerlegbaren Graphen  $\mathcal{G}$  (im Sinne von Definition B.26) und werden auch als *Separatoren* des Cliquesbaumes bezeichnet. Häufig notiert man sie als Label an den Kanten des Cliquesbaumes.



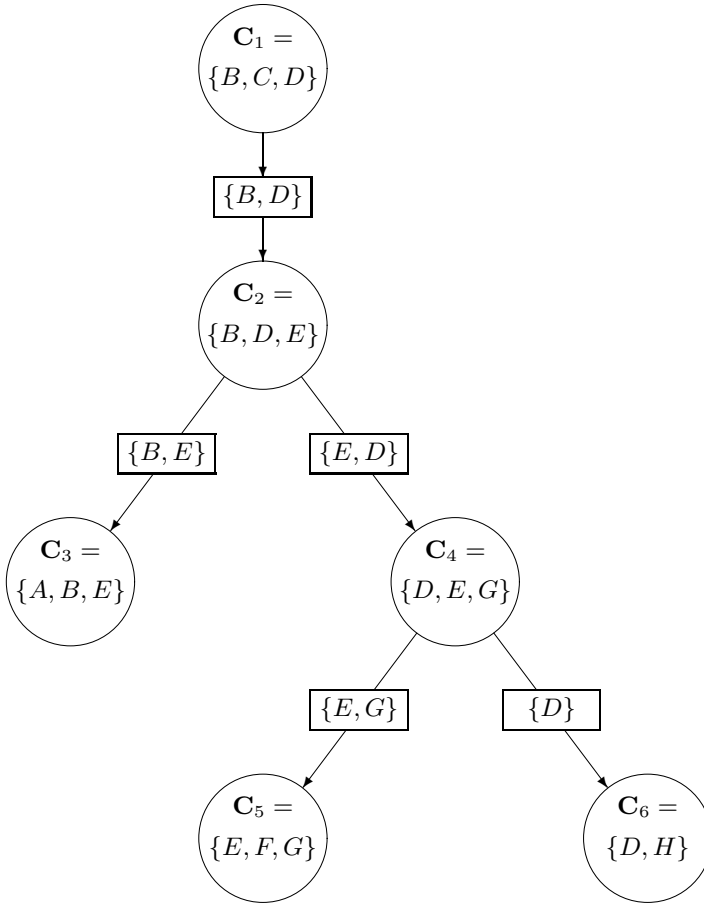


Abbildung B.9 Cliquograph mit Separatoren zu Beispiel B.38

**Beispiel B.38 (Cliquenbaum)** Nach Theorem B.37 besitzt die in Beispiel B.25 angegebene Cliquen-Ordnung die RIP. Wir konstruieren dazu einen passenden Cliquenbaum. In der folgenden Tabelle sind Cliques, Separatoren und mögliche Elterncliques angegeben, und Abbildung B.9 zeigt den fertigen Cliquenbaum.

Cliques	$S_i$	Elternclique
$C_1 = \{B, C, D\}$	--	--
$C_2 = \{B, D, E\}$	$\{B, D\}$	$C_1$
$C_3 = \{A, B, E\}$	$\{B, E\}$	$C_2$
$C_4 = \{D, E, G\}$	$\{D, E\}$	$C_2$
$C_5 = \{E, F, G\}$	$\{E, G\}$	$C_4$
$C_6 = \{D, H\}$	$\{D\}$	$C_4$

□

**Selbsttestaufgabe B.39 (Cliquenbaum)** Geben Sie zum Beispiel B.38 noch einen anderen möglichen Cliquenbaum an. ■

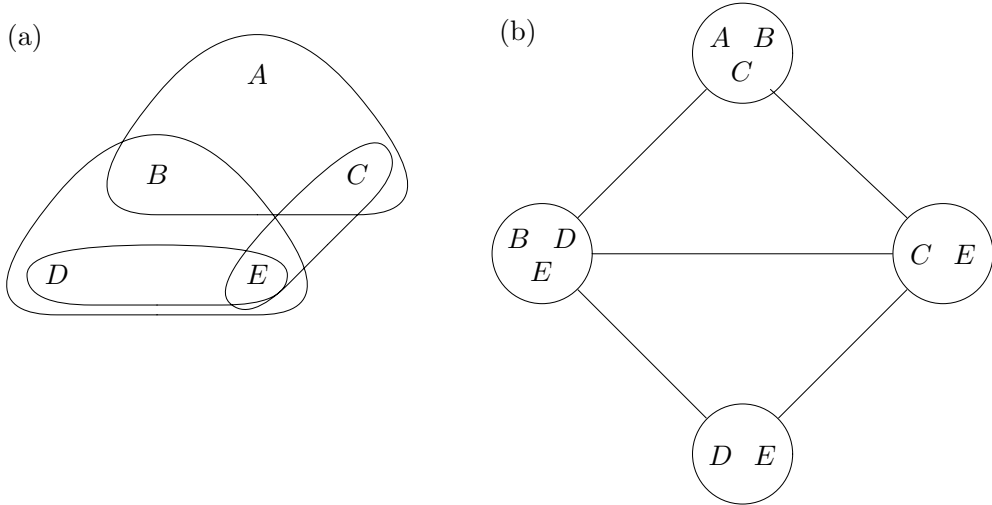


Abbildung B.10 Hypergraph (a) und zugehöriger Verbindungsgraph (b)

## B.4 Hypergraphen

Hypergraphen sind verallgemeinerte Graphen, deren Kanten mehr als zwei Knoten verbinden können.

**Definition B.40 (Hypergraph, Hyperkante)** Sei  $\mathbf{V}$  eine (endliche) Menge von Knoten und  $\mathcal{E} = \{\mathbf{E}_1, \dots, \mathbf{E}_m\}, \emptyset \neq \mathbf{E}_i \subseteq \mathbf{V}, 1 \leq i \leq m$ , eine Menge von Teilmengen<sup>3</sup> von  $\mathbf{V}$  mit  $\mathbf{V} = \cup_{i=1}^m \mathbf{E}_i$ . Dann heißt  $\mathcal{H} = \langle \mathbf{V}, \mathcal{E} \rangle$  *Hypergraph*. Die Elemente von  $\mathcal{E}$  werden *Hyperkanten* genannt.

Ein Hypergraph heißt *reduziert*, wenn keine Hyperkante echt in einer anderen Hyperkante enthalten ist. □

**Beispiel B.41 (Hypergraph)** Abbildung B.10(a) zeigt einen Hypergraphen mit der Knotenmenge  $\mathbf{V} = \{A, B, C, D, E\}$  und den Hyperkanten  $\mathcal{E} = \{\{A, B, C\}, \{B, D, E\}, \{C, E\}, \{D, E\}\}$ . Der Hypergraph ist nicht reduziert, da die Hyperkante  $\{D, E\}$  in der Hyperkante  $\{B, D, E\}$  enthalten ist. □

Hyperkanten, die einen nichtleeren Schnitt besitzen, schaffen Verbindungen zwischen den beteiligten Knoten:

**Definition B.42 (Verbindungsgraph)** Sei  $\mathcal{H} = \langle \mathbf{V}, \mathcal{E} \rangle$  ein Hypergraph. Der  $\mathcal{H}$  zugeordnete *Verbindungsgraph* (*junction graph*)  $J(\mathcal{H})$  ist ein ungerichteter Graph mit den Hyperkanten  $\mathcal{E}$  als Knoten. Zwei solcher Knoten sind genau dann durch eine Kante verbunden, wenn die zugehörigen Hyperkanten nichtleeren Schnitt besitzen. □

---

<sup>3</sup> Auch die Menge der Hyperkanten wird – wie die Menge der Kanten bei Graphen – mit  $\mathcal{E}$  bezeichnet; diese Bezeichnungsweise ist konsistent, da sich ein ungerichteter Graph (ohne isolierte Knoten) auch als Hypergraph mit  $|\mathbf{E}| = 2$  für alle (Hyper)Kanten  $\mathbf{E}$  auffassen lässt.

Der Verbindungsgraph zu dem Hypergraphen aus Beispiel B.41 ist in Abbildung B.10(b) zu sehen.

Wie bei den Cliquesbäumen spielt auch bei allgemeinen Hypergraphen die Baumeigenschaft eine wichtige Rolle für gute Berechnungseigenschaften.

**Definition B.43 (Hyperbaum)** Ein Hypergraph  $\mathcal{H} = \langle \mathbf{V}, \mathcal{E} \rangle$  heißt *azyklisch* oder *Hyperbaum*, wenn es eine (lineare) Anordnung seiner Hyperkanten gibt, die die RIP (s. Definition B.35) besitzt.  $\square$

Bei der Überprüfung der Baumeigenschaft eines Hypergraphen  $\mathcal{H} = \langle \mathbf{V}, \mathcal{E} \rangle$  kann man sich auf Anordnungen beschränken, die durch eine Variante des *maximum cardinality search* entstanden sind:

- Man ordnet einer beliebigen Hyperkante  $\mathbf{E} \in \mathcal{E}$  den Index 1 zu und nummeriert die Knoten in  $\mathbf{E}$  in beliebiger, aufsteigender Reihenfolge.
- Als nächste Hyperkante wählt man nun sukzessive jeweils eine derjenigen Hyperkanten aus, die eine Maximalzahl bereits nummerierter Knoten enthält. Die noch nicht nummerierten Knoten der neuen Hyperkante werden weiter in aufsteigender Reihenfolge nummeriert.

Es gilt der folgende Satz von Tarjan und Yannakakis [213]:

**Proposition B.44** *Ein Hypergraph  $\mathcal{H} = \langle \mathbf{V}, \mathcal{E} \rangle$  ist genau dann ein Hyperbaum, wenn jede MCS-Nummerierung der Hyperkanten von  $\mathcal{H}$  die RIP besitzt.*

**Beispiel B.45 (Hyperbaum)** Wir wenden die *maximum cardinality search* auf den Hypergraphen aus Beispiel B.41 an und erhalten (z. B.) die folgende Anordnung der Hyperkanten:

$$\mathbf{E}_1 = \{A, B, C\}, \mathbf{E}_2 = \{B, D, E\}, \mathbf{E}_3 = \{D, E\}, \mathbf{E}_4 = \{C, E\}$$

wobei die Knoten dem Alphabet entsprechend geordnet werden:  $A < B < C < D < E$ . Diese Ordnung besitzt nicht die RIP, da

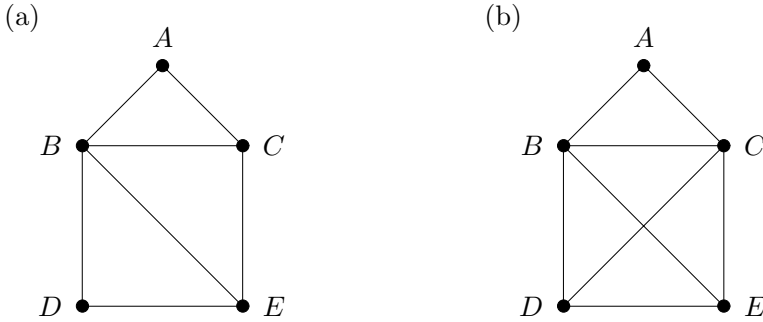
$$\mathbf{E}_4 \cap (\mathbf{E}_1 \cup \mathbf{E}_2 \cup \mathbf{E}_3) = \{C, E\}$$

in keiner der Hyperkanten  $\mathbf{E}_1, \mathbf{E}_2, \mathbf{E}_3$  enthalten ist. Der Hypergraph in Abbildung B.10(a) ist also kein Hyperbaum.  $\square$

Aus einem beliebigen Hypergraphen  $\mathcal{H} = \langle \mathbf{V}, \mathcal{E} \rangle$  kann man durch eine Fill-in-Technik einen überdeckenden Hyperbaum  $\mathcal{H}' = \langle \mathbf{V}, \mathcal{E}' \rangle$  gewinnen. Überdeckend bedeutet, dass jede Hyperkante  $\mathbf{E} \in \mathcal{E}$  Teilmenge einer Hyperkante  $\mathbf{E}' \in \mathcal{E}'$  ist. Zu diesem Zweck betrachtet man den *Schnittgraphen (cut graph)*  $\mathcal{H}_s = \langle \mathbf{V}, \mathcal{E}_s \rangle$  von  $\mathcal{H}$  mit

$$(v, w) \in \mathcal{E}_s \quad \text{gdw.} \quad \exists \mathbf{E} \in \mathcal{E} \text{ mit } v, w \in \mathbf{E}$$

Zwei Knoten aus  $V$  werden im Schnittgraphen  $\mathcal{H}_s$  also genau dann durch eine (normale) Kante verbunden, wenn es eine Hyperkante von  $\mathcal{H}$  gibt, in der beide liegen.



**Abbildung B.11** Schnittgraph  $\mathcal{H}_s$  (a) und Fill-in-Graph (b) zum Hypergraphen in Abbildung B.10(a)

Der Schnittgraph  $\mathcal{H}_s$  wird nun durch Einfügen von Kanten aufgefüllt. Hierbei geht man von einer MCS-Ordnung bzw. -Nummerierung der Knoten aus und verbindet die Menge  $\{v_i \mid (v_i, v_j) \in \mathcal{E}_s, i < j\}$  aller “kleineren” Nachbarn eines jeden Knoten  $v_j$  zu einem vollständigen Graphen. Sind  $\mathbf{C}_1, \dots, \mathbf{C}_q$  die Cliques des Fill-in-Graphen von  $\mathcal{H}_s$ , so ist  $\mathcal{H}' = \langle \mathbf{V}, \{\mathbf{C}_1, \dots, \mathbf{C}_q\} \rangle$  ein überdeckender Hyperbaum zu  $\mathcal{H}$ . Der Hyperbaum  $\mathcal{H}'$  schließlich bietet eine gute Grundlage für effiziente Berechnungen.

**Beispiel B.46 (Überdeckender Hyperbaum)** Wir setzen die Beispiele B.41 und B.45 fort und gehen von der alphabetischen Ordnung der Knoten  $A, B, C, D, E$  aus. Der Schnittgraph  $\mathcal{H}_s$  von  $\mathcal{H} = \langle \{A, B, C, D, E\}, \{\{A, B, C\}, \{B, D, E\}, \{C, E\}, \{D, E\}\} \rangle$  ist in Abbildung B.11(a) zu sehen.

Die kleineren Nachbarn der Knoten sind

Knoten	kleinere Nachbarn
A	--
B	A
C	A, B
D	B
E	B, C, D

Um den Schnittgraphen zu vervollständigen, muss also noch die Kante  $(C, D)$  eingefügt werden (s. Abbildung B.11(b)). Ein überdeckender Hyperbaum zu  $\mathcal{H}$  ist dann  $\mathcal{H}' = \langle \mathbf{V}, \{\{A, B, C\}, \{B, C, D, E\}\} \rangle$ . □

**Selbsttestaufgabe B.47 (überdeckender Hyperbaum)** Geben Sie eine Ordnung der Knoten des Hypergraphen  $\mathcal{H}$  aus Beispiel B.41, für die der Schnittgraph  $\mathcal{H}_s$  (s. Abbildung B.11(a)) nicht mehr vervollständigt werden muss. Wie sieht der zugehörige überdeckende Hyperbaum aus? ■

---

## Literaturverzeichnis

- [1] A. Aamodt and E. Plaza. Case-based reasoning: Foundational issues, methodological variations, and system approaches. *AICom – Artificial Intelligence Communications*, 7(1), March 1996.
- [2] R. Agrawal, H. Mannila, R. Srikant, H. Toivonen, and A.I. Verkamo. Fast discovery of association rules. In U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, editors, *Advances in knowledge discovery and data mining*, pages 307–328. MIT Press, Cambridge, Mass., 1996.
- [3] L.C. Aiello and F. Massacci. Verifying security protocols as planning in logic programming. *ACM Trans. Comput. Logic*, 2(4):542–580, 2001.
- [4] G. Antoniou. *Nonmonotonic reasoning*. MIT Press, Cambridge, Mass., 1997.
- [5] K. Apt, H. Blair, and A. Walker. Towards a theory of declarative knowledge. In J. Minker, editor, *Foundations of Deductive Databases and Logic Programming*, pages 89–148. Morgan Kaufmann, San Mateo, CA, 1988.
- [6] K. R. Apt and M. H. van Emden. Contributions to the theory of logic programming. *Journal of the ACM*, 29(3):841–862, 1982.
- [7] M. Asada and H. Kitano, editors. *RoboCup-98: Robot Soccer World Cup II*. Springer-Verlag, 1999.
- [8] C. Baral and M. Gelfond. Logic programming and knowledge representation. *J. Logic Programming*, 19,20:73–148, 1994.
- [9] M. Beetz, T. Arbuckle, T. Belker, M. Bennewitz, A.B. Cremers, D. Hähnel, and D. Schulz. Enabling autonomous robots to perform complex tasks. *Künstliche Intelligenz*, (4/00):5–10, 2000.
- [10] C. Beierle, O. Dusso, and G. Kern-Isberner. Using answer set programming for a decision support system. In C. Baral, G. Greco, N. Leone, and G. Terracina, editors, *8th International Conference on Logic Programming and Non Monotonic Reasoning (LPNMR 2005)*, volume 3662 of *LNAI*. Springer-Verlag, 2005.
- [11] C. Beierle, U. Hedtstück, U. Pletat, P. H. Schmitt, and J. Siekmann. An order-sorted logic for knowledge representation systems. *Artificial Intelligence*, 55(2–3):149–191, 1992.
- [12] R. Bergmann, K.-D. Althoff, S. Breen, M. Göker, M. Manago, R. Traphöner, and S. Wess, editors. *Developing Industrial Case-Based Reasoning Applications*. Number 1612 in *LNAI*. Springer, Berlin Heidelberg New York, 2nd edition, 2003.
- [13] R. Bergmann and W. Wilke. Towards a new formal model of transformational adaptation in case-based reasoning. In H. Prade, editor, *Proc. 13th European Conference on Artificial Intelligence, ECAI'98*, pages 53–57. John Wiley & Sons, 1998.
- [14] W. Bibel. *Deduktion*. Oldenbourg Verlag, München, Wien, 1992.
- [15] W. Bibel, S. Hölldobler, and T. Schaub. *Wissensrepräsentation und Inferenz: Eine grundlegende Einführung*. Vieweg & Sohn Verlagsgesellschaft mbH, Braunschweig/Wiesbaden, 1993.
- [16] K. H. Bläsius and H.-J. Bürckert, editors. *Deduktionssysteme – Automatisierung des logischen Denkens*. Oldenbourg Verlag, München, 1992.
- [17] T. Bollinger. Assoziationsregeln – Analyse eines Data Mining Verfahrens. *Informatik-Spektrum*, 19:257–261, 1996.

- [18] R. J. Brachman and J. G. Schmolze. An overview of the KL-ONE knowledge representation system. *Cognitive Science*, 9(2):171–216, April 1985.
- [19] R.J. Brachman and H.J. Levesque. *Knowledge representation and reasoning*. Morgan Kaufmann Publishers, San Francisco, California, 2004.
- [20] M. E. Bratman. *Intentions, Plans, and Practical Reason*. Harvard University Press, Cambridge, MA, 1987.
- [21] M.E. Bratman, D.J. Israel, and M.E. Pollack. Plans and resource-bounded practical reasoning. *Computational Intelligence*, 4:349–355, 1988.
- [22] G. Brewka. Reasoning about priorities in default logic. In *Proceedings of the 12th National Conference on Artificial Intelligence (AAAI'94)*, pages 940–945. AAAI/MIT Press, 1994.
- [23] G. Brewka and T. Eiter. Preferred answer sets for extended logic programs. *Artificial Intelligence*, 109(1-2):297–356, 1999.
- [24] G. Brewka, D. Makinson, and K. Schlechta. Cumulative inference relations for JTMS and logic programming. In J. Dix, K.P. Jantke, and P.H. Schmitt, editors, *Nonmonotonic and inductive logic, Proc. 1st International Workshop, Karlsruhe, 1990*, pages 1–12, Berlin, 1991. Springer.
- [25] R. A. Brooks. Intelligence without reason. In *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence (IJCAI-91)*, pages 569–595, Sydney, Australia, 1991.
- [26] F. Buccafurri, N. Leone, and P. Rullo. Strong and weak constraints in disjunctive datalog. In *Proceedings 4th International Conference on Logic Programming and Non-Monotonic Reasoning*, volume 1265 of *Lecture Notes in Computer Science*, pages 2–17, Berlin, 1997. Springer.
- [27] B.G. Buchanan and E.H. Shortliffe. *Rule-based expert systems. The MYCIN experiments of the Stanford Heuristic Programming Project*. Addison-Wesley, Reading, MA, 1984.
- [28] B.G. Buchanan, G.L. Sutherland, and E.A. Feigenbaum. Heuristic DENDRAL: A program for generating explanatory hypotheses in organic chemistry. In B. Meltzer, D. Michie, and M. Swann, editors, *Machine Intelligence 4*, pages 209–254. Edinburgh University Press, Edinburgh, 1969.
- [29] W. Buntine. A guide to the literature on learning probabilistic networks from data. *IEEE Transactions on Knowledge and Data Engineering*, 8(2):195–210, 1996.
- [30] W. Burgard, A. B. Cremers, D. Fox, D. Hähnel, G. Lakemeyer, D. Schulz, W. Steiner, and S. Thrun. The interactive museum tour-guide robot. In *Proc. 15th National Conference on Artificial Intelligence (AAAI'98)*, 1998.
- [31] W. Burgard, A. B. Cremers, D. Fox, D. Hähnel, G. Lakemeyer, D. Schulz, W. Steiner, and S. Thrun. Experiences with an interactive museum tour-guide robot. *Artificial Intelligence*, 114(1-2), 2000.
- [32] J.G. Carbonell, R.S. Michalski, and T.M. Mitchell. An overview of machine learning. In R.S. Michalski, J.G. Carbonell, and T.M. Mitchell, editors, *Machine Learning: An Artificial Intelligence Approach*. Tioga Publishing Company, Palo Alto, 1983.
- [33] K. M. Carley and L. Gasser. Computational organization theory. In G. Weiss, editor, *Multiagent Systems - A Modern Approach to Distributed Artificial Intelligence*, pages 299–330. The MIT Press, Cambridge, Massachusetts, 1999.
- [34] E. Castillo, J. M. Gutierrez, and A. S. Hadi. *Expert systems and probabilistic network models*. Springer, 1997.

- [35] P. Cholewinski, W. Marek, and M. Truszczynski. Default reasoning system DeReS. In *Proceedings International Conference on Principles of Knowledge Representation and Reasoning*, pages 518–528, San Mateo, CA, 1996. Morgan Kaufman.
- [36] K. Clark. Negation as failure. In H. Gallaire and J. Minker, editors, *Logic and Data Bases*, pages 293–322. Plenum Press, New York, 1978.
- [37] W. F. Clocksin and C. S. Mellish. *Programming in Prolog*. Springer-Verlag, Berlin, Heidelberg, New York, 4. edition, 1994.
- [38] P. R. Cohen and H. J. Levesque. Intention is choice with commitment. *Artificial Intelligence*, 42:213–261, 1990.
- [39] G.F. Cooper and E. Herskovits. A bayesian method for the induction of probabilistic networks from data. *Machine learning*, 9:309–347, 1992.
- [40] R.G. Cowell, A.P. Dawid, S.L. Lauritzen, and D.J. Spiegelhalter. *Probabilistic networks and expert systems*. Springer, New York Berlin Heidelberg, 1999.
- [41] I. Csiszár. I-divergence geometry of probability distributions and minimization problems. *Ann. Prob.*, 3:146–158, 1975.
- [42] P. Cunningham, R. Bergmann, S. Schmitt, R. Traphöner, S. Breen, and B. Smyth. WEBSELL: Intelligent sales assistants for the World Wide Web. *Künstliche Intelligenz*, 1/01:28–32, 2001.
- [43] J. de Kleer. An assumption-based TMS. *Artificial Intelligence*, 28:127–162, 1986.
- [44] J. de Kleer. Extending the ATMS. *Artificial Intelligence*, 28:163–196, 1986.
- [45] J. de Kleer. Problem solving with the ATMS. *Artificial Intelligence*, 28:197–224, 1986.
- [46] A.P. Dempster. Upper and lower probabilities induced by a multivalued mapping. *Ann. Math. Stat.*, 38:325–339, 1967.
- [47] M. d’Inverno and M. Luck. *Understanding Agent Systems*. Springer Series on Agent Technology. Springer-Verlag, second edition, 2003.
- [48] J. Dix and J. Leite, editors. *Computational Logic in Multi-Agent Systems: 4th International Workshop, CLIMA IV*. Number 3259 in LNCS. Springer, Berlin Heidelberg New York, 2004.
- [49] J. Doyle. A truth maintenance system. *Artificial Intelligence*, 12:231–272, 1979.
- [50] D. Dubois, J. Lang, and H. Prade. Possibilistic logic. In D.M. Gabbay, C.H. Hogger, and J.A. Robinson, editors, *Handbook of Logic in Artificial Intelligence and Logic Programming*, volume 3. Oxford University Press, 1994.
- [51] D. Dubois and H. Prade. Belief change and possibility theory. In P. Gärdenfors, editor, *Belief revision*, pages 142–182. Cambridge University Press, 1992.
- [52] T. Eiter, W. Faber, N. Leone, and G. Pfeifer. Declarative problem solving using the DLV system. In J. Minker, editor, *Logic-Based Artificial Intelligence*, pages 79–103. Kluwer Academic Publishers, Dordrecht, 2000.
- [53] T. Eiter, G. Gottlob, and H. Mannila. Disjunctive datalog. *ACM Trans. Database Systems*, 22(3):364–418, 1997.
- [54] C. Elkan. A rational reconstruction of nonmonotonic truth maintenance systems. *Artificial Intelligence*, 43:219–234, 1990.
- [55] W. Faber, N. Leone, and G. Pfeifer. Experimenting with heuristics for answer set programming. In *Proceedings International Joint Conference on Artificial Intelligence*, pages 635–640, San Mateo, CA, 2001. Morgan Kaufman.

- [56] R. Fagin, J.Y. Halpern, and N. Megiddo. A logic for reasoning about probabilities. *Information and Computation*, 87:78–128, 1990.
- [57] U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy. *Advances in knowledge discovery and data mining*. MIT Press, Cambridge, Mass., 1996.
- [58] U. Fayyad and R. Uthurusamy. Evolving data mining into solutions for insights. *Communications of the ACM*, 45(8):28–61, 2002.
- [59] U. Fayyad, R. Uthurusamy, et al. Data mining and knowledge discovery in databases. *Communications of the ACM*, 39(11):24–64, 1996.
- [60] R. E. Fikes, P. E. Hart, and N. J. Nilsson. Learning and executing generalized robot plans. *Artificial Intelligence*, 3(4):251–288, 1972.
- [61] R. E. Fikes and N. J. Nilsson. STRIPS: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence*, 2(3-4):189–208, 1971.
- [62] R. E. Fikes and N. J. Nilsson. STRIPS, a retrospective. *Artificial Intelligence*, 59:227–232, 1993.
- [63] V.G. Fischer and M. Schramm. Tabl – a tool for efficient compilation of probabilistic constraints. Technical Report TUM-19636, Technische Universität München, 1996.
- [64] G. Friedrich and M. Stumptner. Einführung. In G. Gottlob, Th. Frühwirth, and W. Horn, editors, *Expertensysteme*, Springer Angewandte Informatik, pages 1–19. Springer-Verlag, 1990.
- [65] D. Gabbay. Theoretical foundations for nonmonotonic reasoning in expert systems. In K. Apt, editor, *Logics and models of concurrent systems*. Springer, Berlin, 1985.
- [66] P. Gärdenfors and H. Rott. Belief revision. In D.M. Gabbay, C.H. Hogger, and J.A. Robinson, editors, *Handbook of Logic in Artificial Intelligence and Logic Programming*, pages 35–132. Oxford University Press, 1994.
- [67] L. Gasser. Social conceptions of knowledge and action: DAI foundations and open system semantics. *Artificial Intelligence*, 47:107–138, 1991.
- [68] L. Gasser and M. N. Huhns, editors. *Distributed Artificial Intelligence, Volume II*. Pitman Publishing, London, 1989.
- [69] D. Geiger and D. Heckerman. Knowledge representation and inference in similarity networks and bayesian multinets. *Artificial Intelligence*, 82:45–74, 1996.
- [70] M. Gelfond and N. Leone. Logic programming and knowledge representation – the A-prolog perspective. *Artificial Intelligence*, 138:3–38, 2002.
- [71] M. Gelfond and V. Lifschitz. The stable model semantics for logic programming. In *Logic Programming: Proceedings Fifth International Conference and Symposium*, pages 1070–1080, Cambridge, Mass., 1988. MIT Press.
- [72] M. Gelfond and V. Lifschitz. Classical negation in logic programs and disjunctive databases. *New Generation Comput.*, pages 365–387, 1991.
- [73] M.R. Genesereth and N.J. Nilsson. *Logical foundations of Artificial Intelligence*. Morgan Kaufmann, Palo Alto, Ca., 1987.
- [74] M. P. Georgeff and A. S. Rao. The semantics of intention maintenance for rational agents. In *Proceedings of the International Joint Conference of Artificial Intelligence (IJCAI)*, pages 704–710, 1995.
- [75] M.P. Georgeff and A.L. Lansky. Reactive reasoning and planning. In *Proceedings of the Sixth National Conference on Artificial Intelligence (AAAI-87)*, pages 677–682, Seattle, WA, 1987.



- [76] M. Ghallab, D. Nau, and P. Traverso. *Automated planning – theory and practice*. Elsevier, San Francisco, 2004.
- [77] R. Goldman and E. Charniak. A language for construction of belief networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(3):196–208, 1993.
- [78] G. Görz, C.-R. Rollinger, and J. Schneeberger, editors. *Handbuch der Künstlichen Intelligenz*. Oldenbourg, 4., korr. Auflage, 2003.
- [79] H. Grosskreutz and G. Lakemeyer. Towards more realistic logic-based robot controllers in the GOLOG framework. *Künstliche Intelligenz*, (4/00):11–15, 2000.
- [80] R. H. Güting and S. Dieker. *Datenstrukturen und Algorithmen*. Leitfäden der Informatik. Teubner-Verlag, Stuttgart, 3. Auflage, 2004.
- [81] J.Y. Halpern. *Reasoning about uncertainty*. MIT Press, Cambridge, Mass., 2003.
- [82] C. et al. Hartshorn, editor. *Collected Papers of C. Sanders Peirce, Band 2*. Harvard University Press, Cambridge, 1931.
- [83] D. Heckerman. A tutorial on learning bayesian networks. Technical report, Microsoft Research, Advanced Technology Division, 1995. Technical Report MSR-TR-95-06.
- [84] D. Heckerman. Bayesian networks for knowledge discovery. In U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, editors, *Advances in knowledge discovery and data mining*. MIT Press, Cambridge, Mass., 1996.
- [85] D. Heckerman, J. Breese, and K. Rommelse. Decision-theoretic troubleshooting. *Communications of the ACM*, 38(3):49–56, 1995.
- [86] W. Heise and P. Quattrocchi. *Informations- und Codierungstheorie*. Springer, Berlin Heidelberg New York, 1995.
- [87] K. Heljanko. Using logic programs with stable model semantics to solve deadlock and reachability problems for 1-safe Petri nets. In *Proceedings 5th International Conference on Tools and Algorithms for the Construction and Analysis of Systems*, pages 240–254, Amsterdam, 1999.
- [88] J. Hertzberg. *Planen – Einführung in die Planerstellungsmethoden der Künstlichen Intelligenz*. Reihe Informatik. BI Wissenschaftsverlag, Mannheim-Wien-Zürich, 1989.
- [89] E. Horvitz and B. Barry. Display of information for time-critical decision making. In Besnard and Hanks, editors, *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, pages 296–305, San Francisco, CA., 1995. Morgan Kaufmann.
- [90] Y.-C. Huang, B. Selman, and H. Kautz. Control knowledge in planning: benefits and tradeoffs. In *Proc. AAAI-99*, pages 511–517, Orlando, FL, 1999.
- [91] M. N. Huhns and D. M. Bridgeland. Multiagent truth maintenance. *IEEE Transactions on Systems, Man, and Cybernetics*, 21(6):1437–1445, December 1991.
- [92] M. N. Huhns and L. M. Stephens. Multiagent systems and the societies of agents. In G. Weiss, editor, *Multiagent Systems - A Modern Approach to Distributed Artificial Intelligence*, pages 79–120. The MIT Press, Cambridge, Massachusetts, 1999.
- [93] E.T. Jaynes. *Papers on Probability, Statistics and Statistical Physics*. D. Reidel Publishing Company, Dordrecht, Holland, 1983.
- [94] N. R. Jennings. Commitments and conventions: The foundation of coordination in multi-agent systems. *The Knowledge Engineering Review*, 2(3):223–250, 1993.
- [95] N. R. Jennings. Coordination techniques for distributed artificial intelligence. In G. M. P. O’Hare and N. R. Jennings, editors, *Foundations of Distributed Artificial Intelligence*, pages 187–210. John Wiley & Sons, Inc., New York, 1996.

- [96] F.V. Jensen. *Introduction to Bayesian networks*. UCL Press, London, 1996.
- [97] F.V. Jensen, S.L. Lauritzen, and K.G. Olesen. Bayesian updating in recursive graphical models by logical computations. *Computational Statistical Quarterly*, 4:269–282, 1990.
- [98] U. Junker and K. Konolige. Computing the extensions of autoepistemic and default logics with a truth maintenance system. In *Proceedings 8th National Conference on Artificial Intelligence, AAAI'90*, pages 278–283, Boston, Mass., 1990.
- [99] H. Katsuno and A.O. Mendelzon. On the difference between updating a knowledge base and revising it. In *Proceedings Second International Conference on Principles of Knowledge Representation and Reasoning, KR'91*, pages 387–394, San Mateo, Ca., 1991. Morgan Kaufmann.
- [100] G. Kern-Isberner. Characterizing the principle of minimum cross-entropy within a conditional-logical framework. *Artificial Intelligence*, 98:169–208, 1998.
- [101] G. Kern-Isberner. *A unifying framework for symbolic and numerical approaches to nonmonotonic reasoning and belief revision*. Fachbereich Informatik der FernUniversität Hagen, 1999. Habilitationsschrift.
- [102] G. Kern-Isberner. *Conditionals in nonmonotonic reasoning and belief revision*. Springer, Lecture Notes in Artificial Intelligence LNAI 2087, 2001.
- [103] D. Kinney and M. Georgeff. Commitment and effectiveness of situated agents. In *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence (IJCAI-91)*, pages 82–88, Sydney, Australia, 1991.
- [104] H. Kitano et al. Special issue on RoboSoccer. *Artificial Intelligence*, 110, 1999.
- [105] H. Kitano and H. Shimazu. The experience-sharing architecture: A case study in corporate-wide case-based software quality control. In D.B. Leake, editor, *Case-based reasoning*, pages 235–268. AAAI Press, Menlo Park, Ca., 1996.
- [106] Y. Kodratoff and P. Langley, editors. *Proc. ECML-93 Workshop on Real-World Applications of Machine Learning*. Wien, 1993.
- [107] Y. Kodratoff and R.S. Michalski, editors. *Machine Learning: An Artificial Intelligence Approach*, volume III. Morgan Kaufmann, San Mateo, Calif., 1990.
- [108] J. Kolodner. *Case-based reasoning*. Morgan Kaufmann, San Mateo, Ca., 1993.
- [109] K. G. Konolige and M. E. Pollack. A representationalist theory of intentions. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 1989.
- [110] D. Krahl, U. Windheuser, and F.K. Zick. *Data Mining*. Addison Wesley, 1998.
- [111] A. Krogh. An introduction to Hidden Markov Models for biological sequences. In S.L. Salzberg, D.B. Searls, and S. Kasif, editors, *Computational methods in molecular biology*, pages 45–63. Elsevier, 1998.
- [112] A. Krogh, M. Brown, I.S. Mian, K. Sjölander, and D. Haussler. Hidden markov models in computational biology. *J. Mol. Biol.*, 235:1501–1531, 1994.
- [113] A. Krüger et al. *Die große GU Nährwert-Kalorien-Tabelle 2002/03*. Gräfe & Unzer, 2002.
- [114] R. Kruse, J. Gebhardt, and F. Klawonn. *Fuzzy-Systeme*. Teubner, Stuttgart, 1993.
- [115] F. Kulmann and E. Reucher. Computergestützte Bonitätsprüfung bei Banken und Handel. *DBW - Die Betriebswirtschaft*, 60:113–122, 2000.
- [116] Z. Kunda. The case for motivated reasoning. *Psychological Bulletin*, 108(3):480–498, 1990.

- [117] S.L. Lauritzen. *Graphical Models*. Clarendon Press, Oxford, UK, 1996.
- [118] S.L. Lauritzen and D.J. Spiegelhalter. Local computations with probabilities in graphical structures and their applications to expert systems. *Journal of the Royal Statistical Society B*, 50(2):415–448, 1988.
- [119] S.L. Lauritzen, B. Thiesson, and D.J. Spiegelhalter. Diagnostic systems by model selection: a case study. In P. Cheeseman and R.W. Oldford, editors, *Selecting models from data*, number 89 in Lecture Notes in Statistics, pages 143–152. Springer, New York Berlin Heidelberg, 1994.
- [120] D.B. Leake, editor. *Case-based reasoning*. AAAI Press, Menlo Park, Ca., 1996.
- [121] D.B. Leake, A. Kinley, and D. Wilson. Learning to improve case adaptation by introspective reasoning and CBR. In D.B. Leake, editor, *Case-based reasoning*, pages 185–197. AAAI Press, Menlo Park, Ca., 1996.
- [122] D.B. Lenat and R.V. Guha. *Building Large Knowledge-Based Systems: Representation and Inference in the CYC Project*. Addison-Wesley, Massachusetts, 1990.
- [123] Y. Lésperance, H. J. Levesque, F. Lin, D. Marcu, R. Reiter, and R. B. Scherl. Foundations of a logical approach to agent programming. In M. Wooldridge, J. P. Müller, and M. Tambe, editors, *Intelligent Agents II: Agent Theories, Architectures, and Languages*, volume 1037 of *LNAI*, pages 331–346. Springer-Verlag, 1996.
- [124] H. Levesque, R. Reiter, Y. Lesperance, F. Lin, and R. Scherl. GOLOG: A logic programming language for dynamic domains. *Journal of Logic Programming. Special Issue on Actions*, 31(1-3):59–83, 1997.
- [125] H. J. Levesque. What is planning in the presence of sensing? In *Proc. of the National Conference on Artificial Intelligence (AAAI'96)*, pages 1139–1146, 1996.
- [126] J. Lieber and A. Napoli. Correct and complete retrieval for case-based problem-solving. In H. Prade, editor, *Proceedings 13th European Conference on Artificial Intelligence, ECAI'98*, pages 68–72, Chichester, 1998. John Wiley & Sons.
- [127] V. Lifschitz. On the semantics of STRIPS. In *Reasoning about Actions and Plans: Proc. of the 1986 Workshop*, pages 1–9. Morgan Kaufmann, 1986.
- [128] V. Lifschitz. Answer set planning. In *Proceedings 16th International Conference on Logic Programming*, pages 25–37, Cambridge, MA, 1999. MIT Press.
- [129] V. Lifschitz. Answer set programming and plan generation. *Artificial Intelligence*, 138:39–54, 2002.
- [130] D. Makinson. General patterns in nonmonotonic reasoning. In D.M. Gabbay, C.H. Hogger, and J.A. Robinson, editors, *Handbook of Logic in Artificial Intelligence and Logic Programming*, volume 3, pages 35–110. Oxford University Press, 1994.
- [131] V.W. Marek and M. Truszczyński. Stable models and an alternative logic programming paradigm. In *The Logic Programming Paradigm: A 25-Year Perspective*, pages 375–398. Springer, Berlin, 1999.
- [132] W. Mark, E. Simoudis, and D. Hinkle. Case-based reasoning: Expectations and results. In D.B. Leake, editor, *Case-based reasoning*, pages 269–294. AAAI Press, Menlo Park, Ca., 1996.
- [133] D. McAllester. Reasoning utility package user's manual. Technical report, MIT, Cambridge, Ma., 1982. AI Lab Report 667.
- [134] J. McCarthy. Circumscription - a form of nonmonotonic reasoning. *Artificial Intelligence*, 13:27–39, 1980.
- [135] J. McCarthy and P. Hayes. Some philosophical problems from the standpoint of Artificial Intelligence. In B. Meltzer and D. Michie, editors, *Machine Intelligence 4*. Edinburgh University Press, Edinburgh, 1969.

- [136] John McCarthy. Ascribing mental qualities to machines. In Martin Ringle, editor, *Philosophical Perspectives in Artificial Intelligence*. Harvester Press, 1979.
- [137] D. McDermott. A general framework for reason maintenance. *Artificial Intelligence*, 50:289–329, 1991.
- [138] D. McDermott and J. Doyle. Non-monotonic logic I. *Artificial Intelligence*, 13:41–72, 1980.
- [139] J. McDermott. R1: A rule-based configurer of computer systems. *Artificial Intelligence*, 19(1):39–88, 1982.
- [140] J. McDermott. R1 (“XCON”) at age 12: lessons from an elementary school achiever. *Artificial Intelligence*, 59:214–247, 1993.
- [141] N. Megiddo and R. Srikant. Discovering predictive association rules. In *Proceedings of the 4th International Conference on Knowledge Discovery in Databases and Data Mining*, 1998.
- [142] O. Mehlmann, J. Landvogt, A. Jameson, T. Rist, and R. Schäfer. Einsatz Bayes’scher Netze zur Identifikation von Kundenwünschen im Internet. *KI*, 3:43–48, 1998.
- [143] P. Mertens, V. Borkowski, and W. Geis. *Betriebliche Expertensystem-Anwendungen*. Springer-Verlag, Berlin-Heidelberg-New York, 3 edition, 1993.
- [144] C.-H. Meyer. *Korrektes Schließen bei unvollständiger Information*. Peter Lang Verlag, 1998.
- [145] C.-H. Meyer, G. Kern-Isberner, and W. Rödder. Analyse medizinisch-soziologischer Daten mittels eines probabilistischen Expertensystems. In *Proceedings Symposium on Operations Research SOR’95*, pages 347–352. Springer, 1995.
- [146] R.S. Michalski. Understanding the nature of learning: Issues and research directions. In R.S. Michalski, J.G. Carbonell, and T.M. Mitchell, editors, *Machine Learning: An Artificial Intelligence Approach*, volume II. Morgan Kaufmann, San Mateo, Calif., 1986.
- [147] R.S. Michalski, J.G. Carbonell, and T.M. Mitchell, editors. *Machine Learning: An Artificial Intelligence Approach*, volume II. Morgan Kaufmann, San Mateo, Calif., 1986.
- [148] R.S. Michalski and Y. Kodratoff. Research in machine learning; recent progress, classification of methods, and future directions. In Y. Kodratoff and R. S. Michalski, editors, *Machine Learning: An Artificial Intelligence Approach*, volume III. Morgan Kaufmann, San Mateo, Calif., 1990.
- [149] D. Michie. Current developments in expert systems. In *Proc. 2nd Australian Conference on Applications of Expert Systems*, pages 163–182, Sydney, Australia, 1986.
- [150] J. Mingers. An empirical comparison of selection measures of decision-tree induction. *Machine Learning Journal*, 3(4):319–342, 1989.
- [151] M. Minor and M. Lenz. Textual CBR im E-Commerce. *Künstliche Intelligenz*, 1/01:12–16, 2001.
- [152] T.M. Mitchell. Generalization as search. *Artificial Intelligence*, 18(2):203–226, 1982.
- [153] T.M. Mitchell. *Machine learning*. McGraw-Hill, New York, 1997.
- [154] T. Munakata. Knowledge discovery. *Communications of the ACM*, 42(11):26–67, 1999.
- [155] D. Nardi. Artificial Intelligence in RoboCup. In *Proceedings of the 14th European Conference on Artificial Intelligence (ECAI’2000)*, pages 756–762. IOS Press, 2000.

- [156] D. Nardi, M. Riedmiller, C. Sammut, and J. Santos-Victor, editors. *RoboCup 2004: Robot Soccer World Cup VIII*. Number 3276 in LNAI. Springer, Berlin Heidelberg New York, 2005.
- [157] R.E. Neapolitan. *Probabilistic Reasoning in expert systems*. Wiley, New York, 1990.
- [158] Allen Newell. The knowledge level. *Artificial Intelligence*, 18(1):87–127, 1982.
- [159] I. Niemelä. Logic programming with stable model semantics as a constraint programming paradigm. *Ann. Math. Artificial Intelligence*, 25(3–4):241–273, 1999.
- [160] N. J. Nilsson. *Artificial Intelligence: A New Synthesis*. Morgan Kaufmann, 1998.
- [161] N.J. Nilsson. Probabilistic logic. *Artificial Intelligence*, 28:71–87, 1986.
- [162] J.B. Paris and A. Vencovská. A note on the inevitability of maximum entropy. *International Journal of Approximate Reasoning*, 14:183–223, 1990.
- [163] J. Pearl. Fusion, propagation and structuring in belief networks. *Artificial Intelligence*, 29:241–288, 1986.
- [164] J. Pearl. *Probabilistic Reasoning in Intelligent Systems*. Morgan Kaufmann, San Mateo, Ca., 1988.
- [165] J. Pearl and A. Paz. Graphoids: A graph based logic for reasoning about relevancy relations. In B.D. Boulay, D. Hogg, and L. Steel, editors, *Advances in Artificial Intelligence II*, pages 357–363. North-Holland, Amsterdam, 1986.
- [166] E.P.D. Pednault. *Toward a Mathematical Theory of Plan Synthesis*. PhD thesis, Department of Electrical Engineering, Stanford University, 1986.
- [167] E.P.D. Pednault. ADL: Exploring the middle ground between STRIPS and the situation calculus. In R.J. Brachman, H. Levesque, and R. Reiter, editors, *Proc. of the First International Conference on Principles of Knowledge Representation and Reasoning (KR'89)*, pages 324–332. Morgan Kaufmann, San Francisco, CA, 1989.
- [168] D. Poole. A logical framework for default reasoning. *Artificial Intelligence*, 36:27–47, 1988.
- [169] D.L. Poole, R.G. Goebel, and R. Aleliunas. Theorist: A logical reasoning system for defaults and diagnosis. In N. Cercone and G. McCalla, editors, *The Knowledge Frontier: Essays in the representation of knowledge*, pages 331–352. Springer, New York, 1987.
- [170] W. Poundstone. *Im Labyrinth des Denkens*. Rowohlt Taschenbuch Verlag, Reinbek bei Hamburg, 4. edition, 2000.
- [171] F. Puppe. *Problemlösungsmethoden in Expertensystemen*. Springer-Verlag, Berlin Heidelberg, 1990.
- [172] J.R. Quinlan. Learning efficient classification procedures and their application to chess end games. In R. S. Michalski, J. Carbonell, and T. M. Mitchell, editors, *Machine Learning: An Artificial Intelligence Approach*. Morgan Kaufmann, 1983.
- [173] J.R. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, San Mateo, California, 1993.
- [174] L.R. Rabiner. A tutorial on Hidden Markov Models and selected applications in speech recognition. In *Proceedings IEEE*, volume 77(2), pages 257–286, 1989.
- [175] A. S. Rao and M. P. Georgeff. Modeling rational agents within a BDI-architecture. In M. N. Huhns and M. P. Singh, editors, *Readings in Agents*, pages 317–328. Morgan Kaufmann, San Francisco, 1997. (Reprinted from Proceedings of the International Conference on Principles of Knowledge Representation and Reasoning, 1991).

- [176] R. Reiter. On closed world data bases. In H. Gallaire and J. Minker, editors, *Logic and Data Bases*, pages 119–140. Plenum Press, New York, 1978.
- [177] R. Reiter. A logic for default reasoning. *Artificial Intelligence*, 13:81–132, 1980.
- [178] F. Rheinberg. *Motivation*. Kohlhammer, Stuttgart, fünfte, überarbeitete und erweiterte Auflage, 2004.
- [179] M.M. Richter and S. Wess. Similarity, uncertainty and case-based reasoning in PATDEX. In R.S. Boyer, editor, *Automated reasoning, essays in Honor of Woody Bledsoe*. Kluwer Acad. Publ., 1991.
- [180] W. Rödder and G. Kern-Isberner. Léa Sombé und entropie-optimale Informationsverarbeitung mit der Expertensystem-Shell SPIRIT. *OR Spektrum*, 19(3):41–46, 1997.
- [181] W. Rödder and G. Kern-Isberner. Representation and extraction of information by probabilistic logic. *Information Systems*, 21(8):637–652, 1997.
- [182] W. Rödder and C.-H. Meyer. Coherent knowledge processing at maximum entropy by SPIRIT. In E. Horvitz and F. Jensen, editors, *Proceedings 12th Conference on Uncertainty in Artificial Intelligence*, pages 470–476, San Francisco, Ca., 1996. Morgan Kaufmann.
- [183] S. Russell and P. Norvig. *Artificial Intelligence – A Modern Approach*. Prentice Hall, New Jersey, 2nd edition, 2003.
- [184] E. Sacerdoti. The non-linear nature of plans. In *Proceedings of the Fourth International Joint Conference on Artificial Intelligence (IJCAI-75)*, pages 206–214. Morgan Kaufmann, 1975. (Reprinted in J. Allen, J. Hendler, and A. Tate (eds.): *Readings in Planning*, pp. 162–170, San Francisco: Morgan Kaufmann, 1990).
- [185] E. Sacerdoti. *A Structure for Plans and Behavior*. American Elsevier, New York, 1977.
- [186] C. Sammut, S. Hurst, D. Kedzier, and D. Michie. Learning to fly. In *Proc. of the Ninth International Conference on Machine Learning*, Aberdeen, 1992. Morgan Kaufmann.
- [187] T. Schaub and K. Wang. Towards a semantic framework for preference handling in answer set programming. *Theory and Practice of Logic Programming, special issue on Answer Set Programming*, 3(4-5):569–607, 2003.
- [188] R. Scherl and H.J. Levesque. The frame problem and knowledge producing actions. In *AAAI-93*, pages 689–695, Washington, DC, 1993.
- [189] P. H. Schmitt. *Theorie der logischen Programmierung*. Springer-Lehrbuch. Springer-Verlag, Berlin, Heidelberg, New York, 1992.
- [190] M. Schramm and W. Ertel. Reasoning with probabilities and maximum entropy: the system PIT and its application in LEXMED. In *Symposium on Operations Research, SOR'99*, 1999.
- [191] M. Schramm and V. Fischer. Probabilistic reasoning with maximum entropy – the system PIT. In *Proceedings of the 12th Workshop on Logic Programming*, 1997.
- [192] B. Schweizer and A. Sklar. *Probabilistic metric spaces*. North-Holland, Elsevier, Amsterdam, 1983.
- [193] G. Shafer. *A mathematical theory of evidence*. Princeton University Press, Princeton, NJ, 1976.
- [194] C.E. Shannon and W. Weaver. *Mathematische Grundlagen der Informationstheorie*. Oldenbourg, München, Wien, 1976.

- [195] J.E. Shore and R.W. Johnson. Axiomatic derivation of the principle of maximum entropy and the principle of minimum cross-entropy. *IEEE Transactions on Information Theory*, IT-26:26–37, 1980.
- [196] J. Siekmann and G. Wrightson, editors. *Automation of Reasoning, Classical Papers on Computational Logic 1957-1966*. Springer-Verlag, Berlin, 1983.
- [197] J. Siekmann and G. Wrightson, editors. *Automation of Reasoning, Classical Papers on Computational Logic 1967-1970*. Springer-Verlag, Berlin, 1983.
- [198] H. A. Simon. Why should machines learn? In R.S. Michalski, J.G. Carbonell, and T.M. Mitchell, editors, *Machine Learning: An Artificial Intelligence Approach*. Tioga Publishing Company, Palo Alto, 1983.
- [199] H. A. Simon. Motivational and emotional controls of cognition. In *Models of Thought*, pages 29–38. Yale University Press, 1997.
- [200] P. Simons, I. Niemelä, and T. Soinen. Extending and implementing the stable model semantics. *Artificial Intelligence*, 138:181–234, 2002.
- [201] M. P. Singh. Semantical considerations on intention dynamics for BDI agents. *Journal of Experimental and Theoretical Artificial Intelligence*, 10(4):551–564, 1998.
- [202] M. P. Singh, A. S. Rao, and M. P. Georgeff. Formal methods in DAI: Logic-based representation and reasoning. In G. Weiss, editor, *Multiagent Systems - A Modern Approach to Distributed Artificial Intelligence*, pages 331–376. The MIT Press, Cambridge, Massachusetts, 1999.
- [203] A. Sloman. Motives, mechanisms, and emotions. *Cognition and Emotion*, 1(3):217–233, 1987.
- [204] P. Smets and R. Kennes. The transferable belief model. *Artificial Intelligence*, 66:191–234, 1994.
- [205] P. Smyth, D. Heckerman, and M.I. Jordan. Probabilistic independence networks for hidden Markov probability models. Technical report, Microsoft Research, 1996. MSR-TR-96-03.
- [206] T. Soinen and I. Niemelä. Developing a declarative rule language for applications in product configuration. In *Proceedings First International Workshop on Practical Aspects of Declarative Languages*, volume 1551 of *Lecture Notes in Computer Science*, pages 305–319, Berlin, 1998. Springer.
- [207] Léa Sombé. *Schließen bei unsicherem Wissen in der Künstlichen Intelligenz*. Vieweg, Braunschweig, 1992.
- [208] R. Srikant and R. Agrawal. Mining generalized association rules. In *Proceedings of the 21st VLDB Conference*, Zürich, Switzerland, 1995.
- [209] L. Steels. Cooperation between distributed agents through self organization. In Y. Demazeau and J.-P. Müller, editors, *Decentralized AI - Proceedings of the First European Workshop on Modelling Autonomous Agents in a Multi-Agent World (MAAMAW-89)*, pages 175–196, Amsterdam, The Netherlands, 1990. Elsevier Science Publishers.
- [210] L. Sterling and E. Shapiro. *The Art of Prolog - Advanced programming techniques*. The MIT Press, Cambridge, Massachusetts, 2. edition, 1994.
- [211] M. Studený. Conditional independence relations have no finite characterization. In *Proceedings of the 11th Prague Conference on Information Theory, Statistical Decision Functions and Random Processes*, Prag, 1990.
- [212] G. Sussman. *A Computer Model of Skill Acquisition*. Elsevier/North Holland, Amsterdam, 1975.

- [213] R.E. Tarjan and M. Yannakakis. Simple linear-time algorithms to test chordality of graphs, test acyclicity of hypergraphs, and selectively reduce acyclic hypergraphs. *SIAM Journal of Computing*, 13:566–579, 1984.
- [214] A. Tate. Generating project networks. In *Proc. of the Fifth International Joint Conference on Artificial Intelligence (IJCAI-77)*, pages 888–893, San Francisco, 1977. Morgan Kaufmann. (Reprinted in J. Allen, J. Hendler and A. Tate (eds): *Readings in Planning*, Morgan Kaufmann, 1990).
- [215] M. Thielscher. *Reasoning Robots*, volume 33 of *Applied Logic Series*. Springer, 2005.
- [216] F. Topsøe. *Informationstheorie*. B. G. Teubner, Stuttgart, 1973.
- [217] M. Veloso, E. Pagello, and H. Kitano, editors. *RoboCup-99: Robot Soccer World Cup III*. Springer-Verlag, 2000.
- [218] F. Voorbraak. On the justification of Dempster’s rule of combination. *Artificial Intelligence*, 48:171–197, 1991.
- [219] F. Voorbraak. Combining unreliable pieces of evidence. Technical Report LP-95-07, Institute for Logic, Language and Computation, University of Amsterdam, 1995.
- [220] F. Voorbraak. Reasoning with uncertainty in AI. In L. Dorst, M. van Lambalgen, and F. Voorbraak, editors, *Proceedings Reasoning with Uncertainty in Robotics (RUR’95)*, LNCS/LNAI 1093, pages 52–90, Berlin, 1996. Springer.
- [221] C. Walther. A mechanical solution of Schubert’s steamroller by many-sorted resolution. *Artificial Intelligence*, 26:217–224, 1985.
- [222] C. Walther. *A Many-Sorted Calculus Based on Resolution and Paramodulation*. Research Notes in Artificial Intelligence. Pitman, London, and Morgan Kaufmann, Los Altos, Calif., 1987.
- [223] R. Watson. An application of action theory to the space shuttle. In G. Gupta, editor, *Proc. First Internat. Workshop on Practical Aspects of Declarative Languages*, volume 1551 of *Lecture Notes in Computer Science*, pages 290–304, Berlin, Heidelberg, New York, 1998. Springer-Verlag.
- [224] G. Weiss, editor. *Multiagent systems: a modern approach to distributed artificial intelligence*. MIT Press, Cambridge, Massachusetts, 1999.
- [225] T. Winograd. Understanding natural language. *Cognitive Psychology*, 3(1), 1972.
- [226] W. A. Woods and J. G. Schmolze. The KL-ONE family. *Computers & Mathematics with Applications*, 23(2–5):133–177, 1992. Special issue on Semantic Networks in Artificial Intelligence.
- [227] W.A. Woods. Progress in natural languages understanding: An application to lunar geology. *AFIPS Conference Proceedings*, 42:441–450, 1973.
- [228] M. Wooldridge. Intelligent agents. In G. Weiss, editor, *Multiagent Systems - A Modern Approach to Distributed Artificial Intelligence*, pages 27–78. The MIT Press, Cambridge, Massachusetts, 1999.
- [229] M. Wooldridge and N. R. Jennings. Intelligent agents: Theory and practice. *The Knowledge Engineering Review*, 10(2):115–152, 1995.
- [230] M.J. Wooldridge. *An Introduction to Multiagent Systems*. John Wiley & Sons Ltd., West Sussex, England, 2002.
- [231] L.A. Zadeh. Fuzzy sets. *Information and Control*, 8:338–353, 1965.
- [232] L.A. Zadeh. Fuzzy sets as a basis for a theory of possibility. *Fuzzy Sets and Systems*, 1(1):3–28, 1978.



# Index

## Symbole:

$\neg$ , 42  
 $\wedge$ , 42  
 $\wedge$ -Einf, 38  
 $\wedge$ -Elim, 38  
 $\vee$ , 42  
 $\Rightarrow$ , 42  
 $\Leftrightarrow$ , 42  
 $\leftarrow$ , 270  
 $:-$ , 270  
 $\forall$ , 52  
 $\forall$ -Inst, 58  
 $\exists$ , 52  
 $\exists$ -Inst, 58  
 $\equiv$ , 45  
 $\models$ , 35, 37  
 $\not\models$ , 37  
 $\models^{as}$ , 289  
 $\models^{stab}$ , 285  
 $\top$ , 431  
 $\perp$ , 431  
 $\square$ , 40  
 $\vdash$ , 39  
 $\vdash$ , 205  
 $\vdash_{\mathcal{J}}$ , 227  
 $\vdash_D^{Poole}$ , 264  
 $\vdash_{\Delta}^{Reiter}$ , 264  
 $\perp\!\!\!\perp$ , 361  
 $\perp\!\!\!\perp_{\mathcal{G}}$ , 361, 458  
 $\perp\!\!\!\perp_P$ , 443  
 $\llbracket - \rrbracket_I$ , 34, 44, 54  
 $\llbracket - \rrbracket_{I,\alpha}$ , 51, 52  
 $\bar{a}$ , 440  
 $\dot{a}$ , 440  
 $Cn$ , 37  
 $C_D^{Poole}$ , 264  
 $C_{\Delta}^{Reiter}$ , 264  
 $\mathcal{H}(P)$ , 279  
 $\mathcal{J}(n)$ , 208  
 $\mathcal{P}^S$ , 283  
 $P(A | B)$ , 436  
 $\mathcal{R}^{Ac}$ , 338

$\mathcal{R}^E$ , 338  
 $\mathcal{R}(Ag, Env)$ , 339  
 $T_{\mathcal{P}}$ , 279  
 $H(P)$ , 115, 448  
 $\Lambda_T$ , 242  
 $\Pi[k]$ , 243  
 $\Omega_A$ , 432  
  
A-Liste, 313  
Abduktion, 23  
abgeschlossen, 211  
Abhängigkeit, 442  
Abhängigkeitsgraph, 364  
Absicht, 336  
Absichtshaltung, 336  
Absorption, 46  
ABSTRIPS, 324  
 $Ac$ , 337  
 $ACons(n)$ , 213  
 $ACons^*(n)$ , 214  
action, 339, 340, 342, 349, 350, 355  
 $ad(A)$ , 226  
 $ad_{\mathcal{J}}(A)$ , 226  
Adaption, 166, 196  
adjazent, 452  
 $Ag$ , 339  
 $\mathcal{AG}$ , 339  
Agent, 331–333  
— Aufgabe eines  $\sim$ , 343  
— BDI- $\sim$ , 351  
— Erfolg eines  $\sim$ , 343  
— intelligenter, 335  
— logikbasierter, 347  
— menschlicher, 333  
— optimaler, 344  
— reaktiver, 340, 346  
Ähnlichkeit, 185  
Ähnlichkeitsmaß, 185, 189, 193  
Ähnlichkeitsproblem, 179  
Aktion, 304, 337  
— optimale, 349  
Algorithmus, 41  
Allabschluss, 53  
allgemeingültig, 35  
Allquantor, 52  
 $an(v)$ , 453

- $An(\mathbf{W})$ , 453  
 Anfrage, 270  
 Annahme  
 — in einem ATMS, 229  
 — in einem JTMS, 220  
 — maximale, 220  
 — STRIPS-Annahme, 318  
 answer set, *siehe* Antwortmenge  
 answer set programming, *siehe* Antwortmengenprogrammierung  
 answer set semantics, *siehe* Antwortmengensemantik  
 $Ant(n)$ , 213  
 $Ant^*(n)$ , 214  
 Antezedenz, 71, 213, 229  
 Antibiotikatherapie, 14  
 Antwort  
 — korrekte, 271  
 Antwortmenge, 288, 289, 296  
 Antwortmengenprogrammierung, 298  
 Antwortmengensemantik, 289  
 Antwortsubstitution, 271  
 — berechnete, 275  
 — korrekte, 271  
 Antwortvollständigkeit, 277  
 — bezüglich der Selektionsfunktion, 278  
 Apriori, 148  
 Apriori-Algorithmus, 147  
 AprioriGen, 149  
 AprioriHybrid, 150  
 AprioriTid, 150  
 Äquivalenz, 42  
 — aussagenlogische, 46  
 — Erfüllbarkeitsäquivalenz, 61  
 — prädikatenlogische, 55  
 — semantische, 45, 55  
 — verhaltensmäßige, 339  
 ARCHIE, 173  
 Assoziationen, 145  
 Assoziationsregel, 145, 146  
 — Konfidenz einer  $\sim$ , 146  
 — Support einer  $\sim$ , 146  
 — verallgemeinerte, 150  
 Assumption-based Truth Maintenance-Systeme, 207, 228  
 ATMS, *siehe* Assumption-based Truth Maintenance-Systeme  
 Atom, 51  
 — ausgewähltes, 275  
 — Grundatom, 52  
 Ausführungsbedingung, 305  
 Aussagenlogik, 34, 42  
 Aussagenvariable, 29, 42  
 — mehrwertige, 440  
 Auswirkungen, 214  
 Autonomie, 333  
 Axiom, 38  
 azyklisch  
 — Graph, 452  
 — Hypergraph, 464  
 BACKCHAIN, 84  
 Backtracking, 278  
 — chronologisches, 221  
 — dependency directed, 220  
 Basismaß, 413  
 Bayessche Glaubensfunktion, 415  
 Bayessche Regel, 438  
 Bayessches Netzwerk, 376  
 BDI-Agent, 351  
 BDI-Architektur, 351  
 Begründung, 239  
 — eines ATMS, 229  
 — eines JTMS, 208  
 — fundiert gültige, 214  
 — fundiert ungültige, 214  
 — gültige, 210  
 — monotone, 208, 220  
 — nicht-fundiert ungültige, 215  
 — nichtmonotone, 208, 220  
 — stützende, 211  
 — zirkuläre, 211  
 behavioristisch, 346  
 Beispiel, 108, 120  
 — fälschlicherweise negatives, 128  
 — fälschlicherweise positives, 128  
 Beispielsprache, 122  
 Bel, 353  
 Belegung, 31, 43  
 — Variablenbelegung, 51  
 belief, 336, 351  
 belief function, 413  
 belief revision, 235, 354  
 Bindungspriorität, 43, 52  
 bit, 447  
 Blockwelt, 302  
 Blue Babies, 404  
 BOBLO, 394  
 BOOL, 34  
 brf, 354  
 C4.5, 115, 117  
 case-based reasoning, 155  
 — interpretatives, 162  
 — problemlösendes, 162  
 CBR, *siehe* case-based reasoning

- CBR-Zyklus, 162  
 certainty factor, *siehe* Sicherheitsfaktor  
*CF*, 90, 91  
 charakteristische Funktion, 420  
 CHEF, 169  
 CHILD, 394, 404  
 CLAVIER, 161  
 Clique, 454  
 Cliquenbaum, 383, 461  
 closed world assumption, 286, 288, 293  
*Cl(P)*, 283  
 Cluster, 144  
 Clustering, 144  
 — demographisches, 145  
*Cn*, 37  
*confidence*, 146  
*cons( $\delta$ )*, 239  
*Cons(n)*, 213  
 Constraint, 281, 292  
 CWA, *siehe* closed world assumption  
 CYC, 18
- D*-Liste, 313  
 D-map, 364  
 d-Separation, 459  
 DAG, *siehe* gerichteter, azyklischer Graph  
 Data Mining, 141, 144  
 DCA, *siehe* domain closure assumption  
 DDB-Algorithmus, 220  
 de Morgansche Regeln, 46  
*de(v)*, 453  
 Deduktion, 23, 26  
 Deduktionstheorem, 38  
 deduktiv abgeschlossen, 37  
 Default, 238, 260  
 — Anwendbarkeit eines  $\sim$ , 241  
 — normaler, 257, 330  
 — Reiter'scher, 238  
 Default-Negation, 281  
 Default-Theorie, 239  
 — Poole'sche, 260  
 Defaultbegründung, 239  
 Defaultkonsequenz, 239  
 Defaultvoraussetzung, 239  
 Deklarativität, 277  
 Deliberation, 351, 352  
 Delphin, 68  
 Dempster'sche Kombinationsregel, 416  
 Dempster-Shafer-Theorie, 412  
 DENDRAL, 13  
 Denken  
 — intelligentes, 1  
 — praktisches, 351
- dependency directed backtracking, 220  
 Derivationswiederholung, 199  
*Des*, 354  
 desire, 336, 351  
 Deskriptor, 174  
 Dialogkomponente, 18  
 Disjunktion, 29, 42  
 — epistemische, 296  
 disjunktives logisches Programm, 296  
 Distributivität, 46  
 DLV, 298, 324  
 DNF, *siehe* disjunktive Normalform  
 do (Situationskalkül), 305  
*Do* (Agenten), 348  
 domain closure assumption, 284  
 Doppelnegation, 46  
 Dreiecksnorm, 422  
 DS-Theorie, *siehe* Dempster-Shafer-Theorie  
*DT*, 112  
 dual, 411
- e-commerce, *siehe* Electronic Commerce  
 Effektaxiome, 306  
 effektorische Fähigkeit, 337  
 Eintopf, 180  
 Electronic Commerce, 143, 160, 394  
 Elementarereignis, 430  
 Eliminationsgraph, 456  
 Elternklausel, 64  
 EMYCIN, 16  
 Entropie, 115, 448  
 — bedingte, 449  
 — maximale, 397  
 — relative, 450  
 — Verbundentropie, 449  
 entscheidbar, 41  
 Entscheidungsbaum, 105  
 Entscheidungsunterstützungssystem, 298  
*Env*, 339  
 Episoden, 159  
 epistemische Disjunktion, 296  
 Ereignis, 430  
 Erfahrungswissen, 155  
 erfüllbar, 35  
 Erfüllbarkeitsäquivalenz, 61  
 Erfüllungsrelation, 32, 35  
 Erklärbarkeit, 261  
 Erklärungskomponente, 18  
 Ersetzbarkeitstheorem, 45, 55  
 Evidenz, 17, 76, 412  
 Evidenztheorie, 412  
*execute*, 355

- Existenzquantor, 52  
 Exklusivität, 433  
 Expertensystem, 11, 12, 18  
 explaining away, 376  
 Extension, 120, 242, 261, 423  
 Extensionsprinzip, 423  
  
 Fakt, 260, 270  
 Faktorisierung, 369  
 Faktorisierungsregel, 65  
 Fall, 168  
 fallbasiertes Schließen, 155  
 Fallbasis, 158  
 Fallbeispiele, 158  
 Fallrepräsentation, 172  
 Fill-in, 456  
*filter*, 355  
 Fixpunkt, 37, 242, 279, 294  
 Fluent, 305  
 fokales Element, 413  
 Folgerung  
 — durch Widerspruch, 40  
 — klassisch-logische, 37  
 — logische, 32, 37  
 — nichtmonotone, 226  
 — semantische, 37  
*Form*, 204, 431  
 Formel, 29, 35  
 — allgemeingültige, 35  
 — atomare, 42, 51, 431  
 — aussagenlogische, 42  
 — bereinigte, 60  
 — erfüllbare, 35  
 — erklärbare, 261  
 — falsifizierbare, 35  
 — geschlossene, 53  
 — Grundformel, 52  
 — inkonsistente, 35  
 — komplexe, 52  
 — konsistente, 35  
 — prädikatenlogische, 52  
 — unerfüllbare, 35  
 — Wert einer  $\sim$ , 44, 51, 52  
 — widersprüchliche, 35  
 — wohlgeformte, 52  
*Formel*( $\Sigma$ ), 30  
*Formel* $_{\Sigma}$ ( $V$ ), 52  
 fortlaufende Schnitteigenschaft, 461  
 FRAIL, 394  
 frame axiom, *siehe* Rahmenaxiom  
 frame problem, *siehe* Rahmenproblem  
 Fundamente, 214  
 Fundiertheit, 211  
  
 Funktionssymbol, 48  
 Funktionsvariable, 67  
 Fuzzy-Logik, 27, 423  
 Fuzzy-Menge, 421  
 — normierte, 422  
 Fuzzy-Theorie, 420  
  
*gain*, 116  
*gain ratio*, 117  
 GASOIL, 114  
 Geldautomat, 7, 73, 76  
 Gelfond-Lifschitz-Reduktion, 283  
 Generalisierung  
 — allgemeinste, 127  
 — speziellste, 127  
 geschlossen unter  $\mathcal{P}$ , 283  
 Glaube, 336, 351  
 Glaubensfunktion, 413  
 Gleichverteilung, 433  
 GOLOG, 331  
 Graph  
 — azyklischer, 452  
 — Eliminationsgraph, 456  
 — Fill-in-Graph, 456  
 — gerichteter, 451  
 — leerer, 454  
 — moraler, 454  
 — perfekter, 365  
 — triangulierter, 455  
 — ungerichteter, 451  
 — vollständiger, 454  
 — zerlegbarer, 459  
 — Zerlegung eines Graphen, 459  
 Grundatom, 52  
 Grundformel, 52  
 Grundinstanz, 63  
 Grundliteral, 52  
 Grundterm, 50  
  
 Hailfinder, 394  
 Hamming-Ähnlichkeit, 186  
 — gewichtete, 188  
 Hamming-Distanz, 186  
 Herbrandbasis, 279  
 Herbrandinterpretation, 279  
 Herbrandmodell, 279  
 Herbranduniversum, 279  
 Hidden Markov Model, 401  
 Historie, 305  
 HMM, *siehe* Hidden Markov Model  
 Holmes, 375  
 Hornklausel, 270  
 Hornklausellogik, 67

- HUGIN, 393  
 Hyperbaum, 464  
 Hypergraph, 463  
   — azyklischer, 464  
   — reduzierter, 463  
 Hyperkanten, 463  
 Hypothese, 119, 122, 260  
   — induktive, 136  
 Hypothesensprache, 122
- I-map, 364  
 ID3, 115  
 Idempotenz, 46  
 Implikation, 30, 42, 46  
   — materiale, 30, 70  
*in*, 208, 244  
*In*( $\Pi$ ), 244  
 Index  
   — eines Falles, 173  
 Indexvokabular, 165, 173, 175  
 Individuenbereich, 49  
 Indizierung  
   — eines Falles, 165, 173  
   — invertierte, 183  
 Indizierungsverfahren, 176  
 inductive bias, *siehe* induktive Hypothese  
 Induktion, 23  
 Induktionsaxiom, 67  
*inf*( $M$ ), 411  
 Inferenz, 20  
   — datengetriebene, 78, 81  
   — leichtgläubige, 226, 263  
   — rückwärtsgerichtete, 83  
   — skeptische, 226, 263  
   — zielorientierte, 78, 83  
 Inferenzoperation, 205  
   — klassisch-logische, 37  
   — nichtmonotone, 264  
 Inferenzregel, 38  
 Inferenzrelation, 20, 205  
   — der Antwortmengensemantik, 289  
   — der stabilen Semantik, 285  
   — nichtmonotone, 264  
   — Poole'sche, 264  
   — Reiter'sche, 264  
   — TMS-Inferenzrelation, 226  
 Information, 446, 447  
   — eigentliche, 449  
   — gegenseitige, 116, 449  
   — Maßeinheit der  $\sim$ , 447  
 Informationsgehalt, 115, 446  
   — mittlerer, 447  
 Informationsgewinn, 116
- in*-Knoten, 210  
 Inkonsistenz, 35  
*in*-Liste, 208  
 Instantiierung, 389  
 Instantiierungsregel, 57, 58  
 Instanz, 120  
   — eines Terms, 63  
*Int*( $\Sigma$ ), 32  
 integrity constraints, 281  
 Intelligent Miner, 145  
 Intelligenz, 1, 97, 333  
   — emergente, 346  
   — verteilte, 357  
 Intention, 336, 351, 352  
 intention stack, 357  
 intentional stance, 336  
*Intent*, 354  
 Interaktion, 358  
 Interaktivität, 333  
 internal store, 336  
 INTERPLAN, 324  
 Interpretation, 31  
   — aussagenlogische, 43  
   — klassisch-logische, 34  
   — partielle, 288  
   — prädikatenlogische, 49  
   — wahrheitsfunktionale, 34  
 Item, 146  
 Itemmenge, 146  
   — große, 147  
   — häufige, 147  
   — k-Itemmenge, 146  
   — Support einer  $\sim$ , 146
- JLO-Algorithmus, 393, 403  
 JTMS, *siehe* Justification-based Truth Maintenance-Systeme  
 JTMS-Verfahren, 215  
 JUDGE, 175  
 junction tree, 461  
 Junktor, 29, 42  
*just*( $\delta$ ), 239  
 Justification-based Truth Maintenance-Systeme, 207, 208
- Kalkül, 38  
   — korrekter, 39  
   — negativer, 38  
   — positiver, 38  
   — Resolutionskalkül, 64  
   — Situationskalkül, 304  
   — vollständiger, 39  
 Kandidaten-Eliminations-Methode, 125

- Kapazität, 411
  - $\infty$ -monotone, 414
- kausales Netzwerk, 376
- KDD, *siehe* knowledge discovery
- KDD-Prozess, 143
- Kettenregel, 437
- Kino, 106
- KL-ONE, 68
- Klassifikation, 105, 119, 144, 193
- klassisch-logisch, 34
- Klausel, 61
  - definite, 270
- Klauselform, 61
- Klauselmenge, 62
- KNF, *siehe* konjunktive Normalform
- Knoten, 208, 228, 451
  - Elternknoten, 452
  - gültiger, 229
  - Kindknoten, 452
  - Nachbarn eines  $\sim$ , 452
  - Nachkomme eines  $\sim$ , 452
  - stützende, 213
  - Vorfahr eines  $\sim$ , 452
- knowledge discovery, 141
- knowledge-level, 336
- Koimplikation, 42, 46
- Kombinationsregel
  - Dempster'sche, 416
- Kommunikation, 358
- Kommutativität, 46
- Konfidenz
  - einer Assoziationsregel, 146
- Konjunktion, 29, 42
- Konklusion, 24, 71
- Konsequenz, 71, 213, 229
  - betroffene, 213
- konsistent, 281
- Konsistenz, 35
  - eines Konzepts, 121
- Konsistenzüberprüfung, 85
- Konstante, 48
- Kontext, 241
  - Anwendbarkeit bzgl. eines Kontextes, 241
  - eines ATMS, 229
- Kontradiktion, 35
- Kontraosition, 46
- Konvention, 358
- Konzept, 120
  - konsistentes, 121
  - korrektes, 121
  - vollständiges, 121
- Konzeptlernproblem, 122
- Konzeptsprache, 68, 122
- Kopf (einer Regel), 281
- Korrektheit
  - eines Kalküls, 39
  - eines Konzepts, 121
- Kritikelement, 98
- Kumulativität, 299
- Künstliche Intelligenz, 2
- Label, 230
- Lauf, 338
  - beendeter, 338
- Lauritzen-Spiegelhalter-Algorithmus, 385
- least commitment, 318
- LEG, 398
- LEG-Netz, 398
- Lernelement, 98
- Lernen
  - induktives, 104, 121
  - maschinelles, 97
  - von Bayesschen Netzen, 394
  - von Entscheidungsbäumen, 104
  - von Konzepten, 118, 136
- LEXMED, 400
- $lfp(T_{\mathcal{P}})$ , 279
- Linearitätsannahme, 318
- Literal, 52, 431
  - komplementäres, 281
- Logik
  - Aussagenlogik, 34
  - Prädikatenlogik, 34
  - Aussagenlogik, 42
  - Fuzzy-Logik, 423
  - Hornklausellogik, 67
  - monotone, 204
  - nichtmonotone, 204
  - ordnungssortierte, 68
  - possibilistische, 426
  - Prädikatenlogik, 47
  - Prädikatenlogik 2. Stufe, 67
  - sortierte, 67
  - terminologische, 68
- logische Regel
  - erweiterte, 281
- logisches Programm, 270
  - disjunktives, 296
  - erweitertes, 281
  - normales, 281
- logisches Programmieren, 66, 67, 269
- lokal stratifiziert, 286
- LTMS, 235
- LUNAR, 14

- Majority Val*, 111  
 marginale Verteilung, 435  
 Marginalisieren, 435  
 Markov-Bedingung  
 — globale gerichtete, 376  
 — lokale gerichtete, 376  
 Markov-Decke, 366  
 Markov-Eigenschaft  
 — globale, 365  
 — lokale, 367  
 — paarweise, 366  
 Markov-Feld, 365  
 Markov-Graph, 365  
 Markov-Netz, 365  
 Markov-Rand, 367  
 maschinelles Lernen, 97  
 Massenspektrogramm, 13  
*MaxAnn*( $n_{\perp}$ ), 220  
 Maxikonsistenz, 260  
 maximale Entropie, 397  
 Maximalzahl-Suche, 457  
 — bei Hypergraphen, 464  
 maximum likelihood, 145, 395  
 maximum cardinality search, *siehe*  
   Maximalzahl-Suche  
 MCS, *siehe* maximum cardinality search  
 means-ends analysis, *siehe* Mittel-Ziel-  
   Analyse  
 means-ends reasoning, 352  
 MEDIATOR, 170  
 Mehrwertsteuer, 25, 60  
 mental state, 336  
 Merkmalsbaum, 136, 184  
 mgu, *siehe* allgemeinsten Unifikator  
*minconf*, 147  
*minsupp*, 147  
 Mittel-Ziel-Analyse, 315  
 Mittel-Ziel-Denken, 352  
*Mod $_{\Sigma}$* ( $F$ ), 35  
 Modell  
 — abgeschlossenes, 211  
 — eines JTMS, 210  
 — fundiertes, 211, 226  
 — klassisches, 35  
 — stabiles, 284  
 — zulässiges, 211, 218, 226  
 Modellsuche, 145  
 modus ponens, 38, 78, 426  
 modus tollens, 38, 80  
 MOLTKE, 193  
 Monotonie, 204, 299, 411  
 — Semi-Monotonie, 255, 257, 262  
 — vorsichtige, 299  
 Motivation, 356  
 MP, 38  
 MT, 38  
 Multiagentensystem, 357  
 multiple alignment, 400  
 MUNIN, 394, 404  
 MYCIN, 14, 89  
  
 Nachkommen, 452  
*nb*( $v$ ), 452  
*nd*( $v$ ), 453  
 Negation, 30, 42  
 — als Fehlschlag, 235, 272, 281  
 negation as failure, *siehe* Negation als Fehl-  
   schlag  
*neg*( $r$ ), 281  
 Netz  
 — Bayessches, 376  
 — kausales, 376  
 — LEG-Netz, 398  
 — Markov-Netz, 365  
*next*, 342, 349  
 nichtmonoton ableitbar, 264  
 Niveau-Abbildung, 286  
 Nixon-Raute, 262  
 NOAH, 324  
 nogood, 218, 229  
 Normalform, 60, 62  
 — disjunktive, 61  
 — Klauselform, 61  
 — konjunktive, 61  
 — verneinungstechnische, 60  
 Normalität, 22  
*not*, 281  
 Notwendigkeitsmaß, 425  
 Nutzen, 344  
 Nutzenfunktion, 344  
  
 Occam's Razor, 109  
 odd loops, 215, 217  
 Operatorabstraktion, 324  
 Option, 351  
*options*, 355  
*out*, 208, 244  
*Out*( $\Pi$ ), 244  
*out*-Knoten, 210  
*out*-Liste, 208  
 overfitting, 108  
  
*pa*( $v$ ), 452  
 Parametersuche, 145  
 PATDEX, 193  
 Peano-Axiome, 67

- Peirce, 23  
*Per*, 340  
 percept, *siehe* Wahrnehmung  
 Performanzelement, 98  
 Persistenz, 352  
 PIT, 400  
 PL1, 34  
 Plan, 314, 353, 356  
 Planen  
 — lineares, 318  
 — mit Antwortmengen, 324  
 — mit unvollständigen Plänen, 320  
 — nichtlineares, 320  
 Plangenerierung, 310  
 Plantransformation, 320  
 Plausibilität, 24, 413  
 Plausibilitätsfunktion, 413  
 Plausibilitätsgrad, 28  
 Politiker, 58  
 positive Form, 292  
*pos(r)*, 281  
 Possibilitätsmaß, 425  
 Possibilitätstheorie, 424  
 Possibilitätsverteilung, 424  
 Potentialdarstellung, 370  
 Potentialfunktionen, 370  
 practical reasoning, *siehe* praktisches Denken  
 Prädikatenspezifikation, 345  
 Prädikat, 48  
 — vages, 27  
 Prädikatenlogik, 34, 47  
 — 2. Stufe, 67  
 Prädikatensymbol, 48  
 Prädikatenvariable, 67  
 Prämisse, 24, 71, 208  
 Pränexform, 60  
 praktisches Denken, 351  
*pre(δ)*, 239  
 Proaktivität, 335  
 probabilistische Regel, 397  
 Problemgenerator, 98  
 Procedural Reasoning System, 356  
 Produktionsregel, 71  
 Programm  
 — logisches, 270  
 Programmieren  
 — deklaratives, 277  
 — logisches, 269  
 Prolog, 67, 270, 278, 349  
 Propagationsalgorithmus, 385  
 Propagationsregeln, 91  
 PROSPECTOR, 13  
 Proteinklassifikation, 400  
 Prozess, 245  
 — erfolgreicher, 245  
 — fehlgeschlagener, 245  
 — geschlossener, 245  
 Prozessbaum, 248  
 PRS, *siehe* Procedural Reasoning System  
 Qualifikationsproblem, 22, 309  
 Quantor, 52  
 Quantorenunverträglichkeit, 56  
 R-STRIPS, 316  
 R1, *siehe* XCON  
 Rahmen, 412  
 Rahmenaxiom, 308  
 Rahmenproblem, 308, 318, 329  
 ramification problem, *siehe* Verzweigungsproblem  
 Randverteilung, 435  
 Rapid Prototyping, 19  
 Reaktivität, 335  
 Rechnungsprüfung, 298  
 Redukt, 283  
 Reflexivität, 228, 299  
 Regel, 71, 270  
 — Assoziationsregel, 146  
 — nichtmonotone, 206  
 — probabilistische, 397  
 — Produktionsregel, 71  
 — Relevanz einer  $\sim$ , 147  
 — STRIPS-Regel, 313  
 Regelnetzwerk, 78  
 Regelumformungen, 73  
 Reinstantiierung, 197  
 Relevanz, 142, 147, 174  
 Relevanzmatrix, 193  
 Residuum, 380  
 Resolution, 64  
 — possibilistische, 426  
 — SLD-Resolution, 274  
 Resolutionskalkül, 64  
 Resolutionsliteral, 64  
 Resolutionsregel, 64, 65  
 Resolvente, 64  
 Retrieve, 165  
 Revision, 166  
 RHINO, 331  
 RIP, *siehe* running intersection property  
 RoboCup, 331  
 Roboter, 267, 302, 325, 331  
 Rückwärtsverkettung, 78, 83  
 — im prädikatenlogischen Fall, 84



- Rückwärtssuche, 315  
 Rumpf (einer Regel), 281  
 Rumpfliterale, 281  
 run, *siehe* Lauf  
 running intersection property, 461  
  
 Satz von Bayes, 438, 439  
 Satz von der totalen Wahrscheinlichkeit, 437  
 Schichtenarchitektur, 346  
 Schließen  
 — abduktives, 23, 259  
 — analoges, 161  
 — deduktives, 23, 26  
 — fallbasiertes, 155  
 — hypothetisches, 259  
 — induktives, 23  
 — menschliches, 20  
 — monotonen, 204  
 — nichtmonotonen, 27, 205  
 — plausibles, 412  
 — probabilistisches, 27  
 — revidierbares, 27, 205  
 — unsicheres, 26  
 Schnitteigenschaft, 228, 279, 299  
 — fortlaufende, 461  
 Schnittgraph, 464  
*see*, 340, 349  
 Sehne, 455  
 Selbstverpflichtung, 352  
 Selektionsfunktion, 278  
 Semantik, 21  
 — Antwortmengen~, 289  
 — stabile, 284  
 — wohlfundierte, 280  
 semantische Ersetzbarkeit, 45  
 semi-entscheidbar, 41  
 Semi-Monotonie, 255, 257, 262  
 Sensor, 331, 418  
 Separation, 361, 458  
 Separator, 380, 461  
 Sequenzanalyse, 145  
 SHRDLU, 14  
 Sicherheitsfaktor, 28, 89, 91  
 Signatur, 29  
 — aussagenlogische, 42  
 — prädikatenlogische, 48  
 Signifikanz, 150  
 Situation, 304, 306  
 Situationsabstraktion, 324  
 Situationskalkül, 304  
 $SJ(n_j)$ , 213  
 Skolemfunktion, 61  
 Skolemisierung, 60  
 Skolemkonstante, 60  
 SLD-Ableitung, 275, 312  
 SLD-Baum, 278  
 SLD-Beweis, 275  
 SLD-Resolution, 66, 274  
 — Antwortvollständigkeit, 277  
 — Korrektheit, 277  
 — Vollständigkeit, 277  
 SMODELS, 298, 324  
 Sorte, 67  
 soziale Kompetenz, 335  
 Space Shuttle, 331  
 “spezieller-als”-Relation, 124  
 Spezifität, 425  
 SPIRIT, 398  
*split info*, 117  
 stabile Semantik, 284  
 stabiles Modell, 284  
 stratifiziert, 286  
 STRIPS, 312, 350  
 STRIPS-Annahme, 318  
 STRIPS-Datenbasis, 312  
 STRIPS-Operator, 313  
 STRIPS-Regel, 313  
 Substitution, 62, 197  
 Subsumptionsarchitektur, 346  
 $\text{sup}(M)$ , 411  
 $\text{Supp}(n)$ , 213  
 $\text{Supp}^*(n)$ , 214  
*support*, 146  
 Support  
 — einer Assoziationsregel, 146  
 — einer Itemmenge, 146  
 SWAN, 394  
 Syntax, 21  
 System  
 — klassisch-logisches, 34  
 — logisches, 28, 33  
 Szenario, 260  
 — maximales, 260  
  
 t-Conorm, 422  
 t-Norm, 422  
 Tautologie, 35  
 Tautologieregeln, 46  
 TDIDT, *siehe* Top-Down Induction of Decision Trees  
 Term, 50  
 $\text{Term}_\Sigma$ , 50  
 $\text{Term}_\Sigma(V)$ , 50  
 Termauswertung, 51  
 Testmenge, 114

- Theorie, 37  
 — Default-Theorie, 239  
 — Poole'sche Default-Theorie, 260  
 THEORIST, 259  
 Thermostat, 334, 340  
 TMN, *siehe* Truth Maintenance-Netzwerk  
 TMS, *siehe* Truth Maintenance-System  
 TMS-Inferenzrelation, 226  
 Top-Down Induction of Decision Trees, 115  
 Trägermenge, 49  
 Trainingsmenge, 108  
 Transaktion, 146  
 transferable belief model, 418  
 Transformation, 199  
 Triangulierung, 456  
 Truth Maintenance-Netzwerk, 208  
 Truth Maintenance-System, 207, 358  
 Türme von Hanoi, 319  
 Tweety, 256
- Umgebung, 333, 334, 339  
 — Aufgaben $\sim$ , 345  
 — charakterisierende, 229  
 — deterministische, 334  
 — diskrete, 335  
 — dynamische, 335  
 — eines ATMS, 229  
 — episodische, 335  
 — inkonsistente, 229  
 — kontinuierliche, 335  
 — offene, 335  
 — statische, 335  
 — zugängliche, 334  
 Umgebungsverband, 230  
 Unabhängigkeit  
 — bedingte, 443  
 — statistische, 442  
 Unabhängigkeitsgraph, 364  
 — minimaler, 365  
 unentscheidbar, 41  
 unerfüllbar, 35  
 Unifikation, 62  
 Unifikator, 63  
 — allgemeinsten, 63  
 Universum, 49
- Vagheit, 27, 420  
 Variable  
 — freie, 53  
 — gebundene, 53  
 Variablenbelegung, 51  
 Variablenumbenennung, 53, 63  
 Verbindungsbaum, 461  
 Verbindungsgraph, 463  
 Verbundentropie, 449  
 verhaltensorientiert, 346  
 Verhandlungen, 357  
 Versionenraum, 126  
 Versionenraum-Lernverfahren, 128  
 Verteilung  
 — marginale, 435  
 — Wahrscheinlichkeitsverteilung, 432  
 Verzweigungsproblem, 309  
 VISTA, 394  
 Vokabular, *siehe* Indexvokabular  
 Vollkonjunktion, 431  
 Vollständigkeit  
 — eines Kalküls, 39  
 — eines Konzepts, 121  
 — Widerlegungsvollständigkeit, 40  
 Vorfahren, 214, 452  
 Vorfahrenmenge, 453  
 Vorwärtssuche, 315  
 Vorwärtsverkettung, 78, 81  
 VS, 130
- wahrheitsfunktional, 32, 34, 359  
 Wahrheitstafel, 44, 47  
 Wahrheitswert, 31, 34  
 — einer atomaren Formel, 51  
 — einer aussagenlogischen Formel, 44  
 — einer quantifizierten Formel, 52  
 Wahrheitswertefunktion, 34  
 Wahrnehmung, 340  
 Wahrnehmungsrahmen, 412  
 Wahrscheinlichkeit, 27, 430  
 — bedingte, 436  
 — marginale, 435  
 — Satz von der totalen  $\sim$ , 437  
 Wahrscheinlichkeitsfunktion, 430  
 Wahrscheinlichkeitsschranke  
 — obere, 411  
 — untere, 411  
 Wahrscheinlichkeitsverteilung, 432  
 Warenkorbanalyse, 150  
 web mining, 143  
 Weg, 452  
 — einfacher, 452  
 well-founded semantics, *siehe* wohlfundierte Semantik  
 Widerlegungsvollständigkeit, 40, 65  
 Widerspruchsknoten, 218  
 Windows, 394  
 Wissen  
 — evidentielles, 17  
 — fallspezifisches, 17

- regelhaftes, 17
- subjektives, 343, 348, 351
- Wissensakquisition, 16
- Wissensbasis, 17
- Wissenseinheit, 95
- Wissenserwerbskomponente, 18
- Wissensrevision, 235, 354
- Wissensverarbeitung, 17
- Wissensverarbeitungskomponente, 18
- wohlfundierte Semantik, 280
- Wunsch, 336, 351
  
- XCON, 14
  
- Zeitreihenanalyse, 145
- Zerlegung, 459
  - eigentliche, 459
- Ziel, 307, 310
  - mögliches, 351
- Zielbeschreibung, 307
- Zielklausel, 270
  - abgeleitete, 275
- Zielkonflikt, 322, 352
- Zielkonzept, 122
- Zielorientiertheit, 333
- Zirkumskription, 67
- Zustand, 281, 304, 306
  - epistemischer, 336
  - geschlossener, 283
  - innerer, 336
  - Umgebungs~, 337
- Zustandstransformationsfunktion, 338
- Zyklus, 452