Beweissysteme für die Aussagenlogik

Automatisches Beweisen

Ein **Beweissystem** $\mathfrak{B} = (\Sigma, f)$ für eine Sprache L ist eine effizient berechenbare surjektive Funktion

$$f : \Sigma^* \to L.$$

- Für $w \in L$ und $f(b) = w$ heißt b ein **Beweis** von w.
- Gibt es stets Beweise polynomieller Länge für Tautologien?

- Wenn ja, dann rate einen Beweis nichtdeterministisch und verifiziere deterministisch.
  - Übungsaufgabe: Die Bestimmung von Tautologien ist a ein (unter polynomiellen Reduktionen) schwierigstes Problem in $\mathrm{coNP} = \{L : \overline{L} \in \mathrm{NP}\}$.
  - Übungsaufgabe: Ein coNP-vollständiges Problem liegt in $\mathrm{NP} \Longrightarrow \mathrm{coNP} = \mathrm{NP}$ folgt.

- Das Ergebnis (der Beweis ist ebenfalls als Übungsaufgabe gestellt): $\mathrm{NP} = \mathrm{coNP} \Longleftrightarrow$ es gibt ein Beweissystem, so dass jede Tautologie τ in polynomieller Länge (in τ) beweisbar ist.

Frege Systeme

Schlussregeln

Wahrheitstafeln bei vielen Variablen sind viel zu groß! Was tun?

$\phi_1(X_1, \ldots, X_k), \ldots, \phi_\ell(X_1, \ldots, X_k)$ und $\psi(X_1, \ldots, X_k)$ seien aussagenlogische Formeln mit den Variablen $X_1, \ldots, X_k$.

(a) Man nennt

$$S := \frac{\phi_1(X_1, \ldots, X_k), \ldots, \phi_\ell(X_1, \ldots, X_k)}{\psi(X_1, \ldots, X_k)}$$

eine **Schlussregel**, falls

$$\phi_1 \wedge \cdots \wedge \phi_\ell \models \psi.$$

(b) Eine Schlussregel S mit $l = 0$ heißt ein **Axiomenschemata**.

(c) Wenn wir S auf beliebige aussagenlogische Formeln $\alpha_1, \ldots, \alpha_k$ anwenden, erhalten wir den **aus S abgeleiteten Schluss**

$$S := \frac{\phi_1(\alpha_1, \ldots, \alpha_k), \ldots, \phi_\ell(\alpha_1, \ldots, \alpha_k)}{\psi(\alpha_1, \ldots, \alpha_k)}$$

Der Modus Ponens

Modus Ponens und Modus Tollens

- Die Schlussregel S des **Modus Ponens** hat die Form

$$\frac{X,\ X \to Y}{Y}$$

  In einer Anwendung folgt α_2 aus α_1 und $\alpha_1 \to \alpha_2$. Man schreibt auch

$$\frac{\alpha_1,\ \alpha_1 \to \alpha_2}{\alpha_2}$$

- Die Schlussregel S des **Modus Tollens** hat die Form

$$\frac{\neg Y,\ X \to Y}{\neg X}$$

  In einer Anwendung folgt $\neg\alpha_1$ aus $\neg\alpha_2$ und $\alpha_1 \to \alpha_2$. Man schreibt auch

$$\frac{\neg\alpha_2,\ \alpha_1 \to \alpha_2}{\neg\alpha_1}$$

Aussagenlogische Beweissysteme

(a) Ein **aussagenlogisches Beweissystem** $\mathfrak{B}$ besteht aus einer Menge von Schlussregeln bzw. Axiomenschemata.

(b) Eine rekursive Definition der **Ableitbarkeit**

$$\Phi \vdash_{\mathfrak{B}} \phi$$

einer Formel ϕ aus einer Menge Φ von Formeln.

Basisregeln:

- Ein **Axiom**, also eine aus einem Axiomenschemata $S \in \mathfrak{B}$ abgeleitete Formel χ, ist aus Φ in $\mathfrak{B}$ ableitbar, es gilt also $\Phi \vdash_{\mathfrak{B}} \chi$.
- Alle **Hypothesen** $\chi \in \Phi$ sind aus Φ in $\mathfrak{B}$ ableitbar, es gilt also $\Phi \vdash_{\mathfrak{B}} \chi$.

Rekursive Regeln:

- Wenn $\Phi \vdash_{\mathfrak{B}} \chi_i$ für $i = 1, \ldots, \ell$ gilt und wenn die Formel χ nach Anwendung einer Schlussregel $S \in \mathfrak{B}$ aus $\chi_1, \ldots, \chi_\ell$ folgt, dann gilt $\Phi \vdash_{\mathfrak{B}} \chi$.

Beweise: Eine nicht-rekursive Definition

Um eine Ableitung

$$\Phi \vdash_{\mathfrak{B}} \chi$$

nachzuweisen, gibt man einen **Beweis**

$$(\chi_1, \chi_2, \ldots, \chi_i, \ldots, \chi_n)$$

an, der aus aussagenlogischen Formeln besteht.

Der Beweis $(\chi_1, \chi_2, \ldots, \chi_i, \ldots, \chi_n)$ muss die folgenden Eigenschaften besitzen:

(a) χ_n ist die Zielformel, d.h. es gilt $\chi_n = \chi$.

(b) Für jedes i mit $1 \leqslant i \leqslant n$ ist
- χ_i ist ein Axiom von $\mathfrak{B}$ oder
- χ_i gehört zur Formelmenge Φ oder
- es gibt $1 \leqslant i_1 < \cdots < i_\ell < i$ und χ_i folgt nach Anwendung einer Schlussregel $S \in \mathfrak{B}$ aus $\chi_{i_1}, \ldots, \chi_{i_\ell}$.

Soundness

Sei $\mathfrak{B}$ ein Beweissystem. Sei χ eine aussagenlogische Formel und Φ eine beliebige Menge aussagenlogischer Formeln. Dann gilt

$$\Phi \vdash_{\mathfrak{B}} \chi \implies \Phi \models \chi.$$

„Ableitbarkeit impliziert semantische Folgerung.“

Beweis: Man zeigt die Behauptung durch eine vollständige Induktion über die Länge eines Beweises $(\chi_1, \ldots, \chi_n)$ für χ.

Frege Systeme

(a) Ein aussagenlogisches Beweissystem $\mathfrak{B}$ heißt **vollständig**, wenn für alle endlichen Mengen Φ von Formeln und für jede Formel ψ gilt

$$\Phi \vdash \psi \iff \Phi \models \psi.$$

(b) Ein vollständiges aussagenlogisches Beweissystem $\mathfrak{B}$ heißt eine **Frege System**, wenn $\mathfrak{B}$ nur aus *endlich vielen* Schlussregeln besteht.

Gibt es gute und schlechte Frege Systeme?

p-Simulation

$\mathfrak{B}_1 = (\Sigma_1, f_1)$ und $\mathfrak{B}_2 = (\Sigma_2, f_2)$ seien Beweissysteme für eine Sprache L.

(a) Wir sagen, dass $\mathfrak{B}_2$ das Beweissystem $\mathfrak{B}_1$ **p-simuliert** (kurz: $\mathfrak{B}_1 \leqslant_p \mathfrak{B}_2$) $:\Longleftrightarrow$ es gibt ein Polynom $q(x)$, so dass

es für jeden Beweis $x \in \Sigma_1^*$ einen Beweis $y \in \Sigma_2^*$ gibt mit $f_1(x) = f_2(y)$ und $|y| \leqslant q(|x|)$.

(b) $\mathfrak{B}_1$ und $\mathfrak{B}_2$ sind **$\mathfrak{p}$-äquivalent**, wenn $\mathfrak{B}_1 \leqslant_p \mathfrak{B}_2$ und $\mathfrak{B}_2 \leqslant_p \mathfrak{B}_1$ gilt.

Wir zeigen: Je zwei Frege Systeme sind p-äquivalent $\Longrightarrow$
Alle Frege Systeme sind zumindest im Hinblick auf Beweislänge ununterscheidbar.

Substitutionslemma

Sei $\mathfrak{B}$ ein aussagenlogisches Beweissystem und es gelte

$$\Phi(X_1, \ldots, X_k) \vdash_{\mathfrak{B}} \phi(X_1, \ldots, X_k)$$

für eine Formelmenge Φ und eine Formel ϕ $\quad \Longrightarrow$

$$\Phi(\alpha_1, \ldots, \alpha_k) \vdash_{\mathfrak{B}} \phi(\alpha_1, \ldots, \alpha_k)$$

für alle Formeln $\alpha_1, \ldots, \alpha_k$. Die Beweislänge ändert sich nicht.

Beweis: Übungsaufgabe.

$\mathfrak{B}_1$ und $\mathfrak{B}_2$ seien Frege-Systeme

Es genügt zu zeigen, dass $\mathfrak{B}_1$ durch $\mathfrak{B}_2$ p-simuliert werden kann.

1. Sei dazu $S \in \mathfrak{B}_1$ eine Schlussregel mit

$$S = \frac{\phi_1(X_1, \ldots, X_k), \ldots, \phi_\ell(X_1, \ldots, X_k)}{\psi(X_1, \ldots, X_k).}$$

   Dann folgt $\{\phi_1, \ldots, \phi_\ell\} \models \psi$.
2. Aber $\mathfrak{B}_2$ ist ein Frege System und deshalb gilt $\{\phi_1, \ldots, \phi_\ell\} \vdash_{\mathfrak{B}_2} \psi$ mit einem Beweis der Länge N_S.
3. Mit dem „Substitutionslemma“ zeige für beliebige Formeln $\alpha_1, \ldots, \alpha_k$:

$$\{\phi_1(\alpha_1, \ldots, \alpha_k), \ldots, \phi_\ell(\alpha_1, \ldots, \alpha_k)\} \vdash_{\mathfrak{B}_2} \psi(\alpha_1, \ldots, \alpha_k)$$

   mit einem Beweis der Länge ebenfalls N_S.

Die Graphstruktur von Beweisen

Beweise mit Baumstruktur

Sei $\chi = (\chi_1, \chi_2, \ldots, \chi_j, \ldots, \chi_i, \ldots, \chi_n)$ ein Beweis in einem aussagenlogischen Beweissystem $\mathfrak{B}$.

(a) χ definiert den gerichteten Graphen $G(\chi) = (V, E)$ mit $V := \{\chi_i : 1 \leqslant i \leqslant n\}$ und $(\chi_j, \chi_i) \in E$ genau dann, wenn χ_i durch eine Schlussregel abgeleitet wird und wenn χ_j in dieser Schlussregel auftaucht.

(b) Der Beweis χ hat **Baumstruktur** genau dann wenn $G(\chi)$ ein Baum ist.

Übungsaufgabe: Sei ϕ eine aussagenlogische Formel. Zeige: Zu jedem Beweis für ϕ in einem Beweissystem gibt es einen Beweis für ϕ mit Baumstruktur.

Sogar: Zu jedem Beweis in einem **Frege System** gibt es einen Beweis mit Baumstruktur, dessen Länge **proportional** zum ursprünglichen Beweis ist.

Mächtigere Beweissysteme: Definitionen

Sei $\mathfrak{B}$ ein aussagenlogisches Beweissystem.
Dann heißt $D(\mathfrak{B})$ die **Definitionserweiterung** von $\mathfrak{B}$, wenn $D(\mathfrak{B})$ genau aus den Schlussregeln von $\mathfrak{B}$ besteht und zusätzlich *Definitionen* zulässt.
Ein Tupel $(\chi_1, \chi_2, \ldots, \chi_i, \ldots, \chi_n)$ ist genau dann ein Beweis von

$$\Phi \vdash_{D(\mathfrak{B})} \chi_n$$

in $D(\mathfrak{B})$, wenn für jedes i mit $1 \leqslant i \leqslant n$ gilt

- χ_i ist ein Axiom oder
- χ_i ist eine Hypothese, d.h. χ_i gehört zu Φ oder
- es gibt $1 \leqslant i_1 < \cdots < i_\ell < i$ und χ_i folgt aus $\chi_{i_1}, \ldots, \chi_{i_\ell}$ mit Hilfe einer Schlussregel in $\mathfrak{B}$ oder
- χ_i ist eine **Definition**, d.h. es gibt j mit $j < i$ und $\chi_i = (X \leftrightarrow \chi_j)$, wobei X weder in einer der Formeln $\chi_1, \ldots, \chi_{i-1}$ noch in der Zielformel χ_n auftaucht.

Mächtigere Beweissysteme: Ersetzungen

Sei $\mathfrak{B}$ ein aussagenlogisches Beweissystem.
Dann heißt $E(\mathfrak{B})$ die **Ersetzungserweiterung** von $\mathfrak{B}$, wenn $E(\mathfrak{B})$ genau aus den Schlussregeln von $\mathfrak{B}$ besteht und zusätzlich *Ersetzungen* zulässt.
Ein Tupel $(\chi_1, \chi_2, \ldots, \chi_i, \ldots, \chi_n)$ ist genau dann ein Beweis von

$$\Phi \vdash_{D(\mathfrak{B})} \chi_n$$

in $E(\mathfrak{B})$, wenn für jedes i mit $1 \leqslant i \leqslant n$ gilt

- χ_i ist ein Axiom oder
- χ_i ist eine Hypothese, d.h. χ_i gehört zu Φ oder
- es gibt $1 \leqslant i_1 < \cdots < i_\ell < i$ und χ_i folgt aus $\chi_{i_1}, \ldots, \chi_{i_\ell}$ mit Hilfe einer Schlussregel in $\mathfrak{B}$ oder
- χ_i folgt durch eine **Ersetzung**, d.h. $\chi_i = \chi_j(\alpha_1, \ldots, \alpha_k)$ für $j < i$.
  χ_j hängt (unter anderen) von den Variablen $X_1, \ldots, X_k$ ab, wobei X_s durch die aussagenlogische Formel α_s ersetzt wird.

Definitionen = Ersetzungen

(a) Weder Beweissysteme mit Definitionen noch Beweissysteme mit Ersetzung sind aussagenlogische Beweissysteme:

Definitionen und Ersetzungen sind keine Schlussregeln!

(b) In beiden Fällen handelt es sich um Beweissysteme $\Longrightarrow$ wahrscheinlich gibt es nicht immer Beweise (mit Definitionen bzw. Ersetzung) polynomieller Länge.

Frege Systeme mit Definitionen und Frege Systeme mit Ersetzung sind p-äquivalent.

Wir zeigen nur, dass ein Frege System mit Ersetzung jedes Frege System mit Definitionen p-simulieren kann. Dazu benötigen wir das Deduktionslemma.

Das Deduktionslemma (1/2)

Sei $\mathfrak{B}$ ein Frege System. ϕ und ψ seien aussagenlogische Formeln, Φ sei eine Menge aussagenlogischer Formeln. Wenn es einen Beweis

$$\Phi \cup \{\phi\} \vdash_{\mathfrak{B}} \psi$$

der Länge N gibt, dann gibt es einen Beweis

$$\Phi \vdash_{\mathfrak{B}} \phi \to \psi$$

der Länge proportional zu N.

Beweis: Wir definieren ein neues aussagenlogisches Beweissystem $\mathfrak{B}^*$.

1. Ersetze jede Schlussregel

$$\frac{\phi_1, \ldots, \phi_\ell}{\psi}$$

von $\mathfrak{B}$ durch die Schlussregel

$$\frac{\phi \to \phi_1, \ldots, \phi \to \phi_\ell}{\phi \to \psi.}$$

Das Deduktionslemma (2/2)

2. Füge das Axiomenschema

$$\frac{}{X \to X}$$

zu $\mathfrak{B}^*$ hinzu.

3. Sei $(\chi_1, \ldots, \chi_n)$ ein Beweis von $\Phi \cup \{\phi\} \vdash_{\mathfrak{B}} \psi$ in $\mathfrak{B}$.
   a. Beginne einen Beweis in $\mathfrak{B}^*$ mit dem Axiom $\phi \to \phi$.
   b. Übersetze den $\mathfrak{B}$-Beweis Schritt für Schritt in den entsprechenden $\mathfrak{B}^*$-Beweis: Wir erhalten den $\mathfrak{B}^*$-Beweis $(\phi \to \phi, \chi_1^*, \ldots, \chi_n^*)$ mit $\chi_i^* := \phi \to \chi_i$.
   c. Es ist $\chi_n^* := \phi \to \psi$. Wir haben einen Beweis von $\Phi \vdash_{\mathfrak{B}^*} \phi \to \psi$ der Länge $n+2$ erhalten.
4. Erweitere $\mathfrak{B}^*$ zu einem Frege System.
5. Da $\mathfrak{B}$ und $\mathfrak{B}^*$ p-äquivalent sind: Es gibt einen Beweis

$$\Phi \vdash_{\mathfrak{B}} \phi \to \psi$$

polynomieller Länge in n. □

Simuliere Definitionen durch Ersetzungen

Beweis: Sei $(\chi_1, \ldots, \chi_n)$ ein Beweis von $\Phi \vdash_{D(\mathfrak{B})} \psi$ in $D(\mathfrak{B})$.

1. O.B.d.A. enthalte $\mathfrak{B}$ die Schlussregel
$$\frac{(X \leftrightarrow X) \to Y}{Y}$$
2. Wenn wir jede Definition $X \leftrightarrow \beta$ als eine Hypothese auffassen, erhalten wir in $\mathfrak{B}$ einen „konventionellen“ Beweis
$$\Phi^* \vdash_{\mathfrak{B}} \psi$$
3. Seien $X_i \leftrightarrow \beta_i$ für $1 \leqslant i \leqslant k$ alle Definitionen im Beweis von ψ. Wenn wir das Deduktionslemma wiederholt anwenden, erhalten wir aus $\Phi^* \vdash_{\mathfrak{B}} \psi$ den Beweis
$$\Phi \vdash_{\mathfrak{B}} (X_1 \leftrightarrow \beta_1) \to \cdots \to ((X_k \leftrightarrow \beta_k) \to \psi) \cdots).$$
4. Führe diesen Beweis in $E(\mathfrak{B})$ aus und **ersetze** nachträglich jedes X_i durch β_i.
5. Entferne die Äquivalenzen $\beta_i \leftrightarrow \beta_i$ mit der Schlussregel $\dfrac{(X \leftrightarrow X) \to Y}{Y}$

   Wir erhalten den Beweis $\Phi \vdash_{E(\mathfrak{B})} \psi$ in $E(\mathfrak{B})$, dessen Länge höchstens polynomiell zugenommen hat. □

Das Frege System des Modus Ponens

Das Beweissystem des Modus Ponens

Die Junktoren $\wedge, \vee, \leftrightarrow, \oplus$ lassen sich durch Implikation und Negation ausdrücken.

$$\mathbf{AL}(\{\neg, \rightarrow\})$$

ist die Menge der Formeln, die nur aus den Junktoren $\neg$ und $\rightarrow$ aufgebaut sind. Die aussagenlogischen Konstanten $\mathbf{0}, \mathbf{1}$ dürfen nicht benutzt werden.

Das „automatische“ Beweissystem

$$\mathbf{ABS} := (A_{\mathsf{ABS}}, S_{\mathsf{ABS}})$$

des Modus Ponens:

- Die **Axiome**: Die Menge A_{ABS} besteht für alle aussagenlogischen Formeln $\phi, \chi, \psi \in \mathbf{AL}(\{\neg, \rightarrow\})$ aus den folgenden drei Klassen von Formeln:
  (1) $\phi \rightarrow (\psi \rightarrow \phi) \in A_{\mathsf{ABS}}$,
  (2) $(\phi \rightarrow (\mu \rightarrow \psi)) \rightarrow ((\phi \rightarrow \mu) \rightarrow (\phi \rightarrow \psi)) \in A_{\mathsf{ABS}}$,
  (3) $(\neg\phi \rightarrow \neg\psi) \rightarrow (\psi \rightarrow \phi) \in A_{\mathsf{ABS}}$.
- Die **Schlussregel** S_{ABS} ist der Modus Ponens.

Das Beweissystem des Modus Ponens: Ein Beispiel

Wir zeigen die „Transitivität der Implikation“, also die Ableitung

$$\{\phi \to \psi, \psi \to \chi\} \vdash_{\mathsf{ABS}} (\phi \to \chi).$$

Hier sind die einzelnen Ableitungsschritte. Setze $\Phi := \{\phi \to \psi, \psi \to \chi\}$.

1. Für $\chi_1 := (\psi \to \chi)$ gilt $\Phi \vdash_{\mathsf{ABS}} \chi_1$, denn $\psi \to \chi \in \Phi$.
2. $\chi_2 := ((\psi \to \chi) \to (\phi \to (\psi \to \chi)))$ ist ein Axiom des Typs (1).
3. Wende den Modus Ponens auf χ_1 und χ_2 an $\Longrightarrow$ $\phi_3 := (\phi \to (\psi \to \chi))$ folgt.
4. $\chi_4 := ((\phi \to (\psi \to \chi)) \to ((\phi \to \psi) \to (\phi \to \chi)))$ ist ein Axiom vom Typ 2.
5. Wende den Modus Ponens auf χ_3 und χ_4 an $\Longrightarrow$ $\chi_5 := ((\phi \to \psi) \to (\phi \to \chi))$ folgt.
6. Die Formel $\chi_6 := \phi \to \psi$ ist ableitbar, denn $\phi \to \psi \in \Phi$.
7. Die Zielformel $\chi_7 := (\phi \to \chi)$ folgt aus dem Modus Ponens angewandt auf χ_5 und χ_6.

Der Vollständigkeitssatz für ABS

Die Formelmenge $\Phi \subseteq \mathbf{AL}(\{\neg, \rightarrow\})$ und die Formel $\chi \in \mathbf{AL}(\{\neg, \rightarrow\})$ seien gegeben. Dann gilt

$$\Phi \models \chi \iff \Phi \vdash_{\mathsf{ABS}} \chi.$$

- Setze $\Phi = \emptyset$: Die Formel χ ist genau dann allgemeingültig, wenn χ aus den Axiomen in A_{ABS} mit Hilfe der Schlussregel des Modus Ponens ableitbar ist.
- Die Richtung $\Longleftarrow$ haben wir bereits für beliebige Beweissysteme gezeigt.

Zeige: Wenn $\Phi \models \chi$, dann $\Phi \vdash_{\mathsf{ABS}} \chi$.

Widerspruchsvoll und widersprüchlich

Sei $\Phi \subseteq \mathsf{AL}(\{\neg, \to\})$ eine Formelmenge.

(a) Φ heißt **widerspruchsvoll** (bzgl. ABS), wenn es eine Formel $\psi \in \mathsf{AL}(\{\neg, \to\})$ gibt mit

$$\Phi \vdash_{\mathbf{ABS}} \psi \textbf{ und } \Phi \vdash_{\mathbf{ABS}} \neg\psi.$$

Gibt es eine solche Formel ψ nicht, dann heißt Φ **widerspruchsfrei**.

(b) Φ heißt **maximal widerspruchsfrei**, wenn Φ widerspruchsfrei ist und wenn ($\psi \in \Phi$ oder $\neg\psi \in \Phi$) für alle Formeln $\psi \in \mathsf{AL}(\{\neg, \to\})$ gilt.

(c) Φ heißt **widersprüchlich**, falls es keine Belegung gibt, die alle Formeln in Ψ erfüllt.

Der kritische Schritt im Beweis des Vollständigkeitssatzes

Die Formelmenge $\Phi \subseteq \mathsf{AL}(\{\neg, \to\})$ sei widerspruchsfrei und $\psi \in \mathsf{AL}(\{\neg, \to\})$ sei gegeben.

(a) Dann ist $\Phi \cup \{\psi\}$ oder $\Phi \cup \{\neg\psi\}$ widerspruchsfrei.

(b) Zu jeder widerspruchsfreien Formelmenge $\Phi \subseteq \mathsf{AL}(\{\neg, \to\})$ gibt es eine maximal widerspruchsfreie Formelmenge Φ^* mit $\Phi \subseteq \Phi^*$.

Wir nehmen zuerst an, dass es stets maximal widerspruchsfreie Obermengen einer widerspruchsfreien Formelmenge gibt und zeigen den Vollständigkeitssatz.

Der Beweis des Vollständigkeitssatzes

Wenn $\Phi \models \chi$, dann $\Phi \vdash_{\mathsf{ABS}} \chi$.

1. Aus $\Phi \models \chi$ folgt, dass $\Phi \cup \{\neg\chi\}$ widersprüchlich ist.
2. Angenommen, $\Phi \cup \{\neg\chi\}$ ist widerspruchsfrei $\Longrightarrow$
   Es gibt eine maximal widerspruchsfreie Obermenge Φ^* von $\Phi \cup \{\neg\chi\}$.
   - Definiere die Belegung $\mathfrak{B}^*$: Für jede Variable V setze
     - $\mathfrak{B}^*(V) := 1$, wenn $V \in \Phi^*$,
     - $\mathfrak{B}^*(V) := 0$, wenn $V \notin \Phi^*$.

     $\mathfrak{B}^*$ erfüllt alle Formeln in Φ^* – vollständige Induktion über den Formelaufbau.
   - $\mathfrak{B}^*$ erfüllt $\Phi \cup \{\neg\chi\}$, denn $\Phi \cup \{\neg\chi\} \subseteq \Phi^*$ ↯
3. Also ist $\Phi \cup \{\neg\chi\}$ doch widerspruchsvoll.
   - Jede Formel kann aus $\Phi \cup \{\neg\chi\}$ gefolgert werden. $\Longrightarrow$ $\Phi \cup \{\neg\chi\} \vdash_{\mathsf{ABS}} \chi$.
   - Übungsaufgabe: Wenn $\Phi \cup \{\phi\} \vdash_{\mathsf{ABS}} \psi$, dann $\Phi \vdash_{\mathsf{ABS}} (\phi \to \psi)$.
     - Also folgt: $\Phi \vdash_{\mathsf{ABS}} (\neg\chi \to \chi)$.
4. Zeige $(\neg\psi \to \psi) \vdash_{\mathsf{ABS}} \psi$.
5. Mit der Transitivität der Implikation folgt $\Phi \vdash_{\mathsf{ABS}} \chi$. □

Existenz maximal widerspruchsfreier Formelmengen

Angenommen, $\Phi \cup \{\psi\}$ wie auch $\Phi \cup \{\neg\psi\}$ sind widerspruchsvoll.

1. $\Phi \cup \{\psi\}$ ist nach Annahme widerspruchsvoll
   $\Longrightarrow \Phi \cup \{\psi\} \vdash_{\mathsf{ABS}} \neg\psi \Longrightarrow \Phi \vdash_{\mathsf{ABS}} \psi \to \neg\psi$.
2. Ebenfalls nach Annahme ist $\Phi \cup \{\neg\psi\}$ widerspruchsvoll $\Longrightarrow \Phi \cup \{\neg\psi\} \vdash_{\mathsf{ABS}} \psi$
   - Also folgt $\Phi \vdash_{\mathsf{ABS}} (\neg\psi \to \psi)$.
   - Zeige $(\neg\psi \to \psi) \vdash_{\mathsf{ABS}} \psi$.
   - $\overset{\text{Transitivität der Implikation}}{\Longrightarrow}$ $\Phi \vdash_{\mathsf{ABS}} \psi$.
3. Wende den Modus Ponens auf ψ und $\psi \to \neg\psi$ an
   $\Longrightarrow \Phi \vdash_{\mathsf{ABS}} \neg\psi \Longrightarrow \Phi$ ist doch nicht widerspruchsfrei $\lightning$

(a) Also ist $\Phi \cup \{\psi\}$ oder $\Phi \cup \{\neg\psi\}$ widerspruchsfrei.

(b) Die Existenz einer maximal widerspruchsfreien Obermenge folgt.

Der Kompaktheitssatz

(a) Übungsaufgabe: Wenn $\Phi \vdash \phi$, dann gibt es eine endliche Teilmenge $\Phi^* \subseteq \Phi$ mit $\Phi^* \vdash \phi \implies$

(b) Jede **widerspruchsvolle** Formelmenge besitzt eine endliche Teilmenge, die **widerspruchsvoll** ist.

Der Kompaktheitssatz:

Sei Φ eine **widersprüchliche** Menge von Formeln.
Dann gibt es eine endliche Teilmenge $\Phi^* \subseteq \Phi$, die **widersprüchlich** ist.

Das KNF-Erfüllbarkeitsproblem

KNF-SAT

Im KNF-Erfüllbarkeitsproblem

kurz: **KNF-SAT**

ist für eine KNF-Formel ϕ, also für eine aussagenlogische Formel in konjunktiver Normalform, zu entscheiden, ob ϕ erfüllbar ist.

- KNF-SAT ist NP-vollständig.
+ Viele wichtige Instanzen lassen sich mit Hilfe von Heuristiken für KNF-SAT, den so genannten SAT-Solvern, lösen.
+ KNF-SAT hat als NP-vollständiges Problem viele Anwendungen.

Model Checking

Model Checking: Überprüfen, ob eine Implementierung für jede Instanz eine vorgebene Spezifikation einhält.

1. Häufig gibt es einen effizienten Algorithmus A, der eine Instanz x genau dann akzeptiert, wenn sich Implementierung und Spezifikation für x unterscheiden :

   Die Implementierung hält die Spezifikation immer ein $\iff L(A) = \emptyset$.

2. Angenommen, wir sind nur an Instanzen $x \in \{0,1\}^n$ interessiert:
   - Baue eine KNF ϕ, so dass

     $\phi(x)$ ist wahr $\iff A$ akzeptiert x.

   - Die Implementierung hält die Spezifikation für alle Eingaben in $\{0,1\}^n$ ein $\iff \phi$ ist nicht erfüllbar.

Widersprüchlichkeit von KNFs: Was ist zu erwarten?

Für ein geeignetes Beweissystem: Gibt es polynomiell lange Beweise für die Widersprüchlichkeit von KNF-Formeln?

> Wir müssen uns ja nicht mit allen möglichen aussagenlogischen Formeln herumplagen, sondern nur mit KNFs.

$\mathrm{NP} = \mathrm{coNP} \iff$ es gibt ein Beweissystem, so dass jede unerfüllbare KNF-Formel ϕ einen Widerspruchsbeweis polynomieller Länge (in der Länge von ϕ) besitzt.

Beweis: Übungsaufgabe.

Resolution:
Ein zentraler Baustein in vielen SAT-Solvern

Die Resolutionsregel

Stelle einen Disjunktionsterm (bzw Klausel)

$$D = \ell_1 \vee \cdots \vee \ell_m$$

als die Menge

$$\{\ell_1, \ldots, \ell_m\}$$

seiner Literale dar. Bezeichne den leeren Disjunktionsterm mit ϵ.

Sei X eine Variable, $\alpha' = \alpha \cup \{X\}$ und $\beta' = \beta \cup \{\neg X\}$ seien Disjunktionsterme. Mit der **Resolutionsregel**

$$\frac{\alpha \cup \{X\},\ \beta \cup \{\neg X\}}{\alpha \cup \beta}$$

darf der Disjunktionsterm $\alpha \vee \beta$ abgeleitet werden.

Wir sagen, dass $\alpha \cup \beta$ eine Anwendung der Resolutionsregel zu X ist.

Die Resolutionsregel ist „sound“, denn $\{\, \alpha \cup \{X\},\ \beta \cup \{\neg X\} \,\} \models \alpha \cup \beta$.

Das Beweisverfahren der Resolution

(a) Das **Beweissystem $\mathfrak{R}$ der Resolution** besitzt die Resolutionsregel als einzige Schlussregel.

(b) Für eine Menge Φ von Disjunktionstermen und einen Disjunktionsterm χ führt man den Begriff

$$\Phi \vdash_{\mathfrak{R}} \chi$$

eines Resolutionsbeweises rekursiv wie folgt ein:

Basisregel: Für jede Hypothese $\alpha \in \Phi$ ist $\Phi \vdash_{\mathfrak{R}} \alpha$.

Rekursive Regel: Wenn $\Phi \vdash_{\mathfrak{R}} \alpha \cup \{X\}$ und $\Phi \vdash_{\mathfrak{R}} \beta \cup \{\neg X\}$, dann gilt auch $\Phi \vdash_{\mathfrak{R}} \alpha \cup \beta$.

(c) Ein **Resolutionsbeweis** für die Ableitung $\Phi \vdash_{\mathfrak{R}} \chi$ ist ein Tupel $(\chi_1, \ldots, \chi_i, \ldots, \chi_n)$ von Disjunktionstermen. Es muss gelten:

(a) χ_n ist die Zielformel, d.h. es gilt $\chi_n = \chi$.

(b) Für jedes i mit $1 \leqslant i \leqslant n$ ist entweder

- χ_i eine Hypothese, also eine Formel aus Φ oder
- es gibt $1 \leqslant i_1 < i_2 < i$ und χ_i folgt aus χ_{i_1}, χ_{i_2} durch Anwendung der Resolutionsregel.

KNF-SAT mit Resolutionsbeweisen lösen?!

Ist die KNF $\phi = D_1 \wedge D_2 \wedge \cdots \wedge D_m$ erfüllbar?

Sei $\Phi := \{ D_1, \ldots, D_m \}$.

:-)) Wir zeigen den **Vollständigkeitssatz für die Resolution**:
Sei Φ eine Menge von Disjunktionstermen. Dann gilt

$$\left(\bigwedge_{D \in \Phi} D \right) \text{ ist unerfüllbar} \iff \Phi \vdash_{\mathfrak{R}} \epsilon.$$

.

Achtung: $\{X\} \nvdash_{\mathfrak{R}} \{X, Y\}$.

:-((Aber es gibt widersprüchliche Formelmengen Φ, wie das **Schubfachprinzip**, die nur exponentiell lange Resolutionsbeweise besitzen!

- Für einige Formeln gibt es nur sehr, sehr lange Resolutionsbeweise.

  Traurig, aber nicht wirklich überrraschend.

+ Moderne Resolutionsverfahren finden aber für „viele“ Formeln kurze Beweise.

Resolution: Ein erstes Beispiel

Zeige die „Transitivität“ der Implikation, d.h. **zeige**

$$\Phi \vdash_{\mathfrak{R}} \chi$$

mit

$$\Phi := \{\{\neg X, Y\}, \{\neg Y, Z\}\} \text{ und } \chi := \{\neg X, Z\}.$$

1. Die Formel $\chi_1 := \{\neg X, Y\}$ gehört zu Φ und ist deshalb in $\mathfrak{R}$ ableitbar.
2. Gleiches gilt für die Formel $\chi_2 := \{\neg Y, Z\}$ und deshalb ist χ_2 ableitbar.
3. Jetzt erhalten wir $\chi_3 := \{\neg X, Z\} = \chi$ nach Anwendung der Resolutionsregel

$$\frac{\{\neg X, Y\}, \{\neg Y, Z\}}{\{\neg X, Z\}}$$

auf χ_1 und χ_2.

Resolution und KNF-SAT

Resolution: Ein zweites Beispiel

1. Die Kunden der Bahn sind nicht **z**ufrieden, wenn
   - sich die **P**reise erhöhen: $P \to \neg Z$ folgt,
   - oder sich die **F**ahrzeiten verlängern, d.h. $F \to \neg Z$ ist die Konsequenz.
2. Wenn der Frankfurter Kopfbahnhof nicht in einen Durchgangsbahnhof umge**b**aut wird, verlängern sich die Fahrzeiten, also gilt $\neg B \to F$.
3. Der Bahnhof kann nur dann umgebaut werden, wenn die Fahrpreise erhöht werden, d.h. $B \to P$ folgt.

Die Bahn kann es niemandem recht machen, denn die Formelmenge

$$\Phi := \Big\{ \{\neg P, \neg Z\},\ \{\neg F, \neg Z\},\ \{B, F\},\ \{\neg B, P\},\ \{Z\} \Big\}$$

ist unerfüllbar.

Wie sieht ein Resolutionsbeweis $\Phi \vdash_{\mathfrak{R}} \epsilon$ aus?

Die Resolution ist vollständig

Vollständigkeit der Resolution (1/5)

Sei Φ eine Menge von Disjunktionstermen. Dann gilt

$$\left(\bigwedge_{D \in \Phi} D \right) \text{ ist unerfüllbar} \iff \Phi \vdash_{\mathfrak{R}} \epsilon.$$

Wir beschreiben das **Davis-Putnam-Verfahren**, das nacheinander Variablen mit Hilfe der Resolutionsregel entfernt bis Erfüllbarkeit festgestellt wird oder der leere Disjunktionsterm abgeleitet wird.

Vollständigkeit der Resolution (2/5)

Das **Davis-Putnam Verfahren**:

Setze $k := 0$ und $\Phi_0 := \Phi$. Wiederhole:

1. Entferne alle *allgemeingültigen* Disjunktionsterme D aus Φ_k.
   (D ist genau dann allgemeingültig, wenn $D = D' \cup \{X, \neg X\}$ gilt.)
2. Die **Unit-Clause-Regel**: Wenn ein Literal l ein Disjunktionsterm in Φ_k ist, dann entferne $\neg l$ aus allen Disjunktionstermen und entferne alle Disjunktionsterme aus Φ_k, in denen l auftaucht. (l muss auf `wahr` gesetzt werden.)
3. Die **Pure-Literal-Regel**: Wenn es ein Literal l gibt, so dass $\neg l$ in keinem Disjunktionsterm vorkommt, dann entferne alle Disjunktionsterme aus Φ_k in denen l vorkommt. (O.B.d.A. kann l auf wahr gesetzt werden.)
4. Wenn $\Phi_k = \emptyset$, dann halte mit der Antwort „Φ ist erfüllbar“.
   Wenn $\epsilon \in \Phi_k$, dann halte mit der Antwort „Φ ist unerfüllbar“.

Vollständigkeit der Resolution (3/5)

5. Wähle eine aussagenlogische Variable X, die in mindestens einem Disjunktionsterm in Φ_k vorkommt.
   - 5.1 Wende, wann immer möglich, die Resolutionsregel zu X an, d.h. bestimme

$$\Psi_1 := \{\alpha \cup \beta \,:\, \alpha \cup \{X\} \in \Phi_k, \beta \cup \{\neg X\} \in \Phi_k\}$$

   - 5.2 Nimm all diese Anwendungen der Resolutionsregel in Φ_k auf und entferne aus Φ_k alle Disjunktionsterme, die das Literal X, bzw. das Literal $\neg X$ enthalten. D.h. bestimme

$$\Psi_2 := \{\alpha \,:\, X \in \mathrm{Var}(\alpha) \text{ und } \alpha \in \Phi_k\}$$

     und setze

$$\Phi_{k+1} := (\Phi_k \cup \Psi_1) \setminus \Psi_2.$$

     Achtung: Die Menge Φ_{k+1} nimmt möglicherweise gewaltig an Größe zu!
   - 5.3 Setze $k := k + 1$.

Vollständigkeit der Resolution (4/5)

Beweis: Das Verfahren terminiert, da nacheinander alle Variablen entfernt werden.

Wir zeigen : Φ_k erfüllbar $\iff$ Φ_{k+1} erfüllbar.

„$\Longrightarrow$“ Die Belegung $\mathfrak{B}$ erfülle alle Disjunktionsterme in Φ_k.

- Ein Disjunktionsterm $D \in \Phi_{k+1}$ gehört entweder zu Φ_k, und wird dann natürlich von $\mathfrak{B}$ erfüllt. ✓
- Oder aber D wurde neu hinzugefügt.
  - Dann ist $D = \alpha \cup \beta \Longrightarrow \alpha \cup \{X\}$ wie auch $\beta \cup \{\neg X\}$ gehören zu Φ_k.
  - Nach Annahme erfüllt $\mathfrak{B}$ die Disjunktionsterme $\alpha \cup \{X\}$ und $\beta \cup \neg X$ $\Longrightarrow$ $\mathfrak{B}$ erfüllt $\alpha \cup \beta$.

Also erfüllt $\mathfrak{B}$ alle Disjunktionsterme in Φ_{k+1}.

Vollständigkeit der Resolution (5/5)

„$\Longleftarrow$“ Die Belegung $\mathfrak{B}$ erfülle alle Disjunktionsterme in Φ_{k+1}.
Zeige, dass Φ_k erfüllbar ist.

O.B.d.A. falsifiziere $\mathfrak{B}$ die Variable X. Sei $\mathfrak{B}'$ die „**Geschwisterbelegung**“, die X erfülle, sonst aber mit $\mathfrak{B}$ übereinstimme.

- Wenn ein Disjunktionsterm $D \in \Phi_k$ bereits zu Φ_{k+1} gehört, dann wird D natürlich von $\mathfrak{B}$ wie auch von $\mathfrak{B}'$ erfüllt. ✓
- Oder aber D wurde aus Φ_k entfernt.
  - Wenn $\mathfrak{B}$ alle entfernten Disjunktionsterme erfüllt, dann ist Φ_k erfüllbar. ✓
  - Also falsifiziere $\mathfrak{B}$ den entfernten Disjunktionsterm D und $D = \alpha \cup \{X\}$ folgt.
    - $\mathfrak{B}$ (und damit auch $\mathfrak{B}'$) erfüllt alle Disjunktionsterme β mit $\beta \cup \{\neg X\} \in \Phi_k$, denn sonst würde $\mathfrak{B}$ den Disjunktionsterm $\alpha \cup \beta \in \Phi_{k+1}$ falsifizieren.
    - Aber dann erfüllt $\mathfrak{B}'$ sowohl $\alpha \cup \{X\}$ wie auch $\beta \cup \{\neg X\}$ für alle entfernten Disjunktionsterme.
    - $\mathfrak{B}'$ erfüllt alle Terme in Φ_k und Φ_k ist erfüllbar. ✓

SAT-Solver: Varianten des Davis-Putnam-Logemann-Loveland Algorithmus

Der DPLL-Algorithmus: Die Grobstruktur (1/2)

Wiederhole:

1. Entferne alle allgemeingültigen Disjunktionsterme D aus Φ.
   (D ist genau dann allgemeingültig, wenn $D = D' \cup \{X, \neg X\}$ gilt.)
2. Die **Unit-Clause-Regel**: Wenn ein Literal l ein Disjunktionsterm in Φ ist, dann entferne $\neg l$ aus allen Disjunktionstermen und entferne alle Disjunktionsterme aus Φ_k in denen l auftaucht. (O.B.d.A. setze l auf `wahr`.)
3. Die **Pure-Literal-Regel**: Wenn es ein Literal l gibt, so dass $\neg l$ in keinem Disjunktionsterm vorkommt, dann entferne alle Disjunktionsterme aus Φ in denen l vorkommt. (O.B.d.A. kann l auf wahr gesetzt werden.)
4. Wenn $\Phi = \emptyset$, dann halte mit der Antwort „Φ ist erfüllbar“. Wenn Φ den leeren Disjunktionsterm ϵ enthält, dann halte mit der Antwort „Φ ist unerfüllbar“.

Der DPLL-Algorithmus: Die Grobstruktur (2/2)

5. „**Choose Literal**“: Wähle eine aussagenlogische Variable X, die in mindestens einem Disjunktionsterm in Φ_k vorkommt. Wende die **Splitting-Regel** an:
   - Setze $X = 1$, entferne alle erfüllten Disjunktionsterme und entferne jedes Vorkommen von $\neg X$ in einem Disjunktionsterm.
     Rufe das DPLL-Verfahren rekursiv für die neue Menge Φ auf.
   - Wenn Antwort „unerfüllbar“, setze $X = 0$, entferne alle erfüllten Disjunktionsterme und entferne jedes Vorkommen von X in einem Disjunktionsterm.

     Rufe das DPLL-Verfahren rekursiv für die neue Menge Φ auf.

Zu den erfolgreichsten Implementierungen des DPLL-Verfahrens gehören Chaff (`http://www.princeton.edu/~chaff/`) und zChaff (`http://www.princeton.edu/~chaff/zchaff.html`).

Der DPLL-Algorithmus und Resolution

ϕ sei eine widersprüchliche KNF-Formel.

Ein **DPLL-Baum** T für ϕ ist ein Rekursionsbaum des DPLL-Verfahrens für Eingabe ϕ mit den folgenden Markierungen.

1. Beschrifte einen inneren Knoten v von T mit
   (a) der aussagenlogischen Variablen X, die im Aufruf von v entfernt wird und
   (b) mit der Angabe, ob die `Unit-Clause`-Regel, die `Pure-Literal`-Regel oder die `Splitting`-Regel angewandt wurde.

   Beschrifte die von v ausgehenden Kanten mit $X = 0$ bzw. $X = 1$ wie verlangt.
2. Beschrifte ein Blatt b von T mit einem Disjunkionsterm, der durch die Belegung des Weges von der Wurzel nach b falsifiziert wird.

Der DPLL-Baum T für ϕ definiert einen Resolutionbeweis b_T dessen Länge beschränkt ist durch die Anzahl der Knoten von T.

Die Komplexität der Resolution und das Schubfachprinzip

Die Komplexität der Resolution

Zur Erinnerung:

(a) Ein **Beweissystem** für eine Sprache L ist eine effizient berechenbare Funktion $B : \Sigma^* \to \Sigma^*$ mit $B(\Sigma^*) = L$.
Für $w \in L$ und $\mathbf{B(b) = w}$ heißt b ein **Beweis** von w.

(b) $\mathrm{NP} = \mathrm{coNP} \Leftrightarrow$ es gibt ein Beweissystem, so dass jede Tautologie τ in polynomieller Länge in τ beweisbar ist.

Also sollte es unerfüllbare Formeln geben, die nur extrem lange Resolutionsbeweise besitzen. Die Formalisierung des Schubfachprinzips wird ein Beispiel sein.

Das Schubfachprinzip

Verteilt man n Objekte in $n-1$ Fächer, erhält ein Fach mehr als ein Objekt.

Eine Formalisierung mit Hilfe der Aussagenlogik

(a) Wir arbeiten mit den Variablen $\mathbf{p_{i,j}}$ (für $1 \leqslant i \leqslant n$ und $1 \leqslant j \leqslant n-1$).

$\mathbf{p_{i,k}} = \mathbf{1}$ *soll bedeuten, dass Objekt i in Fach k gelegt wird.*

(b) Wir haben zwei Typen von Disjunktionstermen.

(1) Der Disjunktionsterm $\mathbf{D_i} = \mathbf{p_{i,1}} \vee \mathbf{p_{i,2}} \vee \cdots \vee \mathbf{p_{i,n-1}}$ ist genau dann erfüllt, wenn Objekt i in (mindestens) ein Fach gelegt wird.

(2) Der Disjunktionsterm $\mathbf{D_{i,j,k}} = \neg\mathbf{p_{i,k}} \vee \neg\mathbf{p_{j,k}}$ „fordert“ für $i \neq j$, dass Fach k nicht sowohl Objekt i wie auch Objekt j erhält.

(c) Das Schubfachprinzip entspricht der Konjunktion

$$\mathbf{S_n} := \Big(\bigwedge_{1 \leqslant i \leqslant n} D_i \Big) \wedge \Big(\bigwedge_{1 \leqslant i \neq j \leqslant n} \bigwedge_{1 \leqslant k \leqslant n-1} D_{i,j,k} \Big)$$

Resolutionsbeweise für die Unerfüllbarkeit von $\mathbf{S_n}$ besitzen die Länge $2^{\Omega(n)}$.

Das Beweisverfahren der Resolution: Der zeitliche Ablauf

1937 Blake entwickelt die Resolutionsregel.

1960 Der Davis-Putnam Algorithmus macht die Resolutionsregel populär.

1962 Zwei Jahre später wird der DPLL-Algorithmus vorgeschlagen.

1968 Tseitin zeigt eine exponentielle untere Schranke für die minimale Beweislänge eines stark eingeschränkten Resolutionsverfahrens.

1985 Haken beweist, dass jeder Resolutionsbeweis für das Schubfachprinzip S_n die Länge $2^{\Omega(n)}$ besitzen muss.

1996 Beame und Pitassi vereinfachen die Argumentation von Haken.

Der Beweis: Die wichtigen Konzepte

(a) Die Belegung **M** ist ein **(i-kritisches) Matching der Größe n − 1** $:\Longleftrightarrow$
- M verteilt die Objekte in $\{1, \ldots, n\} \setminus \{i\}$ bijektiv auf $n-1$ Fächer
- und verteilt Objekt i nicht, d.h. es gilt $M_{i,1} = \cdots = M_{i,n-1} = 0$.

(b) Wir verlangen nur, dass der am Ende aus den Disjunktionstermen von $\mathbf{S_n}$ bewiesene Disjunktionsterm **B**

für alle Matchings der Größe $n-1$ falsch ist.

Wir verlangen noch nicht einmal, dass $B = \epsilon$ gilt und sprechen deshalb von einem **schwachen Beweis** für $\mathbf{S_n}$.

Wir zeigen, dass selbst schwache Widerspruchsbeweise des Schubfachprinzips exponentiell lang sein müssen.

Schwache Beweise und die Negation

Für alle Matchings M der Größe $n-1$ gilt

$$M \models \neg p_{i,k} \iff M \models \bigvee_{j \neq i} p_{j,k}$$

für alle $i \in \{1, \ldots n\}$ und $k \in \{1, \ldots, n-1\}$.

Da wir nur an schwachen Beweisen interessiert sind:

Können wir alle Negationen $\neg p_{i,k}$ ungestraft durch $\bigvee_{j \neq i} p_{j,k}$ ersetzen?

- Natürlich nicht, denn dann ist die Resolutionsregel nicht mehr anwendbar.
- Aber übersetze einen konventionellen Resolutionsbeweis $(D_1, \ldots, D_m)$ in den **monotonen „Beweis“**

$$(D_1^+, \ldots, D_m^+)$$

wobei D_i^+ aus D_i durch Ersetzung aller Negationen entsteht. Es ist

$$M \models D_i \iff M \models D_i^+.$$

und $(D_1^+, \ldots, D_m^+)$ ist tatsächlich ein schwacher (monotoner) Beweis.

Gibt es Disjunktionsterme mit vielen Variablen?

? Besitzen schwache Widerspruchsbeweise $(D_1, \ldots, D_m)$ für S_n stets einen monotonen Disjunktionsterm D_i^+ mit vielen, (d.h. $\Omega(n^2)$) Variablen?

? Und wenn ja, gibt es kurze Widerspruchsbeweise mit dieser Eigenschaft?

Angenommen, *jeder* schwache Widerspruchsbeweis $(D_1, \ldots, D_m)$ für S_m hat einen monotonen Disjunktionsterm D_i^+ mit mindestens $\frac{2m^2}{9}$ Variablen.

Was passiert? (1/2)

Ein monotoner Disjunktionsterm D_i^+ mit mindestens $n^2/11$ Variablen heißt **lang**.

- Sei $\mathbf{L}$ die Anzahl der langen, montonen Disjunktionsterme.
  - Die durchschnittliche Häufigkeit einer Variablen in langen, monotonen Disjunktionstermen ist mindestens $\mathbf{L} \cdot \frac{\mathbf{n^2/11}}{\mathbf{n(n-1)}} \geqslant \mathbf{L/11}$.
    - ⋆ $\Longrightarrow$ es gibt eine Variable $\mathbf{p_{i,k}}$, die in mindestens $\mathbf{L/11}$ langen, monotonen Disjunktionstermen vorkommt.
    - ⋆ Wir setzen $\mathbf{p_{i,k} = 1} \Longrightarrow$ alle monotonen Disjunktionsterme, die $\mathbf{p_{i,k}}$ enthalten, sind wahr und können aus dem monotonen Beweis entfernt werden.
    - ⋆ Wie erhalten wir *nach Setzung* ein neues Schubfachproblem für $n-1$ Objekte?
  - Setze $\mathbf{p_{i,k'} = 0}$ für $k' \neq k$ sowie $\mathbf{p_{i',k} = 0}$ für $i' \neq i$.
    - ⋆ Die verbleibenden Disjunktionsterme des monotonen Widerspruchsbeweises müssen jetzt das Schubfach-Prinzip $\mathbf{S_{n-1}}$ für $n-1$ Objekte beweisen!
- Wiederhole diesen Setzungsprozess.

Was passiert? (2/2)

Unsere Annahme: Jeder Widerspruchsbeweis für S_m besitzt einen monotonen Disjunktionsterm mit $> \mathbf{2m^2/9}$ Variablen.

- Wiederhole den Setzungsprozess r mal: Wenn $r \leqslant n/3$, dann ist

$$\frac{\mathbf{2(n-r)^2}}{\mathbf{9}} \geqslant \frac{\mathbf{2(2n)^2}}{\mathbf{9^2}} = \frac{\mathbf{8n^2}}{\mathbf{81}} > \frac{\mathbf{n^2}}{\mathbf{11}}$$

  und das obige Argument ist für $r \leqslant n/3$ wiederholbar.
- Die ersten $n/3$ Anwendungen des Setzungsprozesses eliminieren jedesmal mindestens ein Elftel aller langen, monotonen Disjunktionsterme, und es folgt

$$\mathbf{L}\left(\frac{\mathbf{10}}{\mathbf{11}}\right)^{\mathbf{n/3}} \geqslant \mathbf{1}.$$

- Beachte $\left(\frac{\mathbf{11}}{\mathbf{10}}\right)^{\mathbf{n/3}} > \mathbf{2^{n/24}}$.

Es gibt stets monotone Disjunktionsterme mit $2n^2/9$ Variablen $\Longrightarrow$
Jeder schwache Resolutionsbeweis für $\mathbf{S_n}$ hat die Länge $\geqslant \mathbf{2^{n/24}}$.

Gibt es Disjunktionsterme mit $\geqslant 2n^2/9$ Variablen? (1/2)

Sei D ein (konventioneller) Disjunktionsterm eines schwachen Beweises von $\mathbf{S_n}$.

- Wir benutzen das „Fortschrittsmaß“

$$\mathbf{Zeugen(D)} = \{\mathbf{i} \mid \mathbf{D} \text{ wird von einem } i\text{-kritischen Matching falsifiziert } \}.$$

- Wie verhält sich das Fortschrittsmaß?
  - Für die Axiome
    - $D_i = p_{i,1} \vee p_{i,2} \vee \cdots \vee p_{i,m}$ ist $\mathbf{Zeugen(D_i)} = \{i\}$ und für
    - $D_{i,j,k} = \neg p_{i,k} \vee \neg p_{j,k}$ (mit $D^+_{i,j,k} = \varepsilon$) ist $\mathbf{Zeugen(D_{i,j,k})} = \emptyset$.
  - Für den letzten Disjunktionsterm $\mathbf{B}$ eines schwachen Beweises gilt

$$|\mathbf{Zeugen(B)}| = \mathbf{n}.$$

- Wenn Disjunktionsterm $\mathbf{D}$ aus $\mathbf{D'}$ und $\mathbf{D''}$ gefolgert wird, muss jede Belegung, die $\mathbf{D}$ falsifiziert, $\mathbf{D'}$ oder $\mathbf{D''}$ falsifizieren:

$$|\mathbf{Zeugen(D)}| \leqslant |\mathbf{Zeugen(D')}| + |\mathbf{Zeugen(D'')}|.$$

Es gibt einen Disjunktionsterm $\mathbf{D}$ mit $\mathbf{n/3 < t} := |\mathbf{Zeugen(D)}| < \mathbf{2n/3}$.

Gibt es Disjunktionsterme mit $\geqslant 2n^2/9$ Variablen? (2/2)

Es gibt einen Disjunktionsterm **D** mit $\mathbf{n/3 < t = |Zeugen(D)| < 2n/3}$.

- Angenommen, das **i-kritische** Matching $\mathbf{M^{(i)}}$ falsifiziert Disjunktionsterm **D**. Sei **j** ein beliebiges Objekt, das aber **nicht** zu **Zeugen(D)** gehört.
- Wir produzieren aus $\mathbf{M^{(i)}}$ ein **j-kritisches** Matching $\mathbf{M^{(j)}}$.
  - Wenn Objekt j für $\mathbf{M^{(i)}}$ in Fach l gelegt wird, dann lege für $\mathbf{M^{(j)}}$ stattdessen Objekt i in Fach l; Objekt j wird keinem Fach zugeteilt.
  - Da $\mathbf{M^{(j)}}$ **j-kritisch** ist, **j** aber nicht zu **Zeugen(D)** gehört, wird D durch $\mathbf{M^{(j)}}$ erfüllt, während D durch $\mathbf{M^{(i)}}$ falsifiziert wird.

$\mathbf{p_{i,l}}$ *kommt in* $\mathbf{D^+}$ *vor, denn* $\mathbf{D^+}$ *ist monoton.*

- Es sei $\mathbf{t = |Zeugen(D)|}$. Für jedes Objekt $\mathbf{i \in Zeugen(D)}$ besitzt $\mathbf{D^+}$ also mindestens $n - t$ Variablen $p_{i,l}$ $\Longrightarrow$

$(*)$ $\mathbf{D^+}$ hat mindestens $\mathbf{t(n-t)}$ Variablen.

$(*)$ Da $\mathbf{t > n/3}$, folgt $\mathbf{t(n-t) > 2n^2/9}$ und $\mathbf{D^+}$ hat mindestens $\mathbf{2n^2/9}$ Variablen.

Zusammenfassung

- Jeder schwache monotone Beweis besitzt einen langen Disjunktionsterm $\mathbf{D^+}$.
  - Wir haben das Fortschrittsmaß $\mathbf{Zeugen(D)}$ eingeführt, das „langsam“ ansteigt.
  - In jedem Widerspruchsbeweis von S_n gibt es einen Disjunktionsterm $\mathbf{D}$ mit

$$\mathbf{n/3 < t = |Zeugen(D)| < 2n/3}.$$

  - Jedes i-kritische Matching mit $i \in \mathbf{Zeugen}(D)$ produziert $n - t$ Variablen $\mathbf{p_{i,k}}$, die zu $\mathbf{D^+}$ gehören.
- Und die Konsequenzen?
  - Es gibt eine Variable X, die in einem konstanten Prozentsatz aller langen Disjunktionsterme auftritt. Setze diese Variable auf Eins $\Longrightarrow$
  - Die Anzahl langer Disjunktionsterme wird um mindestens einen konstanten Prozentsatz, nämlich um die in X erfüllten Disjunktionsterme, reduziert.
  - Der verbleibende Beweis muss aber das Schubfachprinzip für nur eine Dimension weniger herleiten: Der Setzungsprozess kann iteriert werden.

Temporale Aussagenlogik

Verifikation nebenläufiger Systeme

In einem *nebenläufigen* System laufen mehrere Berechnungen oder Prozesse

„fast gleichzeitig nebeneinander“.

(a) Meist sind mehrere Prozesse beteiligt, die mehr oder minder unabhängig voneinander sind, aber miteinander kommunizieren können.

(b) Die Schritte der einzelnen Prozesse können in beliebiger Weise miteinander verschränkt sein. Entsprechend komplex ist die Vermeidung von Fehlern.

Beispiel: Dienste eines Betriebssystems (wie etwa Drucken, Mausbewegungen, Surfen etc.) entsprechen Prozessen, die vielleicht auf einem einzigen Prozessor ausgeführt werden müssen: Ihre miteinander verschränkte Ausführung ist „auf unvorhersehbare Weise“ sequentiell.

Die Verifikation eines solchen Systems von Prozessen muss auf alle „Eventualitäten“ vorbereitet sein.

Was immer schief gehen kann, ...

(a) Die Explosion der Ariane 5, einer Trägerrakete der ESA vernichtete Werte in der Größenordnung von 500 Millionen Dollar.
   - Die Software hatte für die Ariane 4 perfekt funktioniert. Leider war die Geschwindigkeit der Ariane 5 größer $\Longrightarrow$ Overflow-Fehler.

(b) Der „Pentium Bug“ in 1994 kostete der Firma Intel nicht nur Ansehen, sondern auch mindestens 400 Millionen Dollar.

- „Normale Software“ hat bis zu 25 Fehlern auf tausend Programmzeilen,
- in guter Software reduziert sich die Fehlerzahl auf ca. zwei Fehler.
- In der Space Shuttle Software soll die Fehlerzahl bei zehntausend Zeilen sogar unter Eins liegen.

Model Checking: Was ist zu tun?

1. Man überführt den Entwurf des Systems in eine Beschreibung $\mathfrak{A}$, das **Model des Systems**.
   - Wir beschreiben $\mathfrak{A}$ mit Hilfe **reaktiver Systeme (Kripke-Strukturen)**: Ein reaktives System reagiert auf externe Einflüsse der Umwelt.

2. Das gewünschte Verhalten wird in einem Formalismus, häufig in einer Logik durch eine Formel ϕ beschrieben.
   - Wir betrachten **temporale Aussagenlogiken**, um den zeitlichen Ablauf in der Interaktion mit der Umwelt wiederzugeben.

3. Im letzten Schritt des Model Checking hat man nachzuweisen, dass das Modell $\mathfrak{A}$ die gewünschte Eigenschaft ϕ besitzt.
   - Wir brauchen eine temporale Aussagenlogik, die *ausdrucksstark* ist aber *beherrschbar* bleibt: Die **„Computational Tree Logic“ (CTL)**.

Kripke-Strukturen

Kripke-Strukturen

Eine **Kripke-Struktur** $\mathfrak{A} = (A, E, L)$ über einer endlichen Menge **Var** aussagenlogischer Variablen besitzt die folgenden Komponenten:

(a) Die endliche Menge A ist das **Universum** oder der **Zustandsraum**, die Elemente von A sind die **Zustände** von A.

(b) $E \subseteq A \times A$ ist eine Menge von gerichteten Kanten. Zusätzlich fordern wir, dass es für jeden Zustand a mindestens einen Nachfolgezustand gibt.

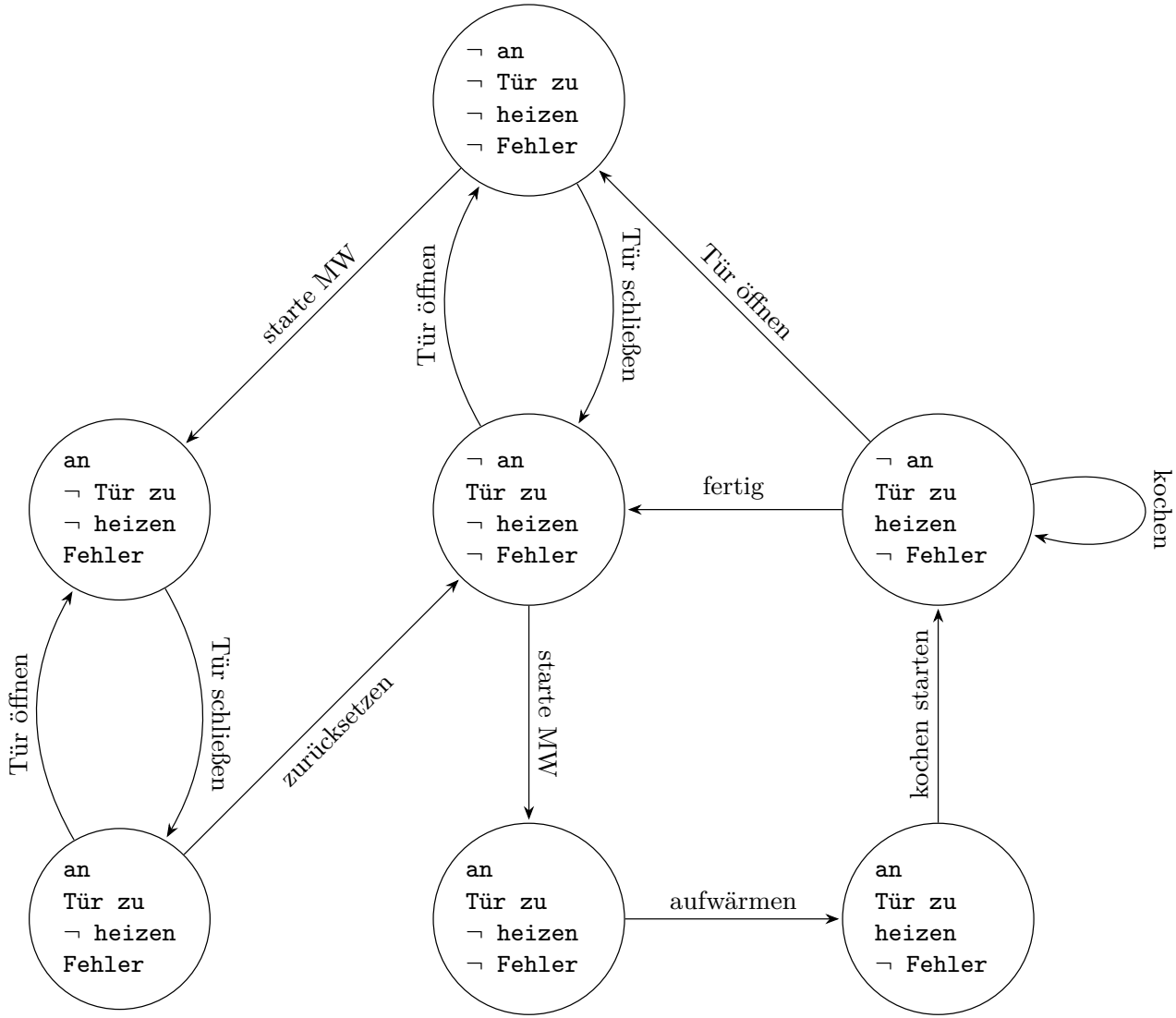
- Definiere $N(a) := \{b \in A : (a, b) \in E\}$ als die (nicht-leere) Menge der Nachfolgezustände.

(c) Die Funktion $L : A \to \mathcal{P}(\text{Var})$ weist jedem Zustand $a \in A$ die Menge $L(a)$ der in a erfüllten aussagenlogischen Variablen zu.

- $L(a)$ definiert für jeden Zustand a eine Belegung $\mathfrak{B}_a : \text{Var} \to \{0, 1\}$ mit $\mathfrak{B}_a(X) = 1 :\Longleftrightarrow X \in L(a)$.

Die Modellierung einer Mikrowelle

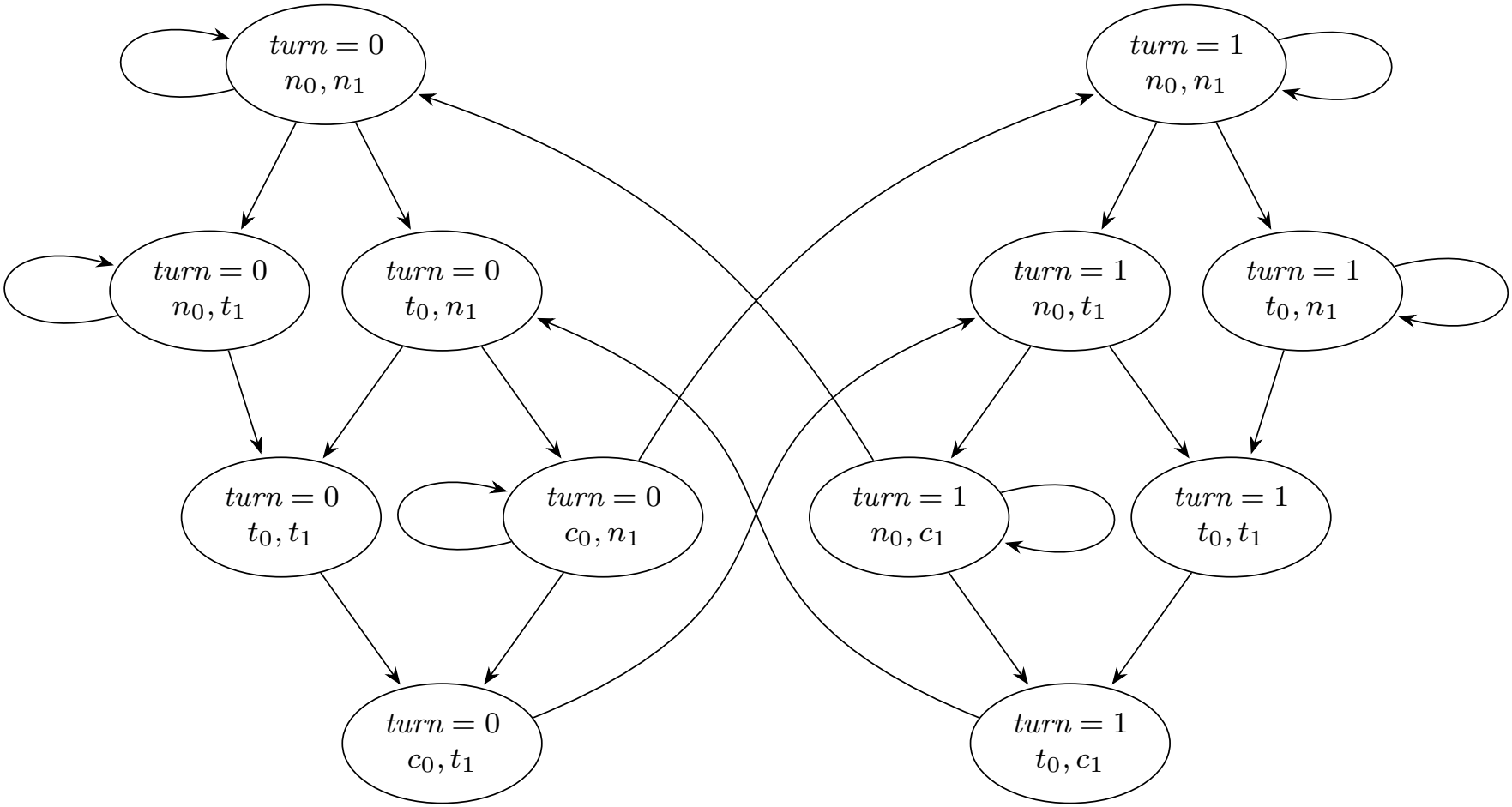
Das **Kripke-Diagramm** zu den Variablen „`an, Tür zu, heizen, Fehler`“.
(Die Aktion zu einem Zustandsübergang ist nicht Teil des Kripke-Systems.)



Wechselseitiger Ausschluss

Zwei Prozesse P_0 und P_1 treten auf: Prozess P_i befindet sich entweder im nicht-kritischen Bereich, bittet um Zugang oder hält sich im kritischen Bereich auf.

Zur Modellierung sei Var $:= \{\, \text{turn} = 0, \text{turn} = 1, c_0, c_1, n_0, n_1, t_0, t_1 \,\}$.



Computational Tree Logic (CTL)

Berechnungen

Sei $\mathfrak{A} = (A, E, L)$ eine Kripke-Struktur über der Variablenmenge Var.
Alle Berechnungen mögen im **Anfangszustand** $a \in A$ beginnen.

1. In einer Kripke-Struktur besitzt jeder Zustand mindestens einen Nachfolge-Zustand.
2. Eine **Berechnung** von $\mathfrak{A}$ entspricht deshalb einem in a beginnenden, unendlich langen Weg im Graphen (A, E).

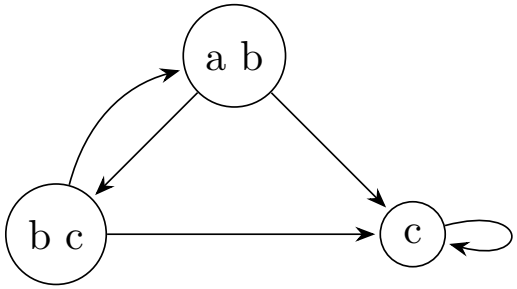
Unendlich lange, im Knoten a beginnende **Wege** des Graphen (A, E) und **Berechnungen** in der Kripke-Struktur sind äquivalente Begriffe.

.

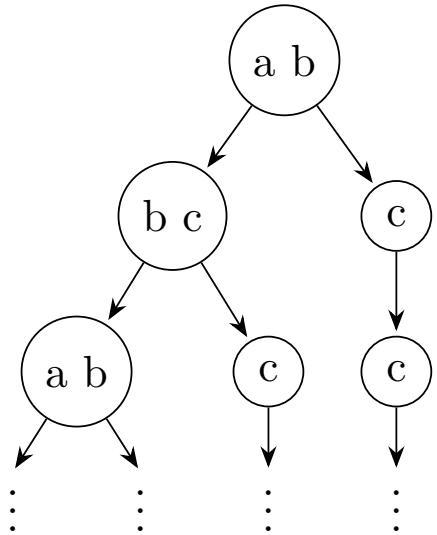
Berechnungsbaum

Der Berechnungsbaum (engl. computation tree) entsteht aus dem Graphen (A, E), indem man **alle** in a beginnenden Wege des Graphen aufführt.

Zum Beispiel, für den Graphen (mit Anfangszustand „a b“)



erhalten wir den Berechnungsbaum



Weg-Quantoren in CTL

In der Computational Tree Logic (CTL) werden Aussagen

(a) über die **Existenz** möglicher Berechnungen (d.h. unendlich langer, in *a* beginnender Wege) durch den **Weg-Quantor E**,

(b) bzw über **alle** möglichen Berechnungen durch den **Weg-Quantor A** ermöglicht.

Temporale Quantoren in CTL

Eine im Zustand a beginnende Berechnung werde ausgewählt.

(a) Eine aussagenlogische Formel ϕ gilt, wenn die von Zustand a definierte Belegung $\mathfrak{B}_a$ die Formel erfüllt, d.h. wenn $[\![\phi]\!]^{\mathfrak{B}_a} = 1$ gilt.

(b) **X**ϕ (bzw. next) soll bedeuten, dass ϕ im zweiten Knoten des Wegs gilt.
- ϕ gilt im **nächsten** Schritt der Berechnung.

(c) **F**ϕ (bzw. in the future) soll bedeuten, dass ϕ für irgendeinen Knoten des Wegs gilt.
- ϕ gilt **irgendwann** während der Berechnung.

(d) **G**ϕ (bzw. globally) soll bedeuten, dass ϕ für alle Knoten des Wegs gilt.
- ϕ gilt **für alle Schritte** der Berechnung.

(e) ϕ **U** ψ (bzw. ϕ until ψ) soll bedeuten, dass es einen Knoten des Weges gibt, in dem ψ gilt und ϕ für alle „vorigen“ Knoten erfüllt ist.
- ϕ gilt **solange bis** ψ erfüllt ist und ψ gilt irgendwann.

Die Syntax von CTL

Sei Var eine Menge aussagenlogischer Variablen. Die Formelmenge

CTL(Var)

wird rekursiv definiert.

Basisregel:

- **0** und **1** gehören zu CTL(Var).
- Wenn p eine aussagenlogische Variable in Var ist, dann gehört p zu CTL(Var), kurz: Var $\subseteq$ CTL(Var).

Rekursive Regeln:

- Wenn ϕ und ψ zu CTL(Var) gehören, dann gehören auch $\neg\phi, (\phi \wedge \psi), (\phi \vee \psi), (\phi \rightarrow \psi), (\phi \leftrightarrow \psi), (\phi \oplus \psi)$ zu CTL(Var).
  - ⋆ Formeln in CTL(Var) sind unter allen aussagenlogischen Junktoren abgeschlossen.
- Wenn ϕ und ψ zu CTL(Var) gehören, dann gehören auch **AX**ϕ, **EX**ϕ, **AF**ϕ, **EF**ϕ, **AG**ϕ, **EG**ϕ, sowie **A**(ϕ**U**ψ) und **E**(ϕ**U**ψ) zu CTL(Var).
  - ⋆ Auf jeden Weg-Quantor **muss** ein temporaler Quantor folgen.

Die Semantik von CTL: Junktoren.

Sei $\mathfrak{A} = (A, E, L)$ eine Kripke-Struktur. Wann gilt eine CTL(Var)-Formel ϕ in der Kripke-Struktur $\mathfrak{A}$ mit dem Anfangszustand a, d.h. wann gilt $(\mathfrak{A}, a) \models \phi$?

Basisregel:

- Es ist $(\mathfrak{A}, a) \models \mathbf{1}$ und $(\mathfrak{A}, a) \not\models \mathbf{0}$.
  - ⋆ Die Konstante $\mathbf{1}$ ist immer wahr, die Konstante $\mathbf{0}$ immer falsch.
- Für jedes $p \in \mathrm{Var}$ gilt $(\mathfrak{A}, a) \models p$ genau dann, wenn $p \in L(a)$, und ansonsten ist $(\mathfrak{A}, a) \not\models p$.
  - ⋆ Die Variable p ist genau dann aktuell erfüllt, wenn p im Zustand a erfüllt ist.

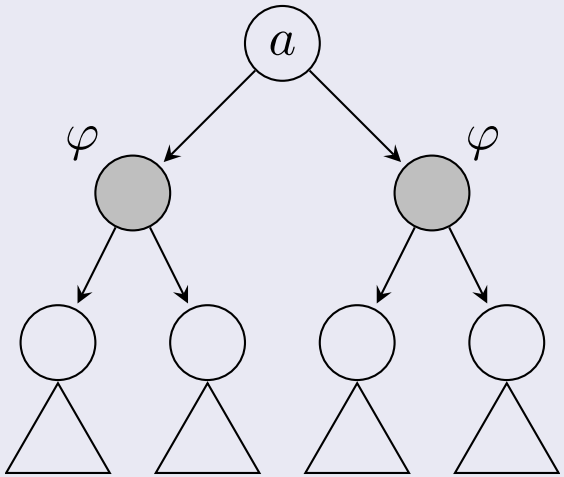
Rekursive Regeln, die Bedeutung der aussagenlogischen Junktoren:
Wenn ϕ und ψ zu CTL(Var) gehören, dann gilt

- $(\mathfrak{A}, a) \models \neg\phi$ genau dann, wenn $(\mathfrak{A}, a) \not\models \phi$.
- $(\mathfrak{A}, a) \models (\phi \wedge \psi)$ genau dann, wenn $(\mathfrak{A}, a) \models \phi$ und $(\mathfrak{A}, a) \models \psi$.
- $(\mathfrak{A}, a) \models (\phi \vee \psi)$ genau dann, wenn $(\mathfrak{A}, a) \models \phi$ oder $(\mathfrak{A}, a) \models \psi$.
- $(\mathfrak{A}, a) \models (\phi \rightarrow \psi)$ genau dann, wenn $(\mathfrak{A}, a) \not\models \phi$ oder $(\mathfrak{A}, a) \models \psi$.
- $(\mathfrak{A}, a) \models (\phi \leftrightarrow \psi)$ genau dann, wenn $(\mathfrak{A}, a) \models (\phi \wedge \psi)$ oder $(\mathfrak{A}, a) \models (\neg\phi \wedge \neg\psi)$).
- $(\mathfrak{A}, a) \models (\phi \oplus \psi)$ genau dann, wenn $(\mathfrak{A}, a) \not\models (\phi \leftrightarrow \psi)$.

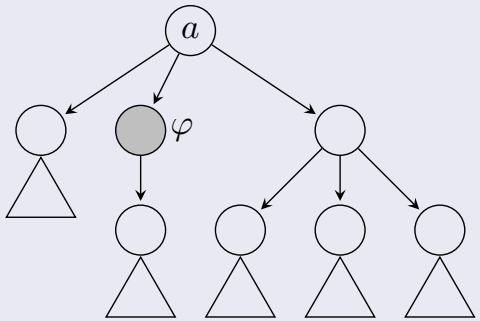
Die Semantik von CTL: Quantoren (1/4)

Wenn ϕ zu CTL(Var) gehört, dann gilt

- $(\mathfrak{A}, a) \models \mathbf{AX}\ \phi$ genau dann, wenn $(\mathfrak{A}, b) \models \phi$ für jeden direkten Nachfolger b von a gilt.



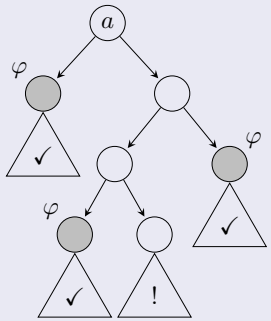
- $(\mathfrak{A}, a) \models \mathbf{EX}\ \phi$ genau dann, wenn $(\mathfrak{A}, b) \models \phi$ für mindestens einen direkten Nachfolger b von a gilt.



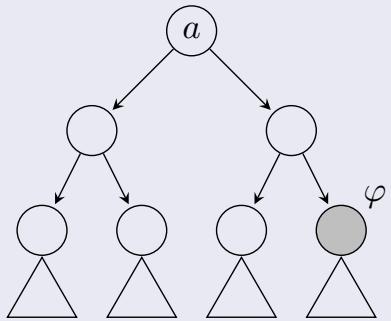
Die Semantik von CTL: Quantoren (2/4)

Wenn ϕ zu CTL(Var) gehört, dann gilt

- $(\mathfrak{A}, a) \models \mathbf{AF}\ \phi$ genau dann, wenn es auf jedem in a beginnenden unendlich langen Weg in (A, E) einen Knoten b des Weges mit $(\mathfrak{A}, b) \models \phi$ gibt.
  - ⋆ Auf jedem Weg gilt ϕ irgendwann.



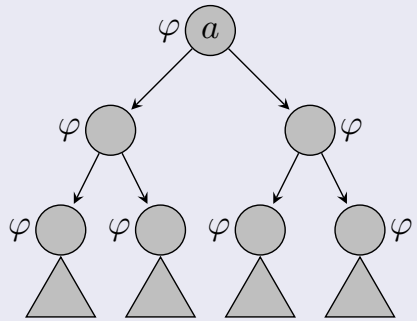
- $(\mathfrak{A}, a) \models \mathbf{EF}\ \phi$ genau dann, wenn für irgendeinen von a aus erreichbaren Knoten b gilt: $(\mathfrak{A}, b) \models \phi$. (Beachte, dass der Knoten a sich selbst erreicht).
  - ⋆ Es gibt einen Weg, auf dem ϕ irgendwann gilt.



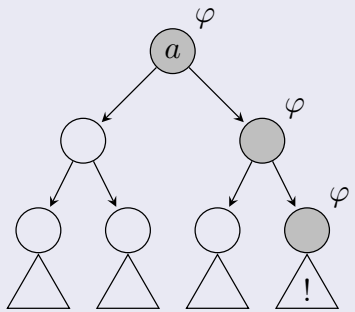
Die Semantik von CTL: Quantoren (3/4)

Wenn ϕ zu CTL(Var) gehört, dann gilt

- $(\mathfrak{A}, a) \models \mathbf{AG}\ \phi$ genau dann, wenn für jeden von a aus erreichbaren Knoten b gilt: $(\mathfrak{A}, b) \models \phi$.
  - ⋆ ϕ gilt immer.



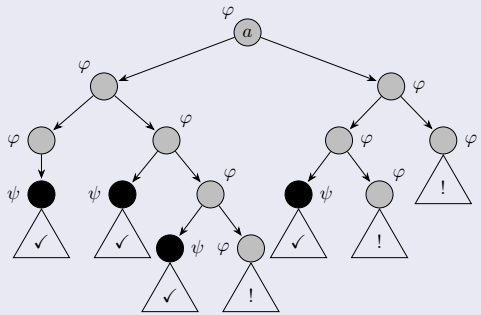
- $(\mathfrak{A}, a) \models \mathbf{EG}\ \phi$ genau dann, wenn es einen in a beginnenden unendlich langen Weg gibt, so dass für alle Knoten b des Wegs gilt: $(\mathfrak{A}, b) \models \phi$.
  - ⋆ Es gibt einen Weg, auf dem ϕ überall gilt.



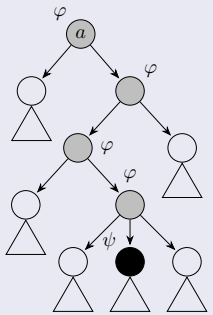
Die Semantik von CTL: Quantoren (4/4)

Wenn ϕ, ψ zu CTL(Var) gehören, dann gilt

- $(\mathfrak{A}, a) \models \mathbf{A}(\phi \mathbf{U} \psi)$ genau dann, wenn es auf jedem in a beginnenden unendlich langen Weg ein Anfangsstück $(b_0, b_1, \ldots, b_k)$ mit $a = b_0$ gibt, sodass $(\mathfrak{A}, b_k) \models \psi$ und $(\mathfrak{A}, b_i) \models \phi$ für alle $i < k$ gilt.
  - ⋆ Auf jedem Weg gilt ϕ solange, bis ψ gilt (und ψ gilt irgendwann).



- $(\mathfrak{A}, a) \models \mathbf{E}(\phi \mathbf{U} \psi)$ genau dann, wenn es einen Weg $(b_0, b_1, \ldots, b_k)$ mit $a = b_0$ gibt, sodass $(\mathfrak{A}, b_k) \models \psi$ und $(\mathfrak{A}, b_i) \models \phi$ für alle $i < k$ gilt.
  - ⋆ Auf mindestens einem Weg gilt ϕ solange, bis ψ gilt (und ψ gilt irgendwann).



Die Semantik von CTL: Beispiele (1/2)

1. Die aussagenlogische Variable „*Cola kommt raus*“ gehöre zu Var. Der Getränkeautomat rückt eine Cola irgendwann heraus, wenn die Formel

$$\mathbf{EF}\ \text{Cola kommt raus}$$

   gilt.
2. Für welche Zustände a gilt die CTL-Formel

$$\mathbf{AG}(\,\phi \to \mathbf{EF}\,\psi),$$

   d.h. wann gilt $(\mathfrak{A}, a) \models \mathbf{AG}(\,\phi \to \mathbf{EF}\,\psi)$?
   - Gilt irgendwann ϕ, dann gibt es eine Berechnung, in der ψ später gilt.
3. Für welche Zustände a gilt die CTL-Formel

$$\mathbf{AG}(\,\phi \to \mathbf{AF}\,\psi)$$

   - Für alle in a beginnenden Berechnungen: Ist ϕ wahr, dann gilt ψ später.
4. Für welche Zustände a gilt die Formel

$$(\mathbf{AG}\,(\mathbf{AF}\,\phi) \wedge \mathbf{AG}\,(\mathbf{AF}\,\neg\phi))$$

   - Für alle in a beginnenden Berechnungen und beliebige Zeitpunkte wird irgendwann ϕ wie auch $\neg\phi$ gelten.

Die Semantik von CTL: Beispiele (2/2)

5. Für welche Zustände a ist die Formel

$$\mathbf{AF}\,\mathbf{AG}\,\phi$$

wahr?

   - Für alle in a beginnenden Berechnungen gibt es einen (mgl. späteren) Zeitpunkt, ab dem ϕ immer gilt.
   - Zum Beispiel könnten wir mit $\phi :=$ (*Prozess 1 endet*) und der Formel $\mathbf{AF}\,\mathbf{AG}\,\phi$ fordern, dass Prozess 1 immer irgendwann endet.

6. Wir fordern für den Zustand a, dass solange ψ nicht zum ersten Mal eingetreten ist, auch ϕ nicht eintritt.
   - In allen in a beginnenden Berechnungen gilt also $\neg\phi$ solange ψ nicht gilt.
   - Wir fordern also die Formel $\mathbf{A}(\neg\phi\,\mathbf{U}\,\psi)$.

Wechselseitiger Ausschluss (1/3)

Formuliere wichtige Eigenschaften als CTL-Formeln.

1. **Sicherheit**: Zu keinem Zeitpunkt dürfen sich beide Prozesse im kritischen Bereich aufhalten.

$$\phi_1 := \mathbf{AG}\neg(c_0 \wedge c_1).$$

2. **Lebendigkeit**: Wenn ein Prozess um Erlaubnis um Zugang zum kritischen Bereich bittet, dann wird die Erlaubnis irgendwann gewährt.

$$\phi_2 := \mathbf{AG}((t_0 \rightarrow \mathbf{AF}c_0) \wedge (t_1 \rightarrow \mathbf{AF}c_1)).$$

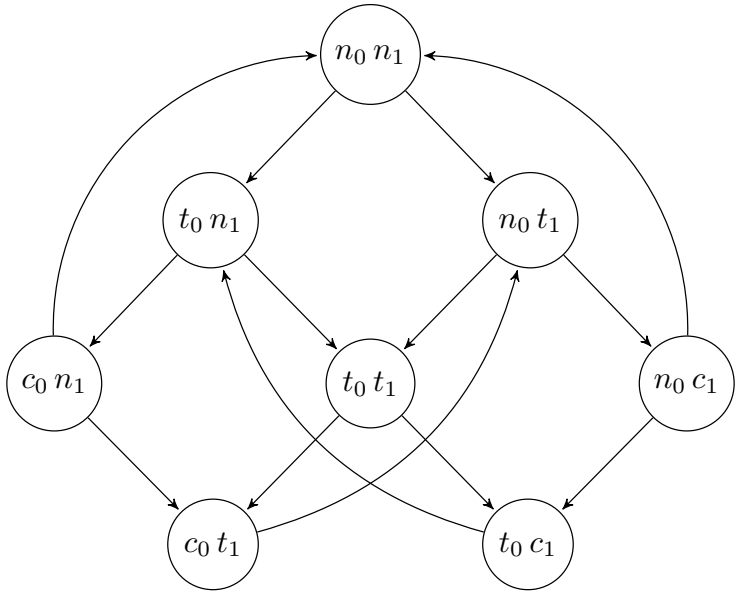
3. **Keine Blockade**: Jeder Prozess darf jederzeit um Zugang zum kritischen Bereich bitten.

$$\phi_3 := \mathbf{AG}(n_0 \rightarrow \mathbf{EX}\, t_0) \wedge \mathbf{AG}(n_1 \rightarrow \mathbf{EX}\, t_1).$$

Wechselseitiger Ausschluss (2/3)

Wir suchen eine Kripke-Struktur mit Sicherheit, Lebendigkeit und ohne Blockade.

Wir versuchen es mit einer neuen Kripke-Struktur $\mathfrak{A}_1$, diesmal über der Variablenmenge $\mathrm{Var} = \{ c_0, c_1, n_0, n_1, t_0, t_1 \}$. $n_0\, n_1$ ist der Anfangszustand a.

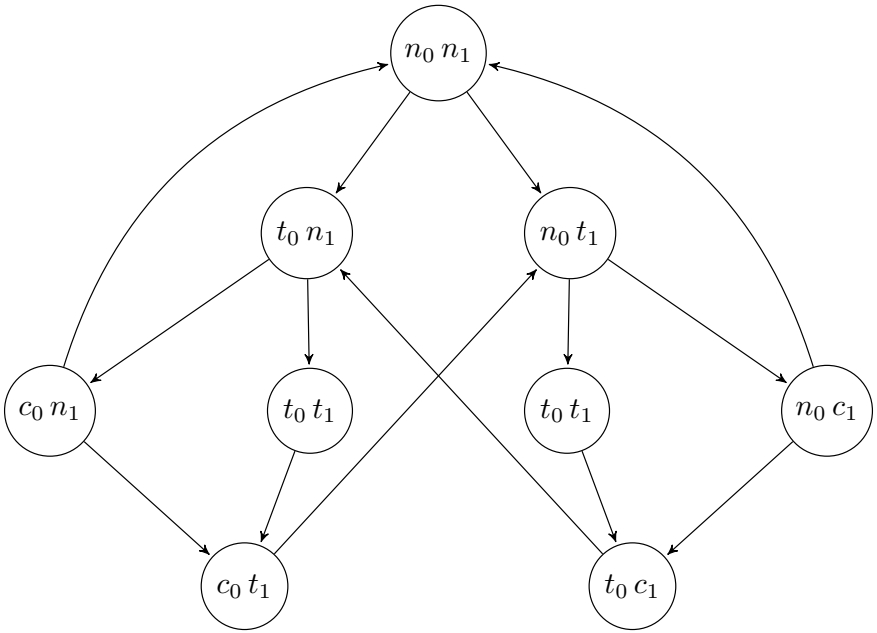


1. Es ist $(\mathfrak{A}_1, a) \models \phi_1 \wedge \phi_3 \Longrightarrow$ Sicherheit und Blockadefreiheit sind gewährleistet.
2. Gilt Lebendigkeit?
   - Wenn Prozess 0 als Erster um Zugang zum kritischen Bereich bittet ($t_0 = 1$, $n_1 = 1$), dann kann Prozess 1 immer noch an Prozess 0 „vorbeiziehen“.
   - Es kann nicht garantiert werden, dass Prozess 0 jemals zum Zug kommt.

Wechselseitiger Ausschluss (3/3)

Wir suchen eine Kripke-Struktur mit Sicherheit, Lebendigkeit und ohne Blockade.

Die Idee : Der Prozess, der als Erster nachfragt, erhält als Erster Zugang. Hier ist das Kripke-Diagramm der neuen Kripke-Struktur $\mathfrak{A}_2$:



$n_0\, n_1$ sei wieder Anfangszustand. Man überzeuge sich, dass Sicherheit, Lebendigkeit und Blockade-Freiheit garantiert sind, d.h. es ist

$$(\mathfrak{A}_2, a) \models \phi_1 \wedge \phi_2 \wedge \phi_3.$$

Äquivalenz

Äquivalente CTL-Formeln

Sei Var eine Menge aussagenlogischer Variablen und ϕ, ψ seien CTL(Var)-Formeln Dann gilt

$$\phi \equiv \psi$$

genau dann, wenn

$$(\mathfrak{A}, a) \models \phi \iff (\mathfrak{A}, a) \models \psi$$

für **alle** Kripke-Strukturen $\mathfrak{A} = (A, E, L)$ und **alle** Zustände $a \in A$ gilt.

Alle acht Quantorenpaare lassen sich mit **EX**, **EG** und **EU** ausdrücken, denn

(a) $\mathbf{AF}\,\phi \equiv \neg\mathbf{EG}\,\neg\phi$,
(b) $\mathbf{EF}\,\phi \equiv \mathbf{E}(\mathbf{1}\,\mathbf{U}\,\phi)$,
(c) $\mathbf{AG}\,\phi \equiv \neg\mathbf{EF}\,\neg\phi$,
(d) $\mathbf{A}(\phi\,\mathbf{U}\,\psi) \equiv \neg\mathbf{E}(\neg\psi\,\mathbf{U}\,(\neg\phi \wedge \neg\psi)) \wedge \neg\mathbf{EG}\neg\psi$,
(e) $\mathbf{AX}\,\phi \equiv \neg\mathbf{EX}\,\neg\phi$.

EX, **EG**, **EU** und **AU** genügen

Es gibt eine Konstante $K > 0$, so dass es zu jeder CTL-Formel ϕ eine äquivalente CTL-Formel ψ mit den folgenden Eigenschaften gibt:

(a) Die Anzahl der Symbole in ψ ist beschränkt durch das K-fache der Anzahl der Symbole in ϕ und

(b) nur die Quantorenpaare **EX**, **EG**, **EU** und **AU** kommen in ψ vor.

Fazit: Nach höchstens linearem Wachstum der Länge der Formel kann man sich auf vier Quantorenpaare beschränken.

Frage: Warum haben wir **AU** mit eingeschlossen?

Effizientes Model Checking

Model Checking: Die Fragestellung

Eine Kripke-Struktur $\mathfrak{A} = (A, E, L)$ über der Menge Var aussagenlogischer Variablen und eine CTL(Var)-Formel ϕ ist gegeben.

Bestimme die Menge

$$[\![\phi]\!]^{\mathfrak{A}} := \{ a \in A : (\mathfrak{A}, a) \models \phi \}$$

aller potentiellen Anfangszustände $a \in A$, in denen ϕ gilt.

Wir möchten zwar nur klären, ob $(\mathfrak{A}, a_0) \models \phi$ gilt, wir tun aber mehr.

- Und wenn ϕ keine Quantorenpaare besitzt, d.h. wenn ϕ eine aussagenlogische Formel ist?
  - Um nachzuprüfen, ob $(\mathfrak{A}, a) \models \phi$ gilt, müssen wir die aussagenlogische Formel ϕ nur für die von $L(a)$ definierte Belegung $\mathfrak{B}_a$ auswerten und das geht fix.
- Und wenn ϕ Quantorenpaare besitzt?
  - Jetzt wird's interessant!

Model Checking: Der rekursive Algorithmus (1/3)

Die Mengen $[\![\psi]\!]^{\mathfrak{A}}$, $[\![\psi_1]\!]^{\mathfrak{A}}$ und $[\![\psi_2]\!]^{\mathfrak{A}}$ seien bekannt.

(a) Wenn ϕ eine aussagenlogische Kombinationen von ψ, ψ_1 und ψ_2 ist:

- Wenn $\phi = \neg\psi$, dann ist $[\![\phi]\!]^{\mathfrak{A}} := A \setminus [\![\psi]\!]^{\mathfrak{A}}$.
- Wenn zum Beispiel $\phi = (\psi_1 \wedge \psi_2)$, dann ist $[\![\phi]\!]^{\mathfrak{A}} = [\![\psi_1]\!]^{\mathfrak{A}} \cap [\![\psi_2]\!]^{\mathfrak{A}}$.
- Die Junktoren $\vee, \rightarrow, \leftrightarrow$ und $\oplus$ lassen sich völlig analog behandeln.

(b) Nur die Quantorenpaare **EG**, **EU**, **EX** und **AU** kommen vor.

- Für $\phi =$ **EG** ψ stelle fest, ob ψ in mindestens einer Berechnung immer gilt.
  1. Berechne $[\![\psi]\!]^{\mathfrak{A}}$.

     Für welche Zustände $a \in [\![\psi]\!]^{\mathfrak{A}}$ gibt es einen in a beginnenden unendlich langen Weg, der nur in $[\![\psi]\!]^{\mathfrak{A}}$ verläuft? Ein solcher unendlich langer Weg wird von a aus zu einer starken Zusammenhangskomponente C laufen und dann in C „kreisen“.
  2. Berechne alle starken Zusammenhangskomponenten in $[\![\psi]\!]^{\mathfrak{A}}$.
  3. Es ist

$$[\![\phi]\!]^{\mathfrak{A}} = \{\, a \in [\![\psi]\!]^{\mathfrak{A}} \;:\; \text{es gibt einen in } [\![\psi]\!]^{\mathfrak{A}} \text{ verlaufenden Weg, der in } a \text{ beginnt und in einer starken Zusammenhangskomponente endet} \,\}.$$

Jetzt lässt sich $[\![\phi]\!]^{\mathfrak{A}}$ schnell mit Hilfe von Tiefensuche berechnen.

Model Checking: Der rekursive Algorithmus (2/3)

- Für $\phi = \mathbf{E}\,(\psi_1\,\mathbf{U}\,\psi_2)$ führe die folgenden Schritte aus:
  1. Berechne $[\![\psi_1]\!]^{\mathfrak{A}}$ und $[\![\psi_2]\!]^{\mathfrak{A}}$: Für welche Zustände a gibt es einen Weg, der in a beginnt, dann nur in $[\![\psi_1]\!]^{\mathfrak{A}}$ verläuft und in einem Knoten aus $[\![\psi_2]\!]^{\mathfrak{A}}$ endet?
  2. Bestimme die Menge B aller Zustände in $[\![\psi_1]\!]^{\mathfrak{A}}$, die einen Knoten in $[\![\psi_2]\!]^{\mathfrak{A}}$ mit einer Kante aus E erreichen, d.h.

$$B := \{\, b \in [\![\psi_1]\!]^{\mathfrak{A}} \;:\; \text{es gibt } c \in [\![\psi_2]\!]^{\mathfrak{A}} \text{ mit } (b,c) \in E \,\}.$$

  3. Dann ist

$$[\![\phi]\!]^{\mathfrak{A}} = \{\, a \in A \;:\; a \in [\![\psi_2]\!]^{\mathfrak{A}} \text{ oder es gibt einen Weg von } a \text{ zu einem Knoten in } B, \text{ der nur über Knoten in } [\![\psi_1]\!]^{\mathfrak{A}} \text{ verläuft} \,\}$$

  und auch diesmal lässt sich $[\![\phi]\!]^{\mathfrak{A}}$ schnell mit Tiefensuche berechnen.

- Für $\phi = \mathbf{EX}\psi$: Für welche Zustände $a \in A$ gilt ψ in einem Nachbarn von a?
  1. Bestimme $[\![\psi]\!]^{\mathfrak{A}}$.
  2. Dann ist

$$[\![\phi]\!]^{\mathfrak{A}} = \{\, a \in A \;:\; a \text{ besitzt einen direkten Nachfolger in } [\![\psi]\!]^{\mathfrak{A}} \,\}.$$

  Und wiederum lässt sich die Bestimmung schnell bewerkstelligen.

Model Checking: Der rekursive Algorithmus (3/3)

- Für $\phi = \mathbf{A}(\phi\ \mathbf{U}\ \psi)$ beachte die Äquivalenz

$$\mathbf{A}(\phi\,\mathbf{U}\,\psi) \equiv \neg\mathbf{E}(\neg\psi\,\mathbf{U}\,(\neg\phi \wedge \neg\psi)) \wedge \neg\mathbf{EG}\neg\psi.$$

Also ist

$$[\![\mathbf{A}(\phi\ \mathbf{U}\ \psi)]\!]^{\mathfrak{A}} = [\![\neg\mathbf{E}(\neg\psi\,\mathbf{U}\,(\neg\phi \wedge \neg\psi))]\!]^{\mathfrak{A}} \cap [\![\neg\mathbf{EG}\neg\psi]\!]^{\mathfrak{A}}.$$

Berechne die Zustandsmengen $[\![\neg\mathbf{E}(\neg\psi\,\mathbf{U}\,(\neg\phi \wedge \neg\psi))]\!]^{\mathfrak{A}}$ und $[\![\neg\mathbf{EG}\neg\psi]\!]^{\mathfrak{A}}$ mit den gerade beschriebenen Methoden.

Für eine Kripke-Struktur $\mathfrak{A} = (A, E, L)$ und eine CTL-Formel ϕ kann die Menge

$$[\![\phi]\!]^{\mathfrak{A}} := \{\, a \in A : (\mathfrak{A}, a) \models \phi\}$$

in Zeit $O(|E| \cdot |\phi|)$ bestimmt werden. ($|\phi|$ bezeichnet die Länge der Formel ϕ.)

(Un-)Vollständigkeit in der Prädikatenlogik: (Effiziente) Beweisbarkeit versus Wahrheit

Einige Grundbegriffe der Prädikatenlogik

(a) Eine **Signatur** σ ist eine Menge, die aus Symbolen für Konstanten, Funktionen und Relationen besteht.

(b) Eine σ-**Formel** besteht aus

- der aussagenlogischen Verknüpfung von Prädikaten aus σ, wobei Konstanten und Funktionen aus Σ in die Prädikate eingesetzt werden dürfen und
- aus Existenz- und Allquantoren.

(c) Eine σ-**Struktur** $\mathfrak{A} = (A, \sigma^{\mathfrak{A}})$ besteht aus einem Universum A und Interpretationen der Symbole aus σ.

- Sei ϕ eine Formel über σ. Dann schreiben wir

$$\mathfrak{A} \models \phi,$$

  wenn ϕ von $\mathfrak{A}$ erfüllt wird.
- Sei Φ eine Menge von Formeln über σ. Dann heißt eine σ-Struktur $\mathfrak{A}$ ein **Modell von** Φ, wenn $\mathfrak{A} \models \phi$ für jede Formel $\phi \in \Phi$ gilt.

Gibt es „vollständige“ Beweissysteme für die Prädikatenlogik mit denen man also „alles Wahre“ zeigen kann?

Der Hilbertkalkül

Der Hilbertkalkül $\mathfrak{H}$ (1/2)

Das Beweissystem des **Hilbertkalküls** benutzt die Schlussregel des Modus Ponens und die folgenden Axiome:

1. *Tautologien*: Eine Formel ϕ ist eine Tautologie, wenn ϕ aus einer aussagenlogischen Tautologie entsteht, indem die aussagenlogischen Variablen durch (beliebige) Formeln der Prädikatenlogik ersetzt werden.

   Alternativ genügt der Modus Ponens und die drei Axiomtypen des Beweissystems ABS, nämlich für alle Formeln ϕ, χ, ψ der Prädikatenlogik.

   (a) $\phi \to (\psi \to \phi)$,
   (b) $(\phi \to (\mu \to \psi)) \to ((\phi \to \mu) \to (\phi \to \psi))$ und
   (c) $(\neg\phi \to \neg\psi) \to (\psi \to \phi)$

2. *Gleichheitsaxiome*: $\forall x(x = x)$, $\forall x \forall y(x = y \to y = x)$ und $\forall x \forall y \forall z((x = y \wedge y = z) \to x = z)$.

   Weiterhin bleiben Funktionswerte unverändert und Relationen behalten ihren Wahrheitswert, wenn Argumente durch gleiche Argumente ersetzt werden.

3. *Spezialisierung*: Die Formel $\phi([t/x])$ entstehe aus der Formel ϕ, wenn die freie Variable x durch den Term t ersetzt wird. Dann ist

$$\forall x\phi \to \phi([t/x])$$

   für jede Formel ϕ, für jede freie Variable x und jeden Term t ein Axiom.

4. *Generalisierung*: Für jede Formel ϕ und jede Variable x, die nicht frei in ϕ vorkommt, ist

$$\phi \to \forall x\phi$$

   ein Axiom.

5. *Distributivität des Allquantors*: Für alle Formeln ϕ und ψ ist

$$\forall x(\phi \to \psi) \to (\forall x\phi \to \forall x\psi)$$

   ein Axiom.

Ableitbarkeit und Allgemeingültigkeit

Sei σ eine Signatur, ϕ eine σ-Formel und Φ eine Menge von σ-Formeln.

(a) ϕ ist aus Φ **ableitbar**,

$$\text{kurz: } \Phi \vdash_{\mathfrak{H}} \phi,$$

wenn ϕ aus Φ und den Axiomen des Hilbertkalküls durch Anwendungen der Schlussregel des Modus Ponens folgt.

(b) Wenn

$$\mathfrak{A} \models \Phi \Longrightarrow \mathfrak{A} \models \phi$$

für *alle* σ-Strukturen $\mathfrak{A}$ gilt, dann sagen wir, dass ϕ eine **semantische Folgerung** von Φ ist,

$$\text{kurz: } \Phi \models \phi.$$

Der Gödelsche Vollständigkeitssatz

Der Gödelsche Vollständigkeitssatz

Sei σ eine Signatur, ϕ eine σ-Formel und Φ eine Menge von σ-Formeln. Dann gilt

$$\Phi \models \phi \iff \Phi \vdash_{\mathfrak{H}} \phi.$$

Semantische Folgerung = Ableitbarkeit.

Der Gödelsche Vollständigkeitssatz verallgemeinert den Vollständigkeitssatz der Aussagenlogik auf die Prädikatenlogik.

Der Beweis des Goedelschen Vollständigkeitssatz: Die Idee

Zeige: Ist die Formelmenge Φ *konsistent*,

d.h. es gibt keine Formel ϕ mit $\Phi \vdash \phi$ und $\Phi \vdash \neg\phi$,

dann gibt es eine Struktur $\mathfrak{A}$ mit $\mathfrak{A} \models \Phi$.

1. Zeige den Satz von Lindenbaum:
   Ist Φ konsistent, dann gibt es eine *maximal konsistente* Obermenge $\Phi \subseteq \Phi^*$

   Die Hinzunahme einer beliebigen neuen Formel zu Φ^* verletzt die Konsistenz von Φ^*.

2. Für jede Formel ϕ füge die „Henkin-Konstante“ c_ϕ zu der Menge aller Konstanten hinzu und füge die Formel $\exists x \phi(x) \rightarrow \phi(c_\phi)$ zur Menge Φ^* hinzu.

   Quantoren-Elimination.

3. Baue aus allen Termen – den Henkin-Konstenten, den Funktionssymbolen

   die sogenannte Term-Interpretation.

   mit allen möglichen Henkin-Konstanten eingesetzt etc. – ein Model für Φ^* .

Vollständige Theorien

Die Vollständigkeit logischer Theorien

Sei σ eine Signatur. Weiterhin sei Φ eine Formelmenge der Prädikatenlogik und ϕ eine Formel der Prädikatenlogik, jeweils über der Signatur σ.

(a) Φ heißt eine **logische Theorie**, wenn jede aus Φ ableitbare Formel in Φ enthalten ist.

(b) Eine logische Theorie Φ ist **vollständig**, wenn für jede Formel ϕ über σ gilt

$$\phi \in \Phi \text{ oder } \neg\phi \in \Phi.$$

Für jede σ-Struktur $\mathfrak{A}$ ist

$$\mathrm{Th}(\mathfrak{A}) := \{\, \phi \,:\, \phi \text{ ist eine } \sigma\text{-Formel und } \mathfrak{A} \models \phi \,\}$$

eine vollständige Theorie.

Die Theorie der reellen Zahlen

Die Theorie der reellen Zahlen (1/3)

Die Signatur $\sigma := \{0, 1, +, \cdot, <\}$ sei gegeben. Dann ist

$$\mathfrak{R} = (\mathbb{R}, 0, 1, +, \cdot, <)$$

die Struktur der reellen Zahlen mit Addition, Multiplikation und vollständiger Ordnung.

$$\mathrm{Th}(\mathfrak{R})$$

ist die **Theorie der reellen Zahlen**, besteht also aus allen σ-Formeln, die in $\mathfrak{R}$ wahr sind.

(a) Besitzt $\mathrm{Th}(\mathfrak{R})$ ein „übersichtliches“ Axiomensystem?

(b) Wie schwierig ist die Auffinden von Beweisen?

Die Theorie der reellen Zahlen (2/3)

Das Axiomensystem für $\mathrm{Th}(\mathbb{R})$ besteht aus

1. den Körperaxiomen:
   - Addition und Multiplikation sind assoziativ und kommutativ und das Distributivgesetz gilt.
   - 0 und 1 sind neutrale Elemente für Addition und Multiplikation.
   - Alle Zahlen (bis auf die Null) besitzen additive und multiplikative Inverse.
2. den Axiomen für eine vollständige Ordnung:
   - $\neg(x < x)$.
   - $((x < y) \wedge (y < z)) \rightarrow (x < z)$.
   - $(x <) \vee (x = y) \vee (y < x)$.
   - $(x < y) \rightarrow (x + z < y + z)$.
   - $((0 < x) \wedge (0 < y)) \rightarrow (0 < x \cdot y)$.
3. und den Axiomen für einen reellen Abschluss:
   - $(0 < x) \rightarrow \exists y(y^2 = x)$
   - und jedes Polynom von ungeradem Grad hat eine reellwertige Wurzel.

Man kann beweisen, dass exponentielle Zeit (in der Länge der Formel) notwendig ist

Die Theorie der reellen Zahlen (3/3)

Aber in $\mathbb{R}$ kann man Grenzwerte definieren und untersuchen.
Das funktioniert mit unserer Signatur nicht!

4. Das Vollständigkeitaxiom: Jede nicht-leere nach oben beschränkte **Menge** reeller Zahlen besitzt ein Supremum, also eine kleinste obere Schranke.

Die reellen Zahlen bilden die **einzige** Struktur, die alle Axiome in 1. - 4. erfüllen!

Das Vollständigkeitsaxiom kann mit unserer Signatur nicht gezeigt werden.

Arbeite entweder mit der Zermelo-Fraenkel Mengenlehre oder

arbeite in der Logik der *zweiten* Stufe:

:-)) Jetzt dürfen Quantoren über Mengen benutzt werden,

:-((aber es gibt keinen Vollständigkeitssatz!

Zwischenfrage:
Reicht die Zermelo-Fraenkel Mengenlehre?

ZF: Extensionalität und die Nullmenge

Die Formeln der Mengenlehre benutzen die Signatur

$$\sigma := \{\dot{\in}\}$$

mit dem 2-stelligen Relationssymbol $\dot{\in}$. Die **Zermelo-Fraenkel Mengenlehre** besteht aus der Menge ZF aller Axiome:

1. Das `Extensionalitätsaxiom`: Zwei Mengen A, B sind genau gleich, wenn sie dieselben Elemente besitzen:

$$A \doteq B \leftrightarrow \forall x \big(x \dot{\in} A \leftrightarrow x \dot{\in} B\big).$$

2. Das `Nullmengenaxiom` fordert, dass die leere Menge eine Menge ist:

$$\exists A \forall x \, \neg \big(x \dot{\in} A\big)$$

   Als Folgerung des Extensionalitätsaxioms und des Nullmengenaxioms gibt es genau eine leere Menge, die wir natürlich mit $\emptyset$ bezeichnen.

ZF: Das Paarmengenaxiom ...

3. Das `Paarmengenaxiom` fordert, dass für alle Mengen A, B auch $\{A, B\}$ eine Menge ist:
$$\forall A \forall B \exists C \forall x \Big(x \dot{\in} C \leftrightarrow (x \dot{=} A \vee x \dot{=} B) \Big).$$
4. Das `Vereinigungsaxiom` besagt, dass mit jeder Menge A von Mengen auch die Vereinigung $\bigcup_{a \in A} a$ aller Elemente von A eine Menge ist:
$$\forall A \exists B \forall x \Big(x \dot{\in} B \leftrightarrow \exists y (y \dot{\in} A \wedge x \dot{\in} y) \Big).$$
Paarmengenaxiom und Vereinigungsaxiom zusammen garantieren, dass auch die Vereinigung $A_1 \cup A_2$ von zwei Mengen A_1, A_2 eine Menge ist. Dazu bilden wir zuerst die Paarmenge $\{A_1, A_2\}$ und beachten $A_1 \cup A_2 = \bigcup_{a \in \{A_1, A_2\}} a$.
5. Das `Potenzmengenaxiom` fordert, dass die Menge aller Teilmengen einer Menge A eine Menge ist:
$$\forall A \exists B \forall x \Big(x \dot{\in} B \leftrightarrow \forall y (y \dot{\in} x \rightarrow y \dot{\in} A) \Big).$$

ZF: Das Aussonderungs- und Unendlichkeitsaxiom

6. Das `Aussonderungsaxiom`: Für jede Menge A und jede Formel α der Mengenlehre ist $B = \{C \in A \ : \ \alpha(C)\}$ eine Menge. D.h.

$$\forall A \, \exists B \, \forall C \, \big(\ C \dot{\in} B \leftrightarrow C \dot{\in} A \wedge \alpha(C)\big).$$

7. Das `Unendlichkeitsaxiom`: Es gibt eine Menge A, die die leere Menge und mit jedem Element x auch die Menge $x \cup \{x\}$ enthält.

$$\exists A \, \big(\ \emptyset \dot{\in} A \ \wedge \ \forall X \ (\ X \dot{\in} A \rightarrow X \cup \{X\} \dot{\in} A)\big)$$

Der Schnitt all dieser Mengen A ist die kleinste solche unendliche Menge, nämlich die Menge der natürlichen Zahlen:

$$\mathbb{N} := \{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}, \{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}\}, \ldots\}.$$

Die Bildung der Schnittmenge erfolgt mit Hilfe des Aussonderungsaxioms.

Wir erhalten die Induktionsaxiome.
Aber wie kann man über Addition und Multiplikation sprechen?

ZF: Das Fundierungs- und Ersetztungsaxiom

8. `Fundierungsaxiom`: Jede nichtleere Menge A enthält ein Element B mit $A \cap B = \emptyset$

$$\forall A \left(\neg(A = \emptyset) \rightarrow \exists B \left(B \dot{\in} A \ \wedge \ \neg \exists C (C \dot{\in} A \wedge C \dot{\in} B) \right) \right).$$

$\Longrightarrow$ Eine Folge $x_1 \ni x_2 \ni x_3 \ni \ldots$ ist nicht möglich, denn $A = \{x_1, x_2, x_3, \ldots\}$ widerspricht dem Axiom: Für jedes Element $x_i \in A$ ist $x_{i+1} \in x_i \cap A$
$\Longrightarrow$ Eine Menge kann sich nicht selbst als Element enthalten.

9. Das `Ersetzungsaxiom`: Wird jedes Element einer Menge 1-deutig durch eine Menge ersetzt, so erhält man eine Menge. D.h. für jede Formel $\alpha(X, Y)$ gilt:

$$\forall X, Y, Z \left(\alpha(X, Y) \ \wedge \alpha(X, Z) \rightarrow Y = Z \right) \rightarrow$$
$$\forall A \, \exists B \, \forall C \left(C \dot{\in} B \leftrightarrow \exists D \left(D \dot{\in} A \ \wedge \ \alpha(D, C) \right) \right).$$

Beachte $B = \{ Y \ : \ D \in A \ \wedge \ \alpha(D, Y) \}$.

Zahlentheorie

Die Presburger-Arithmetik (1/2)

Die Signatur $\sigma_{\{+\}} = \{0, S, +, <\}$ sei gegeben. Dann ist

$$\mathfrak{N}_{\{+\}} = (\mathbb{N}, 0, \text{Nachfolger}, +, <)$$

die Struktur der natürlichen Zahlen mit Nachfolger$(n) = n + 1$, Addition und Ordnung.

$$\text{Th}(\mathfrak{N}_{\{+\}})$$

ist die **additive Theorie der natürlichen Zahlen**, besteht also aus allen $\sigma_{\{+\}}$-Formeln, die in $\mathfrak{N}_{\{+\}}$ wahr sind.

(a) Besitzt $\text{Th}(\mathfrak{N}_{\{+\}})$ ein „übersichtliches“ Axiomensystem?

(b) Wie schwierig ist die Auffinden von Beweisen?

Die Presburger-Arithmetik (2/2)

(a) Die Signatur $\sigma_{\{+\}} = \{0, S, +, <\}$ mit dem „Successor-Symbol“ S sei gegeben.

(b) Die **Axiome** der Presburger-Arithmetik lauten:

$$\begin{array}{llll} (P1) & S(x) \neq 0, & (P2) & S(x) = S(y) \to x = y, \\ (P3) & x + 0 = x, & (P4) & x + S(y) = S(x + y), \\ (P5) & \neg(x < 0), & (P6) & x < S(y) \leftrightarrow x < y \vee x = y, \\ & & (P7) & x < y \vee x = y \vee y < x, \end{array}$$

Weiterhin wird für jede $\sigma_{\{+\}}$-Formel ϕ das **Induktionsaxiom**

$$\phi(0) \wedge \forall x(\phi(x) \to \phi(S(x))) \to \phi$$

gefordert.

(a) PR-A ist die Menge aller Formeln der Prädikatenlogik, die aus den Axiomen der Presburger-Arithmetik ableitbar sind.

(b) Ein **Vollständigkeitssatz** (Ableitbarkeit „=“ Wahrheit) gilt:

$$\mathfrak{N}_{\{+\}} \models \phi \iff \phi \in \text{PR-A}.$$

Die Peano-Arithmetik (1/2)

Die Signatur $\sigma_{\{+,*\}} = \{0, S, +, *\}$ sei gegeben. Dann ist

$$\mathfrak{N}_{\{+,*\}} = (\mathbb{N}, 0, \text{Nachfolger}, +, *)$$

die Struktur der natürlichen Zahlen mit Nachfolger$(n) = n + 1$, Addition und Multiplikation.

$$\text{Th}(\mathfrak{N}_{\{+,*\}})$$

ist die **Theorie der natürlichen Zahlen**, besteht also aus allen $\sigma_{\{+,*\}}$-Formeln, die in $\mathfrak{N}_{\{+,*\}}$ wahr sind.

(a) Hat die „Zahlentheorie“ TH$(\mathfrak{N}_{\{+,*\}})$ ein „übersichtliches“ Axiomensystem?

(b) Was heißt übersichtlich?

Die Peano-Arithmetik (2/2)

(a) Die Signatur $\sigma_{\{+,*\}} = \{0, S, +, *, <\}$ sei gegeben.

(b) Die **Axiome** der Presburger-Arithmetik lauten:

$$\begin{array}{llll} (P1) & S(x) \neq 0, & (P2) & S(x) = S(y) \rightarrow x = y, \\ (P3) & x + 0 = x, & (P4) & x + S(y) = S(x + y), \\ (P5) & x * 0 = 0, & (P6) & x * S(y) = x * y + x, \\ (P7) & \neg(x < 0), & (P8) & x < S(y) \leftrightarrow x < y \vee x = y, \\ & & (P9) & x < y \vee x = y \vee y < x, \end{array}$$

Weiterhin wird für jede $\sigma_{\{+,*\}}$-Formel ϕ das **Induktionsaxiom**

$$\phi(0) \wedge \forall x(\phi(x) \rightarrow \phi(S(x))) \rightarrow \phi$$

gefordert.

(a) PE-A ist die Menge aller Formeln der Prädikatenlogik, die aus den Axiomen der Peano-Arithmetik ableitbar sind.

(b) Gilt ein **Vollständigkeitssatz**?

Der Gödelsche Unvollständigkeitssatz

Der Gödelsche Unvollständigkeitssatz

Für jedes vernünftige(!?) Axiomensystem der Zahlentheorie gibt es ein „nicht-standard Modell“ mit „nicht-standard Eigenschaften“.

Nicht alle wahren Aussagen der Zahlentheorie sind auch ableitbar:

- Es ist nicht ausgeschlossen, dass es in einigen Modellen der Peano-Arithmetik unendlich viele Primzahlzwillinge gibt, in anderen hingegen nicht.
- In einem solchen Fall genügen die Axiomen der Peano-Arithmetik nicht, um die Frage nach unendlich vielen Primzahlzwillingen zu beantworten.

Aber welche Axiome sollte man hinzufügen? Diese „neuen“ Axiome werden nicht unmittelbar einsichtig sein: Wer „garantiert“, dass sie in der Zahlentheorie gelten?

Das Hilbertsche Programm

- In 1903 zeigt Bertrand Russel, dass die naive Mengenlehre nicht widerspruchsfrei ist.
- Als Reaktion darauf entwickelt David Hilbert in den 20'er Jahren das „Hilbertsche Programm“:
  - Finde ein Axiomensystem mit unmittelbar einsichtigen Axiomen, um die Aussagen der Mathematik zweifelsfrei nachzuweisen.
  - Das Axiomensystem und seine Schlussregeln müssen mächtig genug sein, um alle **wahren** Aussagen **abzuleiten** zu können.
- In 1929 zeigt Kurt Goedel den Vollständigkeitssatz und unterstützt damit das Hilbertsche Programm.

Mit seinem Unvollständigkeitssatz in 1932, zeigt Goedel, dass das Hilbertsche Programm nicht wie ursprünglich gedacht durchgeführt werden kann.

Der Goedelsche Unvollständigkeitssatz: Die Idee (1/3)

Man kann zeigen: Für *jede rekursiv aufzählbare Sprache* L über dem Alphabet $\{0,1\}$ gibt es eine Formel φ_L der Peano-Arithmetik mit

$$x \in L \Longleftrightarrow \varphi_L(x) \text{ ist wahr}$$

- Wenn Wahrheit und Ableitbarkeit übereinstimmen, dann folgt also

$$\begin{aligned} x \in L &\Longleftrightarrow \varphi_L(x) \text{ ist wahr} \\ &\Longleftrightarrow \varphi_L(x) \text{ ist in der Peano Arithmetik beweisbar.} \end{aligned}$$

  bzw. $L = \{x | (\varphi_L(x)) \text{ ist in der Peano Arithmetik beweisbar } \}$.
- und ebenso natürlich

$$\begin{aligned} x \notin L &\Longleftrightarrow \neg\varphi_L(x) \text{ ist wahr} \\ &\Longleftrightarrow (\neg\varphi_L(x)) \text{ ist beweisbar.} \end{aligned}$$

- Also ist $\bar{L} = \{x | (\neg\varphi_L(x)) \text{ ist beweisbar } \}$.

Der Goedelsche Unvollständigkeitssatz: Die Idee (2/3)

- Aber $\bar{L} = \{x | (\neg\varphi_L(x)) \text{ ist beweisbar } \}$ ist dann auch rekursiv aufzählbar.
  - Um zu prüfen, ob eine Formel φ ableitbar ist, zähle alle möglichen Beweise auf und akzeptiere, wenn ein Beweis für φ gefunden wird.
- L war aber eine beliebige rekursiv aufzählbare Sprache und somit ist jede rekursiv aufzählbare Sprache auch entscheidbar.
  - Wenn L rekursiv aufzählbar und $\bar{L}$ rekursiv aufzählbar, dann ist L entscheidbar.

Blanker Unsinn $\Longrightarrow$ Beweisbarkeit ist schwächer als Wahrheit!

Und wenn wir das Axiomensystem um weitere wahre Aussagen erweitern?

Der Goedelsche Unvollständigkeitssatz: Die Idee (3/3)

Wir fügen weitere – möglicherweise sogar unendlich viele – wahre Aussagen als Axiome hinzu und erhalten ein neues Axiomensystem P.

- Wenn P nicht rekursiv aufzählbar ist, dann können wir noch nicht einmal verifizieren, dass ein Axiom zu P gehört.
- Wenn das Axiomensystem P rekursiv aufzählbar ist **und** Vollständigkeit gilt, dann ist $\overline{L}$ für jede rekursiv aufzählbare Menge L rekursiv aufzählbar!

Sei P eine rekursiv aufzählbare Menge von wahren Formeln der Zahlentheorie.

Dann gibt es eine wahre Formel der Zahlentheorie, die nicht aus P ableitbar ist.

Und die Konsequenz für die Informatik/Philosophie?

- ✓ Ein Vollständigkeitsatz für die Prädikatenlogik gilt!

  Wahrheit = Beweisbarkeit.

- ↯ Einige komplexe Realitäten

  wie etwa Addition und Multiplikation in den natürlichen Zahlen,

  lassen sich nicht beherrschen!

Auswahlaxiom und Kontinuumshypothese

Die beiden folgenden Aussagen können in der Zermelo-Fraenkel Mengenlehre ZF formalisiert werden, sind aber **nicht** Folgerungen von ZF.

(a) Das **Auswahlaxiom**:

Für alle Mengen A, deren Elemente paarweise disjunkte Mengen sind, gibt es eine Menge B, die genau ein Element aus jedem Element von A enthält.

(b) Die **Kontinuumshypothese**:

Jede überabzählbare Teilmenge der reellen Zahlen ist gleichmächtig mit der Menge der reellen Zahlen.

Gilt das Auswahlaxiom? Ist die Kontinuumshypothese richtig?
Es gibt absolute Erkenntnis-Grenzen!

Die Komplexität der Presburger-Arithmetik

Zur Erinnerung: Die Presburger-Arithmetik

(a) Benutze die Konstante 0, die Prädikate $x = y$, $x + y = z$ und $x < y$ sowie das Funktionssymbol S (für Successor).

(b) Die **Axiome** der Presburger-Axiome lauten:

$$\begin{array}{llll} (P1) & S(x) \neq 0, & (P2) & S(x) = S(y) \to x = y, \\ (P3) & x + 0 = x, & (P4) & x + S(y) = S(x + y), \\ (P5) & \neg(x < 0), & (P6) & x < S(y) \leftrightarrow x < y \vee x = y, \\ & & (P7) & x < y \vee x = y \vee y < x, \end{array}$$

Weiterhin fordere für jede Formel ϕ das **Induktionsaxiom**

$$\phi(0) \wedge \forall x(\phi(x) \to \phi(S(x))) \to \phi.$$

PR-A ist die Menge aller Formeln der Prädikatenlogik, die aus den Axiomen der Presburger-Arithmetik ableitbar sind.

Es wird gruselig: Die Komplexitätsklasse E

(a) Für eine Funktion $t : \mathbb{N} \to \mathbb{N}$ besteht die Komplexitätsklasse

$$\mathrm{DTIME}(t)$$

aus allen Sprachen, die in Zeit höchstens $O(t(n))$ für Eingaben der Länge n berechnet werden können.

(b) Definiere

$$\mathrm{E} := \bigcup_{k \in \mathbb{N}} \mathrm{DTIME}(2^{kn})$$

als die Komplexitätsklasse aller in Exponentialzeit berechenbaren Sprachen.

E-harte Sprachen werden mit Hilfe der polynomiellen Reduktion definiert:

Eine Sprache L heißt **hart für** E , wenn $K \leqslant_P L$ für jede Sprache $K \in \mathrm{E}$ gilt.

Hierarchie-Satz für Laufzeit: Wenn die Sprache L hart für E ist, dann folgt $L \notin \mathrm{P}$

Die Presburger-Arithmetik ist sehr schwierig

Die Presburger-Arithmetik PR-A ist hart für E. Die wahren Formeln von PR-A können also nicht in polynomieller Zeit erkannt werden.

(a) PR-A ist sogar hart für $\mathrm{NTIME}(2^{2^{q \cdot n}})$ für eine positive rationale Zahl q. (Auch diesmal wird Härte durch die polynomielle Reduktion definiert.)

(b) Als „positives“ Ergebnis ist nur die Inklusion

$$\text{PR-A} \in \mathrm{DTIME}(2^{2^{2^{c \cdot n}}})$$

für eine positive Konstante c bekannt.

Die Komplexität der Presburger-Arithmetik ist also „irgendwo“ zwischen doppelt und dreifach exponentieller Laufzeit anzusiedeln.

Die Beweisidee

Es gibt eine Formel $\phi_{M,w}$ der **Peano-Arithmetik**, so dass $\phi_{M,w}$ genau dann wahr ist, wenn M die Eingabe w akzeptiert.

1. Um Härte für E zu zeigen, betrachte eine deterministische Turingmaschine M, die stets nach $T = 2^{\alpha \cdot n}$ Schritten hält.
2. Die Existenz- und Allquantoren von $\phi_{M,w}$ können durch die „beschränkten“ Quantoren $\exists x\,(x \leqslant 2^{O(T^2)})$ und $\forall x\,(x \leqslant 2^{O(T^2)})$ ersetzen werden. Warum?
   - Die Variablen der Formel kodieren Konfigurationen bzw. Konfigurationsfolgen von M.
   - Eine Konfiguration K wird durch eine Zahl $c_K \leqslant 2^{O(T)}$ kodiert $\Longrightarrow$ Eine Folge $\vec{k}$ von T Konfigurationen wird durch eine Zahl $d_{\vec{k}} = (2^{O(T)})^T = 2^{O(T^2)}$ kodiert.
3. Entferne die Multiplikation aus $\phi_{M,w}$. Aber wie?

Zeige: Sei p_n eine Zahl mit $p_n \geqslant 2^{2^n}$. Die Formel

$$m_n(a, b, c) := (a \cdot b = c \wedge a \leqslant p_n)$$

besitzt eine äquivalente Formel $m_n^*(a, b, c)$ der Presburger-Arithmetik, die in polynomieller Zeit berechenbar ist und lineare Länge in n hat.

Konstruktion von m_n^* (1/3)

Eine rekursive Definition:

(a) *Basisschritt*: Für $n = 0$ setze $p_0 := 2^{2^0} = 2$ und

$$\begin{aligned} m_0^*(a,b,c) \quad := \quad & ((a = 0 \wedge c = 0) \vee (a = S(0) \wedge c = b) \\ & \vee (a = S(S(0)) \wedge c = b + b)). \end{aligned}$$

(b) Für den *Rekursionsschritt* beachte

1. $a \cdot b = c \Leftrightarrow \exists a_1 \exists a_2 \exists a_3 \exists a_4 (((a_1 \cdot a_2 + a_3 + a_4) \cdot b = c) \wedge (a_1 \cdot a_2 + a_3 + a_4 = a))$.
2. Für die Definition von m_n^* dürfen wir $a_1, a_2, a_3, a_4 \leqslant \lfloor \sqrt{a} \rfloor$ annehmen. Warum?
   - ⋆ Wenn $a = x^2 - y$ für $1 \leqslant y < 2x - 1$, dann wähle $a_1 = a_2 = \lfloor \sqrt{a} \rfloor = x - 1 \Longrightarrow$
   - ⋆ $x^2 - 2x + 1 \leqslant a_1 \cdot a_2 + a_3 + a_4 \leqslant x^2 - 1$ folgt.

Statt der einen Multiplikation $a \cdot b = c$ großer Zahlen führe vier Multiplikationen

$$a_1 \cdot a_2 \cdot b, a_3 \cdot b, a_4 \cdot b \text{ und } a_1 \cdot a_2$$

für kleinere Zahlen aus.

Konstruktion von m_n^* (2/3)

Statt einer Multiplikation $a \cdot b = c$ großer Zahlen führe vier Multiplikationen $a_1 \cdot a_2 \cdot b, a_3 \cdot b, a_4 \cdot b$ und $a_1 \cdot a_2$ für kleinere Zahlen aus.

Breche die Dreier-Multiplikation $a_1 \cdot a_2 \cdot b$ in Zweier-Multiplikationen auf, nämlich

$$a_1 \cdot a_2 \cdot b = c_1 \Leftrightarrow \exists c_2 (a_1 \cdot c_2 = c_1 \wedge a_2 \cdot b = c_2).$$

Insgesamt haben wir also erhalten, dass

$$\begin{aligned} a \cdot b = c \quad \Leftrightarrow \quad & \exists\, a_1, a_2, a_3, a_4, c_1, c_2, c_3, c_4, d \;\; (c_1 + c_3 + c_4 = c \wedge d + a_3 + a_4 = a \\ \wedge \quad & (a_1 \cdot c_2 = c_1 \wedge a_2 \cdot b = c_2) \wedge a_3 \cdot b = c_3 \wedge a_4 \cdot b = c_4 \wedge a_1 \cdot a_2 = d). \end{aligned}$$

Wähle p_{n+1} so dass $\lfloor \sqrt{p_{n+1}} \rfloor = p_n \Longrightarrow p_0 = 2, p_{n+1} \geqslant p_n^2 \Longrightarrow p_n \geqslant 2^{2^n}$.

Wenn wir die Multiplikation $a \cdot b = c$ durch die fünf Formeln m_n^* ersetzen, dann erhalten wir die Längenrekursion

$$L(n+1) = 5 \cdot L(n) + O(1)$$

und damit leider eine **exponentielle Länge** in $n \Longrightarrow$ Pech gehabt, Niete $\lightning$

Konstruktion von m_n^* (3/3)

Benutze All-Quantoren, um die fünf Multiplikationen auf eine zu reduzieren.

$$
\begin{aligned}
m_{n+1}^*(a,b,c) \quad := \quad & \exists\, a_1, a_2, a_3, a_4, c_1, c_2, c_3, c_4, d\ \forall\, e, f, g \\
& (c_1 + c_3 + c_4 = c \wedge d + a_3 + a_4 = a) \\
\wedge \quad & [((e = a_1 \wedge f = c_2 \wedge g = c_1) \\
\vee \quad & (e = a_2 \wedge f = b \wedge g = c_2) \vee (e = a_3 \wedge f = b \wedge g = c_3) \\
\vee \quad & (e = a_4 \wedge f = b \wedge g = c_4) \vee (e = a_1 \wedge f = a_2 \wedge g = d)) \\
\rightarrow \quad & m_n^*(e, f, g)].
\end{aligned}
$$

m_n^* hat lineare Länge und kann in polynomieller Zeit konstruiert werden.