

# Logik II

Markus Lohrey

Universität Siegen

Sommersemester 2016

**Informationen** finden Sie unter

<http://www.eti.uni-siegen.de/ti/lehre/ss16/logikii/>

z. B.

- Aktuelle Version der Folien
- Übungsblätter

## **Literaturempfehlung:**

- Schöning: Logik für Informatiker, Spektrum Akademischer Verlag
- Ebbinghaus, Flum, Thomas: Einführung in die mathematische Logik, Spektrum Akademischer Verlag

Die **Übungen** werden von Herrn Danny Hucke organisiert.

## Definition (semi-entscheidbar)

Eine Sprache  $L \subseteq \Sigma^*$  ist **semi-entscheidbar**, falls es gibt einen Algorithmus mit folgenden Eigenschaften gibt:

Für alle  $x \in \Sigma^*$  gilt:

- Wenn  $x \in L$ , dann terminiert der Algorithmus bei Eingabe  $x$ .
- Wenn  $x \notin L$ , dann terminiert der Algorithmus bei Eingabe  $x$  nicht.

Äquivalenter Begriff: Rekursiv aufzählbar.

## Definition (rekursiv aufzählbar)

Eine Sprache  $L \subseteq \Sigma^*$  ist **rekursiv aufzählbar**, falls es gibt eine berechenbare totale Funktion  $f : \mathbb{N} \rightarrow \Sigma^*$  gibt mit  $L = \{f(i) \mid i \in \mathbb{N}\}$ .

## Definition (entscheidbar und unentscheidbar)

Eine Sprache  $L \subseteq \Sigma^*$  ist **entscheidbar**, falls es gibt einen Algorithmus mit folgenden Eigenschaften gibt: Für alle  $x \in \Sigma^*$  gilt:

- Wenn  $x \in L$ , dann terminiert der Algorithmus bei Eingabe  $x$  mit der Ausgabe “Ja”.
- Wenn  $x \notin L$ , dann terminiert der Algorithmus bei Eingabe  $x$  mit der Ausgabe “Nein”.

Eine Sprache  $L \subseteq \Sigma^*$  ist **unentscheidbar**, falls sie nicht entscheidbar ist.

## Satz

Eine Sprache  $L \subseteq \Sigma^*$  ist entscheidbar genau dann, wenn  $L$  und  $\Sigma^* \setminus L$  beide semi-entscheidbar sind.

# Wiederholung aus Logik I

Eine prädikatenlogische Formel  $F$  ist:

- **erfüllbar**, falls es eine zu  $F$  passende Struktur  $\mathcal{A}$  mit  $\mathcal{A} \models F$  gibt (d.h.  $F$  ist wahr in der Struktur  $\mathcal{A}$ ).
- **gültig**, falls  $\mathcal{A} \models F$  für jede zu  $F$  passende Struktur  $\mathcal{A}$  gilt.

## Konsequenz aus dem Satz von Gilmore

Die Menge der unerfüllbaren prädikatenlogischen Formeln ist semi-entscheidbar.

## Korollar

Die Menge der gültigen prädikatenlogischen Formeln ist semi-entscheidbar.

**Beweis:**  $F$  ist gültig, genau dann, wenn  $\neg F$  unerfüllbar ist.

# Unentscheidbarkeit der Prädikatenlogik

Wir wollen nun den folgenden zentralen Satz beweisen:

## Satz von Church

Die Menge der gültigen prädikatenlogischen Formeln ist unentscheidbar.

## Korollar

Die Menge der erfüllbaren prädikatenlogischen Formeln ist nicht semi-entscheidbar.

**Beweis:** Die Menge der unerfüllbaren Formeln ist semi-entscheidbar.

Wäre also die Menge der erfüllbaren Formeln semi-entscheidbar, so wäre sie entscheidbar.

Also wäre auch die Menge der unerfüllbaren Formeln und damit die Menge der gültigen Formeln entscheidbar.  $\square$

Wir beweisen den Satz von Church durch eine Reduktion vom Halteproblem für **Registermaschinenprogramme**.

Seien  $R_1, R_2, \dots$  Bezeichner für **Register**.

Intuition: Jedes Register speichert eine natürliche Zahl ab.

Eine **Registermaschinenprogramm** (kurz **RMP**)  $P$  besteht aus einer Folge  $A_1; A_2; \dots; A_l$  von Anweisungen, wobei  $A_l$  die Anweisung STOP ist, und für alle  $1 \leq i \leq l-1$  die Anweisung  $A_i$  von einem der folgenden Typen ist:

- $R_j := R_j + 1$  für ein  $1 \leq j \leq l$
- $R_j := R_j - 1$  für ein  $1 \leq j \leq l$
- IF  $R_j = 0$  THEN  $k_1$  ELSE  $k_2$  für  $1 \leq j, k_1, k_2 \leq l$ ,

Eine **Konfiguration** von  $P$  ist ein Tupel  $(i, n_1, \dots, n_l) \in \mathbb{N}^{l+1}$  mit  $1 \leq i \leq l$ .

Intuition:  $i$  ist die Nummer der Anweisung, die als nächste ausgeführt wird, und  $n_j$  ist der aktuelle Inhalt von Register  $R_j$ .

Für Konfigurationen  $(i, n_1, \dots, n_l)$  und  $(i', n'_1, \dots, n'_l)$  schreiben wir

$$(i, n_1, \dots, n_l) \rightarrow_P (i', n'_1, \dots, n'_l)$$

genau dann, wenn  $1 \leq i \leq l - 1$  und einer der folgenden Fälle gilt:

- $A_i = (R_j := R_j + 1)$  für ein  $1 \leq j \leq l$ ,  $i' = i + 1$ ,  $n'_j = n_j + 1$ ,  $n'_k = n_k$  für  $k \neq j$ .
- $A_i = (R_j := R_j - 1)$  für ein  $1 \leq j \leq l$ ,  $i' = i + 1$ ,  $n_j = n'_j = 0$  oder  $(n_j > 0, n'_j = n_j - 1)$ , und  $n'_k = n_k$  für  $k \neq j$ .
- $A_i = (\text{IF } R_j = 0 \text{ THEN } k_1 \text{ ELSE } k_2)$  für ein  $1 \leq j, k_1, k_2 \leq l$ ,  $n'_k = n_k$  für alle  $1 \leq k \leq l$ ,  $i' = k_1$  falls  $n_j = 0$ ,  $i' = k_2$  falls  $n_j > 0$ .

Wir definieren

$$\text{HALT} = \{P \mid P = A_1; A_2; \dots; A_l \text{ ist ein RMP mit } l \text{ Anweisungen, } (1, 0, \dots, 0) \rightarrow_P^* (l, n_1, \dots, n_l) \text{ für } n_1, \dots, n_l \geq 0\}$$



# Beweis des Satzes von Church

Registermaschinenprogramme entsprechen genau den GOTO-Programmen aus der GTI.

Dort haben wir gezeigt, dass eine Turingmaschine durch ein GOTO-Programm simuliert werden kann (und umgekehrt).

Da das Halteproblem für Turingmaschinen auf dem leeren Band (Hält eine Turingmaschine, wenn Sie mit dem leeren Band gestartet wird?) unentscheidbar ist, erhalten wir:

## Unentscheidbarkeit des Halteproblems für RMPs

Die Menge HALT ist unentscheidbar.

**Bemerkung:** HALT ist semi-entscheidbar: Simuliere ein gegebenes RMP auf der Startkonfiguration  $(1, 0, \dots, 0)$  und stoppe, wenn das RMP bei der STOP-Anweisung ankommt.

# Beweis des Satzes von Church

Wir beweisen den Satz von Church, indem wir jedem RMP  $P$  effektiv eine prädikatenlogischen Aussage  $F_P$  zuordnen, so dass gilt:

$$F_P \text{ ist gültig} \iff P \in \text{HALT}$$

Sei  $P = A_1; A_2; \dots; A_l$  ein RMP.

Wir fixieren folgende Symbole:

- $<$ : 2-stelliges Prädikatensymbol
- $c$ : Konstante
- $f, g$ : 1-stellige Funktionssymbole
- $R$ :  $(l + 2)$ -stelliges Prädikatensymbol

# Beweis des Satzes von Church

Wir definieren eine Struktur  $\mathcal{A}_P$  durch Fallunterscheidung:

1. Fall:  $P \notin \text{HALT}$ :

- Universum  $U_{\mathcal{A}_P} = \mathbb{N}$
- $<^{\mathcal{A}_P} = \{(n, m) \mid n < m\}$  (gewöhnliche Ordnung auf  $\mathbb{N}$ )
- $c^{\mathcal{A}_P} = 0$
- $f^{\mathcal{A}_P}(n) = n + 1$ ,  $g^{\mathcal{A}_P}(n + 1) = n$ ,  $g^{\mathcal{A}_P}(0) = 0$
- $R^{\mathcal{A}_P} = \{(s, i, n_1, \dots, n_l) \mid (1, 0, \dots, 0) \rightarrow_P^s (i, n_1, \dots, n_l)\}$

2. Fall:  $P \in \text{HALT}$ :

Sei  $t$  so, dass  $(1, 0, \dots, 0) \rightarrow_P^t (l, n_1, \dots, n_l)$  und  $e = \max\{t, l\}$ .

- Universum  $U_{\mathcal{A}_P} = \{0, 1, \dots, e\}$
- $<^{\mathcal{A}_P} = \{(n, m) \mid n < m\}$  (gewöhnliche Ordnung auf  $\{0, 1, \dots, e\}$ )
- $c^{\mathcal{A}_P} = 0$
- $f^{\mathcal{A}_P}(n) = n + 1$  für  $0 \leq n \leq e - 1$  und  $f^{\mathcal{A}_P}(e) = e$ .
- $g^{\mathcal{A}_P}(n + 1) = n$  für  $0 \leq n \leq e - 1$  und  $g^{\mathcal{A}_P}(0) = 0$ .
- $R^{\mathcal{A}_P} = \{(s, i, n_1, \dots, n_l) \mid 0 \leq s \leq t, (1, 0, \dots, 0) \rightarrow_P^s (i, n_1, \dots, n_l)\}$

# Beweis des Satzes von Church

Im folgenden verwenden wir die Abkürzung  $\bar{m}$  für den Term  $f^m(c)$ .

Wir definieren nun eine Aussage  $G_P$  (in der  $<, c, f, g$  und  $R$  vorkommen) mit folgenden Eigenschaften:

(A)  $\mathcal{A}_P \models G_P$

(B) Für jedes Modell  $\mathcal{A}$  von  $G_P$  gilt Folgendes:

Wenn  $(1, 0, \dots, 0) \rightarrow_P^s (i, n_1, \dots, n_l)$ , dann:

$$\mathcal{A} \models R(\bar{s}, \bar{i}, \bar{n}_1, \dots, \bar{n}_l) \wedge \bigwedge_{q=0}^{s-1} \bar{q} < \overline{q+1}.$$

Wir definieren

$$G_P = G_0 \wedge R(\bar{0}, \bar{1}, \bar{0}, \dots, \bar{0}) \wedge G_1 \wedge \dots \wedge G_{l-1}$$

wobei die Aussagen  $G_0, G_1, \dots, G_{l-1}$  wie folgt definiert sind.

# Beweis des Satzes von Church

$G_0$  sagt aus:

- $<$  ist eine lineare Ordnung mit kleinstem Element  $c$ ,
- $x \leq f(x)$  und  $g(x) \leq x$  für alle  $x$ ,
- für jedes  $x$ , das nicht das größte Element bzgl.  $<$  ist, ist  $f(x)$  der unmittelbare Nachfolger von  $x$ , und
- für jedes  $x$ , das nicht das kleinste Element  $c$  ist, ist  $g(x)$  der unmittelbare Vorgänger von  $x$ .

$$\begin{aligned} \forall x, y, z & ((\neg x < x) \wedge (x = y \vee x < y \vee y < x) \wedge ((x < y \wedge y < z) \rightarrow x < z) \\ & \wedge (x = c \vee c < x) \\ & \wedge (x = f(x) \vee x < f(x)) \\ & \wedge (x = g(x) \vee g(x) < x) \\ & \wedge (\exists u(x < u) \rightarrow (x < f(x) \wedge \forall u(x < u \rightarrow (u = f(x) \vee f(x) < u)))) \\ & \wedge (\exists u(u < x) \rightarrow (g(x) < x \wedge \forall u(u < x \rightarrow (u = g(x) \vee u < g(x))))) \end{aligned}$$

Bemerkung: Für jedes Modell  $\mathcal{A}$  von  $G_0$  gilt:

- $\mathcal{A} \models g(c) = c$
- $\mathcal{A} \models \forall x (\exists u (x < u) \rightarrow g(f(x)) = x)$

# Beweis des Satzes von Church

$G_i$  für  $1 \leq i \leq l - 1$  beschreibt die Wirkung der Anweisung  $A_i$ .

1. Fall:  $A_i = (R_j := R_j + 1)$ . Dann sei

$$G_i = \forall x \forall x_1 \cdots \forall x_l \left( R(x, \bar{i}, x_1, \dots, x_l) \rightarrow \right. \\ \left. (x < f(x) \wedge R(f(x), \overline{i+1}, x_1, \dots, x_{j-1}, f(x_j), x_{j+1}, \dots, x_l)) \right)$$

2. Fall:  $A_i = (R_j := R_j - 1)$ . Dann sei

$$G_i = \forall x \forall x_1 \cdots \forall x_l \left( R(x, \bar{i}, x_1, \dots, x_l) \rightarrow \right. \\ \left. (x < f(x) \wedge R(f(x), \overline{i+1}, x_1, \dots, x_{j-1}, g(x_j), x_{j+1}, \dots, x_l)) \right)$$

# Beweis des Satzes von Church

3. Fall:  $A_j = (\text{IF } R_j = 0 \text{ THEN } k_1 \text{ ELSE } k_2)$  für ein  $1 \leq j, k_1, k_2 \leq l$ .  
Dann sei

$$G_j = \forall x \forall x_1 \cdots \forall x_l \left( R(x, \bar{i}, x_1, \dots, x_l) \rightarrow (x < f(x) \wedge (x_j = c \wedge R(f(x), \bar{k}_1, x_1, \dots, x_l)) \vee (x_j > c \wedge R(f(x), \bar{k}_2, x_1, \dots, x_l))) \right)$$

Aussage (A) folgt sofort aus der Definition von  $\mathcal{A}_P$  und  $G_P$ .

Aussage (B) beweisen wir durch eine Induktion über  $s$ .

IA:  $s = 0$ . Gelte  $(1, 0, \dots, 0) \rightarrow_P^0 (i, n_1, \dots, n_l)$ , d.h.  $i = 1$  und  $n_1 = n_2 = \dots = n_l = 0$ .

Aus  $\mathcal{A} \models G_P$  folgt  $\mathcal{A} \models R(\bar{0}, \bar{1}, \bar{0}, \dots, \bar{0})$ , d. h.  $\mathcal{A} \models R(\bar{s}, \bar{i}, \bar{n}_1, \dots, \bar{n}_l)$ .



# Beweis des Satzes von Church

IS: Sei nun  $s > 0$  und gelte Aussage (B) für  $s - 1$ .

Sei  $(1, 0, \dots, 0) \rightarrow_P^s (i, n_1, \dots, n_l)$ .

Dann gibt es  $j, m_1, \dots, m_l$  mit

$$(1, 0, \dots, 0) \rightarrow_P^{s-1} (j, m_1, \dots, m_l) \rightarrow_P (i, n_1, \dots, n_l)$$

Aus der IH folgt

$$\mathcal{A} \models R(\overline{s-1}, \bar{j}, \overline{m_1}, \dots, \overline{m_l}) \wedge \bigwedge_{q=0}^{s-2} \bar{q} < \overline{q+1}.$$

Wir machen nun eine Fallunterscheidung bezüglich der Anweisung  $A_j$ , wobei wir nur den Fall betrachten, dass  $A_j$  von der Form  $R_k := R_k - 1$  ist.

Es gilt dann  $i = j + 1$ ,  $n_1 = m_1, \dots, n_{k-1} = m_{k-1}$ ,

$n_{k+1} = m_{k+1}, \dots, n_l = m_l$ , ( $n_k = m_k = 0$  oder  $m_k > 0$  und  $n_k = m_k - 1$ ).

# Beweis des Satzes von Church

Wegen  $\mathcal{A} \models G_j$  gilt:

$$\mathcal{A} \models \forall y, y_1, \dots, y_l \left( R(y, \bar{j}, y_1, \dots, y_l) \rightarrow \right. \\ \left. (y < f(y) \wedge R(f(y), \overline{j+1}, y_1, \dots, y_{k-1}, g(y_k), y_{k+1}, \dots, y_l)) \right)$$

Wegen  $\mathcal{A} \models R(\overline{s-1}, \bar{j}, \overline{m_1}, \dots, \overline{m_l})$  folgt

$$\mathcal{A} \models \overline{s-1} < f(\overline{s-1}) \wedge \\ R(f(\overline{s-1}), \overline{j+1}, \overline{m_1}, \dots, \overline{m_{k-1}}, g(\overline{m_k}), \overline{m_{k+1}}, \dots, \overline{m_l})$$

d.h.

$$\mathcal{A} \models \overline{s-1} < \bar{s} \wedge R(\bar{s}, \bar{i}, \overline{n_1}, \dots, \overline{n_{k-1}}, g(\overline{m_k}), \overline{n_{k+1}}, \dots, \overline{n_l})$$

# Beweis des Satzes von Church

Wegen  $\mathcal{A} \models \overline{s-1} < \bar{s}$  gilt

$$\mathcal{A} \models \bigwedge_{q=0}^{s-1} \bar{q} < \overline{q+1}.$$

Ausserdem folgt aus  $\mathcal{A} \models G_0$ , dass  $\mathcal{A} \models g(\overline{m_k}) = \overline{n_k}$ .

Also gilt auch  $\mathcal{A} \models R(\bar{s}, \bar{i}, \bar{n}_1, \dots, \bar{n}_l)$ .

Damit sind (A) und (B) gezeigt.

## **Beweis des Satzes von Church:**

Setze  $F_P = (G_P \rightarrow \exists x \exists x_1 \dots \exists x_l R(x, \bar{l}, x_1, \dots, x_l))$

Behauptung:  $F_P$  ist gültig  $\iff P \in \text{HALT}$ .

# Beweis des Satzes von Church

Ist  $F_P$  gültig, so gilt insbesondere  $\mathcal{A}_P \models F_P$ .

Wegen (A) gilt  $\mathcal{A}_P \models \exists x \exists x_1 \cdots \exists x_l R(x, \bar{l}, x_1, \dots, x_l)$ .

Also gibt es  $s, n_1, \dots, n_l \geq 0$  mit  $(s, l, n_1, \dots, n_l) \in R^{\mathcal{A}_P}$ .

Es folgt  $P \in \text{HALT}$ .

Sei nun  $P \in \text{HALT}$  und gelte  $(1, 0, \dots, 0) \rightarrow_P^s (l, n_1, \dots, n_l)$

Sei  $\mathcal{A}$  eine Struktur mit  $\mathcal{A} \models G_P$ .

Aus (B) folgt  $\mathcal{A} \models R(\bar{s}, \bar{l}, \bar{n}_1, \dots, \bar{n}_l)$ .

Also ist  $F_P$  gültig. □

# Der Satz von Trachtenbrot

Eine Formel  $F$  ist **im Endlichen erfüllbar** genau dann, wenn  $F$  ein Modell mit einem endlichen Universum hat, sonst ist  $F$  **im Endlichen unerfüllbar**.

## Lemma

Die Menge der im Endlichen erfüllbaren Formeln ist semi-entscheidbar.

### Beweis:

Sei  $\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3, \dots$  eine systematische Auflistung aller endlichen zu  $F$  passenden Strukturen (o.B.d.A. ist  $I_{\mathcal{A}_i}$  nur auf den in  $F$  vorkommenden Prädikaten- und Funktionssymbolen definiert).

Folgender Algorithmus terminiert genau dann, wenn  $F$  im endlichen erfüllbar ist:

```
 $i := 1;$   
while true do  
  if  $\mathcal{A}_i \models F$  then STOP else  $i := i + 1$   
end
```



# Der Satz von Trachtenbrot

Eine Formel  $F$  ist **im Endlichen gültig** genau dann, wenn jede endliche zu  $F$  passende Struktur ein Modell von  $F$  ist.

**Beispiel:** Die Formel

$$\forall x \forall y (f(x) = f(y) \rightarrow x = y) \leftrightarrow \forall y \exists x (f(x) = y)$$

ist im Endlichen gültig, aber nicht (allgemein) gültig.

## Satz von Trachtenbrot

Die Menge der im Endlichen erfüllbaren Formeln ist unentscheidbar.

## Korollar

Die Menge der im Endlichen unerfüllbaren Formeln sowie die Menge der im Endlichen gültigen Formeln ist nicht semi-entscheidbar.

## Beweis des Satzes von Trachtenbrot:

Wir verwenden die Konstruktion aus dem Beweis des Satzes von Church.

Behauptung:  $G_P$  ist im Endlichen erfüllbar  $\iff P \in \text{HALT}$ .

(1) Gelte  $P \in \text{HALT}$ .

Dann ist  $\mathcal{A}_P$  endlich und es gilt  $\mathcal{A}_P \models G_P$  nach Aussage (A).

Also ist  $G_P$  im Endlichen erfüllbar.

# Der Satz von Trachtenbrot

(2) Sei  $G_P$  im Endlichen erfüllbar.

Sei  $\mathcal{A}$  eine endliche Struktur mit  $\mathcal{A} \models G_P$ .

Angenommen  $P \notin \text{HALT}$  gilt.

Also gibt es für jede Zahl  $s \geq 0$  Zahlen  $i, n_1, \dots, n_l$  mit  $(1, 0, \dots, 0) \rightarrow_P^s (i, n_1, \dots, n_l)$ .

Aussage (B) impliziert, dass  $\mathcal{A} \models \bar{i} < \overline{i+1}$  für alle  $i \geq 0$ .

Da  $<^{\mathcal{A}}$  eine lineare Ordnung ist (wegen  $\mathcal{A} \models G_0$ ) ist die Menge  $\{\mathcal{A}(\bar{i}) \mid i \geq 0\}$  unendlich, was ein Widerspruch ist. □



# (Un)entscheidbare Theorien

Sei  $\mathcal{A}$  eine Struktur, wobei der Definitionsbereich von  $I_{\mathcal{A}}$  endlich sei und keine Variablen enthält.

Sei  $f_1, \dots, f_n, R_1, \dots, R_m$  der Definitionsbereich von  $I_{\mathcal{A}}$ .

Wir identifizieren dann  $\mathcal{A}$  mit dem Tupel  $(U^{\mathcal{A}}, f_1^{\mathcal{A}}, \dots, f_n^{\mathcal{A}}, R_1^{\mathcal{A}}, \dots, R_m^{\mathcal{A}})$ , wofür wir auch  $(U^{\mathcal{A}}, f_1, \dots, f_n, R_1, \dots, R_m)$  schreiben.

## Definition

Die **Theorie von  $\mathcal{A}$**  ist die Menge von Formeln

$$\text{Th}(\mathcal{A}) = \{F \mid F \text{ ist eine Aussage, } \mathcal{A} \text{ passt zu } F, \mathcal{A} \models F\}.$$

Wir interessieren uns für die Frage, ob eine Struktur eine entscheidbare Theorie hat.

## Satz

*Sei  $\mathcal{A}$  eine beliebige Struktur. Dann ist  $\text{Th}(\mathcal{A})$  entscheidbar genau dann, wenn  $\text{Th}(\mathcal{A})$  semi-entscheidbar ist.*

**Beweis:** Sei  $\text{Th}(\mathcal{A})$  semi-entscheidbar und sei  $F$  eine beliebige Aussage. Dann gilt entweder  $F \in \text{Th}(\mathcal{A})$  oder  $\neg F \in \text{Th}(\mathcal{A})$ .

Wir können daher einen Semi-Entscheidungsalgorithmus für  $\text{Th}(\mathcal{A})$  mit Eingabe  $F$  und  $\neg F$  parallel laufen lassen.

Einer der beiden Läufe wird irgendwann mit der Antwort terminieren.  $\square$

# (Un)entscheidbare Theorien

Für die Frage nach der Entscheidbarkeit einer Struktur können wir uns auf sogenannte **relationale Strukturen** beschränken.

Eine Struktur  $\mathcal{A} = (A, f_1, \dots, f_n, R_1, \dots, R_m)$  ist **relational**, falls  $n = 0$  gilt.

Für eine beliebige Struktur  $\mathcal{A} = (A, f_1, \dots, f_n, R_1, \dots, R_m)$  definieren wir

$$\mathcal{A}_{\text{rel}} = (A, P_1, \dots, P_n, R_1, \dots, R_m)$$

wobei

$$P_i = \{(a_1, \dots, a_n, a) \mid f_i(a_1, \dots, a_n) = a\}.$$

## Lemma

$\text{Th}(\mathcal{A})$  ist entscheidbar genau dann, wenn  $\text{Th}(\mathcal{A}_{\text{rel}})$  entscheidbar ist.

**Beweis:** Übung.

# Unentscheidbarkeit der Arithmetik (nach Ebbinghaus, Flum, Thomas)

Satz (Gödel 1931)

$\text{Th}(\mathbb{N}, +, \cdot)$  ist unentscheidbar.

Korollar

$\text{Th}(\mathbb{N}, +, \cdot)$  ist nicht semi-entscheidbar, also nicht rekursiv aufzählbar.

Für den Beweis reduzieren wir die Menge HALT von terminierenden RMPs auf  $\text{Th}(\mathbb{N}, +, \cdot)$ .

Um den Beweis etwas komfortabler zu machen, betrachten wir  $\text{Th}(\mathbb{N}, +, \cdot, s, 0)$  mit  $s(n) = n + 1$ .

Übung:  $\text{Th}(\mathbb{N}, +, \cdot, s, 0)$  ist unentscheidbar genau dann, wenn  $\text{Th}(\mathbb{N}, +, \cdot)$  unentscheidbar ist.

# Unentscheidbarkeit der Arithmetik

Sei nun  $P = A_1; A_2; \dots; A_l$  ein RMP, in dem die Register  $R_1, \dots, R_l$  verwendet werden.

Wir konstruieren eine arithmetische Formel  $F_P$  mit den freien Variablen  $x, x_1, \dots, x_l$ , so dass für alle  $1 \leq i \leq l$  und  $n_1, \dots, n_l \in \mathbb{N}$  folgende beiden Aussagen äquivalent sind:

- $(\mathbb{N}, +, \cdot, s, 0)_{[x/i, x_1/n_1, \dots, x_l/n_l]} \models F_P$
- $(1, 0, \dots, 0) \rightarrow_P^* (i, n_1, \dots, n_l)$

Dann gilt  $P \in \text{HALT} \iff (\mathbb{N}, +, \cdot, s, 0) \models \exists x_1 \dots \exists x_l F_P[x/s^l(0)]$ .

# Unentscheidbarkeit der Arithmetik

Intuitiv sagt die Formel  $F_P$  Folgendes aus:

Es gibt ein  $s \geq 0$  und Konfigurationen  $C_0, C_1, \dots, C_s$  mit:

- $C_0 = (1, 0, \dots, 0)$
- $C_s = (x, x_1, \dots, x_l)$
- $C_i \rightarrow_P C_{i+1}$  für alle  $0 \leq i \leq s - 1$

Wir können die  $(l + 1)$ -Tupel  $C_0, C_1, \dots, C_s$  durch ein  $(s + 1)(l + 1)$ -Tupel kodieren, und müssen dann Folgendes ausdrücken, wobei  $k = l + 1$  sei.

Es gibt ein  $s \geq 0$  und ein Tupel

$(y_0, y_1, \dots, y_{k-1}, y_k, y_{k+1}, \dots, y_{2k-1}, \dots, y_{sk}, y_{sk+1}, \dots, y_{(s+1)k-1})$  mit:

- $y_0 = 1, y_1 = 0, \dots, y_{k-1} = 0$
- $y_{sk} = x, y_{sk+1} = x_1, \dots, y_{(s+1)k-1} = x_l$
- $(y_{ik}, \dots, y_{(i+1)k-1}) \rightarrow_P (y_{(i+1)k}, \dots, y_{(i+2)k-1})$  für alle  $0 \leq i \leq s - 1$

# Unentscheidbarkeit der Arithmetik

Will man dies durch eine arithmetische Formel ausdrücken, hat man das Problem, dass man nicht über Folgen von Zahlen quantifizieren kann ( $\exists y \exists x_1 \cdots \exists x_y$  ist nicht zulässig).

Um trotzdem eine Quantifizierung über beliebig lange Folgen zu simulieren, benötigen wir Gödels  $\beta$ -Funktion.

## Lemma

Es gibt eine Funktion  $\beta : \mathbb{N}^3 \rightarrow \mathbb{N}$  mit:

- Für jede Folge  $(a_0, \dots, a_r)$  über  $\mathbb{N}$  gibt es  $t, p \in \mathbb{N}$ , so dass  $\beta(t, p, i) = a_i$  für alle  $0 \leq i \leq r$
- Es gibt eine arithmetische Formel  $B$  mit freien Variablen  $v, x, y, z$ , so dass für alle  $t, p, i, a \in \mathbb{N}$  gilt:

$$(\mathbb{N}, +, \cdot, s, 0)_{[v/t, x/p, y/i, z/a]} \models B \iff \beta(t, p, i) = a$$

Man sagt auch:  $\beta$  ist arithmetisch definierbar.

## Beweis des Lemmas:

Sei  $(a_0, \dots, a_r)$  eine beliebige Folge über  $\mathbb{N}$ .

Sei  $p$  eine Primzahl mit  $p > r + 1$  und  $p > a_i$  für alle  $i$ .

Sei weiter

$$t = 1p^0 + a_0p^1 + 2p^2 + a_1p^3 + \dots + (i+1)p^{2i} + a_ip^{2i+1} + \dots + (r+1)p^{2r} + a_rp^{2r+1}.$$

D.h.  $(1, a_0, 2, a_1, \dots, (i+1), a_i, \dots, (r+1), a_r)$  ist die Darstellung von  $t$  zur Basis  $p$ .

**Behauptung:** Für alle  $a \in \mathbb{N}$  und alle  $0 \leq i \leq r$  gilt  $a = a_i$  genau dann, wenn es  $b_0, b_1, b_2 \in \mathbb{N}$  gibt mit:

(a)  $t = b_0 + b_1((i+1) + ap + b_2p^2)$

(b)  $a < p$

(c)  $b_0 < b_1$

(d) Es gibt ein  $m$  mit  $b_1 = p^{2m}$ .



# Unentscheidbarkeit der Arithmetik

⇒: Wenn  $a = a_i$  dann können wir  $b_0, b_1, b_2$  wie folgt wählen:

$$b_0 = 1p^0 + a_0p^1 + 2p^2 + a_1p^3 + \dots + ip^{2i-2} + a_{i-1}p^{2i-1}$$

$$b_1 = p^{2i}$$

$$b_2 = (i + 2) + a_{i+1}p + \dots + a_r p^{2(r-i)-1}$$

⇐: Gelte (a)-(d), d.h.

$$\begin{aligned} t &= b_0 + b_1((i + 1) + ap + b_2p^2) \\ &= b_0 + (i + 1)p^{2m} + ap^{2m+1} + p^{2m+2}b_2. \end{aligned}$$

wobei  $b_0 < b_1 = p^{2m}$ ,  $a < p$  und  $(i + 1) < p$ .

Ein Vergleich mit

$$t = 1p^0 + a_0p^1 + 2p^2 + a_1p^3 + \dots + (i+1)p^{2i} + a_ip^{2i+1} + \dots + (r+1)p^{2r} + a_rp^{2r+1}$$

liefert  $m = i$  und  $a = a_i$ .

# Unentscheidbarkeit der Arithmetik

Da  $p$  eine Primzahl ist, ist (d) äquivalent zu:  $b_1$  ist ein Quadrat, und für alle  $d \geq 2$  mit  $d|b_1$  gilt  $p|d$ .

Wir definieren nun für alle Zahlen  $t, p, i \in \mathbb{N}$  die Zahl  $\beta(t, p, i)$  als die kleinste Zahl  $a$ , so dass  $b_0, b_1, b_2 \in \mathbb{N}$  existieren mit:

(a)  $t = b_0 + b_1((i + 1) + ap + b_2p^2)$ ,

(b)  $a < p$ ,

(c)  $b_0 < b_1$ ,

(d)  $b_1$  ist ein Quadrat, und für alle  $d \geq 2$  mit  $d|b_1$  gilt  $p|d$ .

Sollten solche Zahlen  $b_0, b_1, b_2 \in \mathbb{N}$  nicht existieren, so setzen wir  $\beta(t, p, i) = 0$ .

Aus der gerade gezeigten Behauptung folgt dann: Für jede Folge  $(a_0, \dots, a_r)$  über  $\mathbb{N}$  gibt es  $t, p \in \mathbb{N}$ , so dass  $\beta(t, p, i) = a_i$  für alle  $0 \leq i \leq r$ .

Außerdem ist  $\beta$  offensichtlich arithmetisch definierbar. □

# Unentscheidbarkeit der Arithmetik

Wir können nun den Beweis für die Unentscheidbarkeit der Arithmetik beenden.

Wir müssen folgende Aussage durch eine arithmetische Formel (mit freien Variablen  $x, x_1, \dots, x_l$ ) ausdrücken:

Es gibt ein  $s$  und ein Tupel

$(y_0, y_1, \dots, y_{k-1}, y_k, y_{k+1}, \dots, y_{2k-1}, \dots, y_{sk}, y_{sk+1}, \dots, y_{(s+1)k-1})$  mit:

- $y_0 = 1, y_1 = 0, \dots, y_{k-1} = 0$
- $y_{sk} = x, y_{sk+1} = x_1, \dots, y_{(s+1)k-1} = x_l$
- $(y_{ik}, \dots, y_{(i+1)k-1}) \rightarrow_P (y_{(i+1)k}, \dots, y_{(i+2)k-1})$  für alle  $0 \leq i \leq s-1$

Beachte:  $k = l + 1$  ist hierbei eine Konstante, die durch das RMP  $P$  festgelegt ist.

Dies ist äquivalent zu: Es gibt  $s, t, p$  mit:

- $\beta(t, p, 0) = 1, \beta(t, p, 1) = 0, \dots, \beta(t, p, k - 1) = 0$
- $\beta(t, p, sk) = x, \beta(t, p, sk + 1) = x_1, \dots, \beta(t, p, (s + 1)k - 1) = x_l$
- Für alle  $0 \leq i \leq s - 1$  gilt:

$$\left( \beta(t, p, ik), \dots, \beta(t, p, (i + 1)k - 1) \right) \rightarrow_P$$
$$\left( \beta(t, p, (i + 1)k), \dots, \beta(t, p, (i + 2)k - 1) \right)$$

Eine arithmetische Formel für  $(y, y_1, \dots, y_l) \rightarrow_P (x, x_1, \dots, x_l)$  ist einfach als Disjunktion über alle Anweisungen  $A_i$  des RMPs  $P$  anzugeben (Übung). □

Wir werden im folgenden **automatische Strukturen** einführen.

Die Hauptresultate zu automatische Strukturen, die wir beweisen, sind:

- Jede automatische Struktur hat eine entscheidbare Theorie.
- $(\mathbb{N}, +)$  ist automatisch.
- $(\mathbb{Q}, \leq)$  ist automatisch.

# Konvolution von Wörtern

Sei  $n \geq 1$ . Sei  $\Sigma$  ein endliches Alphabet und sei  $\# \notin \Sigma$ .

Sei  $\Sigma_{\#} = \Sigma \cup \{\#\}$  im Weiteren.

Seien  $w_1, w_2, \dots, w_n \in \Sigma^*$ . Wir definieren die **Konvolution**

$$w_1 \otimes w_2 \otimes \dots \otimes w_n \in (\Sigma_{\#}^n)^*$$

wie folgt:

- Sei  $w_i = a_{i,1}a_{i,2} \dots a_{i,\ell_i}$ , d.h.  $\ell_i = |w_i|$ .
- Sei  $\ell = \max\{\ell_1, \dots, \ell_n\}$
- Für alle  $1 \leq i \leq n$  und  $\ell_i < j \leq \ell$  sei  $a_{i,j} = \#$
- $w_1 \otimes w_2 \otimes \dots \otimes w_n := (a_{1,1}, \dots, a_{n,1})(a_{1,2}, \dots, a_{n,2}) \dots (a_{1,\ell}, \dots, a_{n,\ell})$ .

**Beispiel:**  $abba \otimes babaaa = (a, b)(b, a)(b, b)(a, a)(\#, a)(\#, a)$

# Synchrone Mehrbandautomaten

Ein **synchroner  $n$ -Bandautomat**  $A$  über dem Alphabet  $\Sigma$  ist ein gewöhnlicher endlicher Automat über dem Alphabet  $\Sigma_{\#}^n$ .

$$\leadsto L(A) \subseteq (\Sigma_{\#}^n)^*.$$

Es sei  $K(A) = \{(w_1, \dots, w_n) \mid w_1, \dots, w_n \in \Sigma^*, w_1 \otimes \dots \otimes w_n \in L(A)\}$ .

Eine  $n$ -stellige Relation  $R$  über  $\Sigma^*$  ist **synchron-rational**, falls ein synchroner  $n$ -Bandautomat  $A$  mit  $K(A) = R$  existiert.

Beachte: Elemente in  $L(A)$  die nicht zu  $\{w_1 \otimes \dots \otimes w_n \mid w_1, \dots, w_n \in \Sigma^*\}$  gehören, haben keinen Einfluss auf die Relation  $K(A)$  (es handelt sich sozusagen um Müll).

Man kann aus  $A$  jedoch leicht einen synchronen  $n$ -Bandautomaten  $B$  mit  $L(B) = L(A) \cap \{w_1 \otimes \dots \otimes w_n \mid w_1, \dots, w_n \in \Sigma^*\}$  konstruieren.

Beachte:  $\{w_1 \otimes \dots \otimes w_n \mid w_1, \dots, w_n \in \Sigma^*\} \subseteq (\Sigma_{\#}^n)^*$  ist regulär.

# Synchrone Mehrbandautomaten

Veranschaulichung der Arbeitsweise eines synchronen Mehrbandautomaten:

$v$	$b_0$	$b_1$	$b_2$	$\dots$	$b_{m-1}$	$b_m$	$\#$	$\dots$	$\#$
$u$	$a_0$	$a_1$	$a_2$	$\dots$	$a_{m-1}$	$a_m$	$a_{m+1}$	$\dots$	$a_n$



# Synchrone Mehrbandautomaten

Veranschaulichung der Arbeitsweise eines synchronen Mehrbandautomaten:

	$q_0$								
$v$	$b_0$	$b_1$	$b_2$	$\dots$	$b_{m-1}$	$b_m$	$\#$	$\dots$	$\#$
$u$	$a_0$	$a_1$	$a_2$	$\dots$	$a_{m-1}$	$a_m$	$a_{m+1}$	$\dots$	$a_n$

# Synchrone Mehrbandautomaten

Veranschaulichung der Arbeitsweise eines synchronen Mehrbandautomaten:

		$q_1$							
$v$	$b_0$	$b_1$	$b_2$	$\dots$	$b_{m-1}$	$b_m$	$\#$	$\dots$	$\#$
$u$	$a_0$	$a_1$	$a_2$	$\dots$	$a_{m-1}$	$a_m$	$a_{m+1}$	$\dots$	$a_n$

# Synchrone Mehrbandautomaten

Veranschaulichung der Arbeitsweise eines synchronen Mehrbandautomaten:

		$q_2$							
$v$	$b_0$	$b_1$	$b_2$	$\dots$	$b_{m-1}$	$b_m$	$\#$	$\dots$	$\#$
$u$	$a_0$	$a_1$	$a_2$	$\dots$	$a_{m-1}$	$a_m$	$a_{m+1}$	$\dots$	$a_n$

# Synchrone Mehrbandautomaten

Veranschaulichung der Arbeitsweise eines synchronen Mehrbandautomaten:

						$q_m$			
$v$	$b_0$	$b_1$	$b_2$	$\dots$	$b_{m-1}$	$b_m$	$\#$	$\dots$	$\#$
$u$	$a_0$	$a_1$	$a_2$	$\dots$	$a_{m-1}$	$a_m$	$a_{m+1}$	$\dots$	$a_n$

# Synchrone Mehrbandautomaten

Veranschaulichung der Arbeitsweise eines synchronen Mehrbandautomaten:

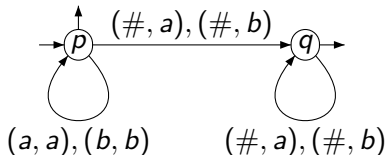
						$q_{m+1}$			
$v$	$b_0$	$b_1$	$b_2$	$\dots$	$b_{m-1}$	$b_m$	#	$\dots$	#
$u$	$a_0$	$a_1$	$a_2$	$\dots$	$a_{m-1}$	$a_m$	$a_{m+1}$	$\dots$	$a_n$

# Synchrone Mehrbandautomaten

Veranschaulichung der Arbeitsweise eines synchronen Mehrbandautomaten:

								$q_n$	
$v$	$b_0$	$b_1$	$b_2$	$\dots$	$b_{m-1}$	$b_m$	$\#$	$\dots$	$\#$
$u$	$a_0$	$a_1$	$a_2$	$\dots$	$a_{m-1}$	$a_m$	$a_{m+1}$	$\dots$	$a_n$

**Beispiel 1:** Sei  $A$  der folgende synchrone 2-Bandautomat:



Es gilt  $K(A) = \{(u, v) \mid u, v \in \{a, b\}^*, u \text{ ist Präfix von } v\}$ .

## Definition

Eine relationale Struktur  $\mathcal{A} = (A, R_1, \dots, R_m)$  (wobei  $R_i$  eine  $n_i$ -stellige Relation ist) ist **automatisch**, falls ein endliches Alphabet  $\Sigma$ , ein endlicher Automat  $B$  über dem Alphabet  $\Sigma$  und synchrone  $n_i$ -Bandautomaten  $B_i$  über dem Alphabet  $\Sigma$  ( $1 \leq i \leq m$ ) existieren mit:

- $L(B) = A$
- $K(B_i) = R_i$  für  $1 \leq i \leq m$

## Definition

Eine Struktur  $\mathcal{A}$  ist **automatisch präsentierbar**, falls  $\mathcal{A}$  isomorph zu einer automatischen Struktur ist.



# $(\mathbb{N}, +)$ ist automatisch

## Satz

$(\mathbb{N}, +)$  mit  $+ = \{(a, b, c) \mid a + b = c\}$  ist automatisch präsentierbar.

**Beweis:** Sei  $A$  ein endlicher Automat mit  $L(A) = \{0\} \cup \{0, 1\}^*1$ .

Dann ist die folgende Abbildung  $h : L(A) \rightarrow \mathbb{N}$  eine Bijektion:

$h(w) =$  die durch  $w$  repräsentierte Binärzahl, rückwärts gelesen

Sei  $B_+$  der synchrone 3-Bandautomat auf der nächsten Folie.

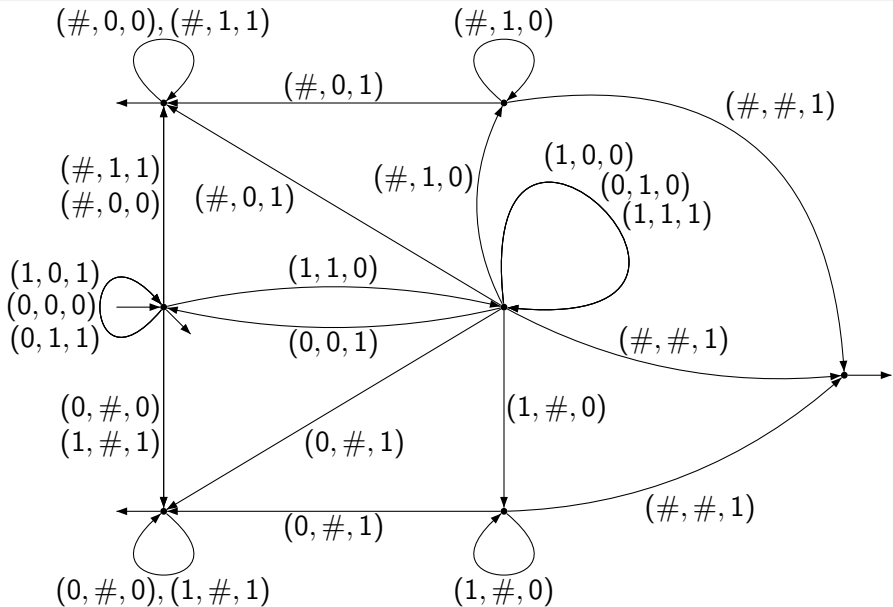
$B_+$  erkennt “fast” die Relation  $\{(u, v, w) \in L(A)^3 \mid h(u) + h(v) = h(w)\}$ , es gilt z. B.  $(00, 0000, 0000) \in K(B_+)$ .

Sei  $A_+$  ein synchroner 3-Bandautomat mit

$$L(A_+) = L(B_+) \cap \{u \otimes v \otimes w \mid u, v, w \in L(A)\}.$$

Dann gilt  $K(A_+) = \{(u, v, w) \in L(A)^3 \mid h(u) + h(v) = h(w)\}$ . □

# $(\mathbb{N}, +)$ ist automatisch



## Weitere Beispiele

Man kann den vorherigen Satz noch erweitern: Sei  $p > 1$  und  $(\mathbb{N}, +, |_p)$ , wobei  $x |_p y$  genau dann, wenn  $\exists n, k \in \mathbb{N} : x = p^n, y = k \cdot x$ , ist automatisch präsentierbar.

### Satz

$(\mathbb{Q}, \leq)$  ist automatisch präsentierbar.

Für den Beweis benutzen wir den Satz von Cantor.

Eine lineare Ordnung  $(A, \leq)$  ist **dicht** falls gilt:

$$\forall x \forall y (x < y \rightarrow \exists z (x < z < y)).$$

### Satz von Cantor

Seien  $(A, \leq_A)$  und  $(B, \leq_B)$  zwei abzählbare dichte lineare Ordnungen ohne kleinstes Element und ohne größtes Element. Dann sind  $(A, \leq_A)$  und  $(B, \leq_B)$  isomorph.

## Beweis des Satzes von Cantor:

Wir konstruieren Auflistungen

$$a_1, a_2, a_3, a_4, \dots \text{ und } b_1, b_2, b_3, b_4, \dots$$

mit folgenden Eigenschaften:

- $a_i \neq a_j$  und  $b_i \neq b_j$  für  $i \neq j$
- $A = \{a_i \mid i \geq 1\}$  und  $B = \{b_i \mid i \geq 1\}$
- $a_i < a_j$  genau dann wenn  $b_i < b_j$  für alle  $i, j$ .

Dann ist  $f : A \rightarrow B$  mit  $f(a_i) = b_i$  ein Isomorphismus.

Da  $A$  und  $B$  abzählbar unendlich sind, können wir beide Mengen auflisten:

$$A = \{x_1, x_2, x_3, \dots\} \text{ und } B = \{y_1, y_2, y_3, \dots\}$$

Der folgende "Algorithmus" konstruiert die Auflistungen:

# Satz von Cantor

$L_A := [x_1, x_2, x_3, \dots]$ ;  $L_B := [y_1, y_2, y_3, \dots]$

**for all**  $i \geq 1$  **do** ( $a_1, \dots, a_{i-1}, b_1, \dots, b_{i-1}$  sind bereits definiert)

**if**  $i$  ist ungerade **then**

sei  $x$  das erste Element aus  $L_A$

entferne  $x$  aus der Liste  $L_A$

sei  $y$  ein Element aus  $L_B$  mit folgender Eigenschaft:

$\forall 1 \leq j \leq i-1 : a_j < x \iff b_j < y$

entferne  $y$  aus der Liste  $L_B$

$a_i := x$ ;  $b_i := y$

**else**

sei  $y$  das erste Element aus  $L_B$

entferne  $y$  aus der Liste  $L_B$

sei  $x$  ein Element aus  $L_A$  mit folgender Eigenschaft:

$\forall 1 \leq j \leq i-1 : a_j < x \iff b_j < y$

entferne  $x$  aus der Liste  $L_A$

$a_i := x$ ;  $b_i := y$

**endfor**

# $(\mathbb{Q}, \leq)$ ist automatisch

## Beweis, dass $(\mathbb{Q}, \leq)$ automatisch ist:

Auf Grund des Satzes von Cantor genügt es, eine abzählbare dichte automatische lineare Ordnung ohne kleinstes und größtes Element anzugeben.

Sei hierzu  $L = \{0, 1\}^*$ .

Sei  $\leq$  die lexikographische Ordnung auf  $L$ , d.h. für  $x, y \in L$  gilt  $x \leq y$  genau dann, wenn einer der folgenden Fälle gilt:

- Es gibt ein  $u \in \{0, 1\}^*$  mit  $y = xu$  ( $x$  ist Anfangsstück von  $y$ )
- Es gibt  $z, u, v \in \{0, 1\}^*$  mit  $x = z0u$  und  $y = z1v$ .

Offensichtlich ist  $(L, \leq)$  eine lineare Ordnung.

- $(L, \leq)$  hat kein größtes Element:

Sei  $x \in L$  beliebig. Dann gilt  $x < x1 \in L$

# $(\mathbb{Q}, \leq)$ ist automatisch

- $(L, \leq)$  hat kein kleinstes Element:

Sei  $x = u1 \in L$  beliebig. Dann gilt  $u01 < u1 = x$

- $(L, \leq)$  ist dicht:

Seien  $x, y \in L$  mit  $x < y$  beliebig.

1. Fall:  $x = u1, y = u1v1$ :

Dann gilt:  $x = u1 < u10^{|\nu|+1}1 < u1v1 = y$

2. Fall:  $x = u0v1, y = u1w$ :

Dann gilt:  $x = u0v1 < u01^{|\nu|+2} < u1w = y$

- $(L, \leq)$  ist automatisch: Übung



Von den folgenden Strukturen kann man zeigen, dass sie nicht automatisch sind:

- $(\mathbb{R}, +)$  (denn jede automatische Struktur ist abzählbar)
- jede Struktur mit einer unentscheidbaren Theorie (siehe nächste Folie).

Beispiele hierfür:

- $(\mathbb{N}, +, \cdot)$  (Satz von Gödel)
- $(\Sigma^*, \circ)$  (das freie Monoid über  $\Sigma$ ) falls  $|\Sigma| > 1$  (Quine 1946)
- $(\mathbb{N}, \cdot)$  und  $(\mathbb{N}, |)$
- $(\mathbb{Q}, +)$  (Tsankov 2009)



Unser Hauptresultat über automatische Strukturen lautet:

Satz (Khossainov, Nerode 1994)

Für jede automatisch präsentierbare Struktur  $\mathcal{A}$  ist  $\text{Th}(\mathcal{A})$  entscheidbar.

Korollar (Presburger 1929)

$\text{Th}(\mathbb{N}, +)$  ist entscheidbar.

Korollar

$\text{Th}(\mathbb{Q}, \leq)$  ist entscheidbar.

## Beweis des Satzes von Khossainov und Nerode:

Sei  $\mathcal{A} = (L, R_1, \dots, R_m)$  eine automatische Struktur mit  $L \subseteq \Sigma^*$ .

Für jede Formel  $F$  mit höchstens den freien Variablen  $x_1, \dots, x_n$  werden wir durch Induktion einen synchronen  $n$ -Bandautomaten  $B_F$  konstruieren, so dass gilt:

$$K(B_F) = \{(w_1, \dots, w_n) \in L^n \mid \mathcal{A}_{[x_1/w_1] \dots [x_n/w_n]} \models F\}.$$

**Fall 1:**  $F = R_i(x_{i_1}, \dots, x_{i_k})$ , wobei  $1 \leq i_1, \dots, i_k \leq n$ :

Definiere den Homomorphismus  $f : (\Sigma_{\#}^n)^* \rightarrow (\Sigma_{\#}^k)^*$  wie folgt, wobei  $a_1, \dots, a_n \in \Sigma_{\#}$ :

$$f(a_1, \dots, a_n) = \begin{cases} \varepsilon & \text{falls } a_{i_1} = \dots = a_{i_k} = \# \\ (a_{i_1}, \dots, a_{i_k}) & \text{sonst} \end{cases}$$

Beachte:  $f(w_1 \otimes \dots \otimes w_n) = w_{i_1} \otimes \dots \otimes w_{i_k}$  für alle  $w_1, \dots, w_n \in \Sigma^*$ .

Sei  $B_i$  der synchrone  $k$ -Bandautomat für  $R_i$ . Aus  $B_i$  konstruieren wir nun einen  $n$ -Bandautomaten  $B_F$  mit

$$L(B_F) = f^{-1}(L(B_i)) \cap \{w_1 \otimes \dots \otimes w_n \mid w_1, \dots, w_n \in L\}.$$

Beachte: Die regulären Sprachen sind unter inversen Homomorphismen abgeschlossen.

**Fall 2:**  $F = (x_i = x_j)$ , wobei  $1 \leq i, j \leq n$ :

Analog zu Fall 1, da  $\{(v, v) \mid v \in L\}$  synchron rational ist.

**Fall 3:**  $F = \neg G$ :

IH  $\rightsquigarrow$   $n$ -Bandautomat  $B_G$  für  $G$

Wir wählen dann  $B_F$  so, dass gilt:

$$L(B_F) = \{w_1 \otimes \cdots \otimes w_n \mid w_1, \dots, w_n \in L\} \setminus L(B_G)$$

**Fall 4:**  $F = G \vee H$ , wobei  $F$  höchstens freie Variablen  $x_1, \dots, x_n$  enthält:

IH  $\rightsquigarrow$   $n$ -Bandautomaten  $B_G, B_H$  für  $G$  und  $H$

Wir wählen dann  $B_F$  so, dass gilt:

$$L(B_F) = L(B_G) \cup L(B_H)$$

**Fall 5:**  $F = \exists x_{n+1} : G(x_1, \dots, x_n, x_{n+1})$ :

IH  $\rightsquigarrow$   $(n+1)$ -Bandautomat  $B_G$  für  $G$

Definiere den Homomorphismus  $f : (\Sigma_{\#}^{n+1})^* \rightarrow (\Sigma_{\#}^n)^*$  wie folgt, wobei  $a_1, \dots, a_n, a_{n+1} \in \Sigma_{\#}$ :

$$f(a_1, \dots, a_n, a_{n+1}) = \begin{cases} \varepsilon & \text{falls } a_1 = \dots = a_n = \# \\ (a_1, \dots, a_n) & \text{sonst} \end{cases}$$

Beachte:  $f(w_1 \otimes \dots \otimes w_n \otimes w_{n+1}) = w_1 \otimes \dots \otimes w_n$  für alle  $w_1, \dots, w_{n+1} \in \Sigma^*$ .

Dann wählen wir für  $B_F$  einen  $n$ -Bandautomaten mit  $L(B_F) = f(L(B_G))$ .

Beachte: Die regulären Sprachen sind unter Homomorphismen abgeschlossen.

Dies beendet die Konstruktion von  $B_F$ .

Sei nun  $F$  eine Aussage (keine freien Variablen).

O.B.d.A. können wir davon ausgehen, dass  $F$  von der Form  $F = \exists x G(x)$  ist (wir können immer einen Dummy- $\exists$ -Quantor hinzufügen).

Dann gilt:  $\mathcal{A} \models F \iff L(B_G) \neq \emptyset$ .

Letzteres ist entscheidbar, da Leerheit der von einem endlichen Automaten akzeptierten Sprache entscheidbar ist. □

## Bemerkungen zur Komplexität:

Unser Algorithmus, der  $F \in \text{Th}(\mathcal{A})$  entscheidet, ist nicht sehr effizient.

**Grund:** Für jede Negation  $\neg$  in  $F$  müssen wir einen Automaten komplementieren. Dies verursacht einen exponentiellen Blow-Up in der Automatengröße.

Die Laufzeit unseres Algorithmus ist deshalb in etwa  $f_{|F|}(O(1))$ , wobei  $f_0(n) = n$  und  $f_{i+1}(n) = 2^{f_i(n)}$  für  $i \geq 0$ .

Dies ist jedoch auch nicht vermeidbar:

Sei  $T_2 = (\{0, 1\}^*, S_0, S_1, \leq)$  wobei:

- $S_0 = \{(w, w0) \mid w \in \{0, 1\}^*\}$
- $S_1 = \{(w, w1) \mid w \in \{0, 1\}^*\}$
- $\leq = \{(w, wu) \mid w, u \in \{0, 1\}^*\}$

Beachte:  $T_2$  ist eine automatische Struktur.

## Meyer 1974

Es gibt kein  $i \in \mathbb{N}$  und einen Algorithmus, der  $\text{Th}(T_2)$  korrekt entscheidet und dessen Laufzeit durch  $f_i(n)$  (bei einer Eingabeformel der Länge  $n$ ) beschränkt ist.

Man sagt auch: Es existiert kein **elementarer Algorithmus** für  $\text{Th}(T_2)$ .

Es gibt jedoch viele Spezialfälle von automatischen Strukturen, für die ein elementarer Algorithmus zur Entscheidung der Theorie existiert.

Hier sind zwei Beispiele:

## Oppen 1978

Es existiert ein Algorithmus, der  $\text{Th}(\mathbb{N}, +)$  in Zeit  $2^{2^{O(n)}}$  entscheidet.



## Satz (Tarski 1948)

$\text{Th}(\mathbb{R}, +, \cdot)$  ist entscheidbar.

### Beweis:

Zunächst betrachten wir anstatt  $\text{Th}(\mathbb{R}, +, \cdot)$  die Theorie  $\text{Th}(\mathbb{R}, +, \cdot, <, 0, 1, -1)$ .

Wir schreiben im Folgenden  $\mathbb{R}$  für  $(\mathbb{R}, +, \cdot, <, 0, 1, -1)$ .

**Quantorenelimination:** Wir konstruieren zu einer gegebenen prädikatenlogischen Formel  $F$  eine **quantorenfreie** Formel  $F'$  mit:  
 $F \in \text{Th}(\mathbb{R}) \iff F' \in \text{Th}(\mathbb{R})$ .

Es genügt, dies für eine Formel  $F = \exists x G$  zu zeigen, wobei  $G$  quantorenfrei ist.

Seien  $y_1, \dots, y_n$  die freien Variablen von  $F$ .

Außerdem können wir annehmen, dass  $G$  folgende Gestalt hat:

$$G = s(x, y_1, \dots, y_n) = 0 \wedge \bigwedge_{i=1}^m t_i(x, y_1, \dots, y_n) > 0,$$

wobei  $s, t_1, \dots, t_m \in \mathbb{Z}[x, y_1, \dots, y_n]$ .

Beachte hierzu:

- $\exists x(G_1 \vee G_2) \equiv (\exists x G_1) \vee (\exists x G_2)$
- $s_1 = s_2 \iff s_1 - s_2 = 0$
- $s_1 < s_2 \iff s_2 - s_1 > 0$
- $\neg(s = 0) \iff (s > 0 \vee -s > 0)$
- $\neg(s > 0) \iff (s = 0 \vee -s > 0)$
- $\bigwedge_{i=1}^k s_i = 0 \iff \sum_{i=1}^k s_i^2 = 0$

# Entscheidbarkeit der reellen Arithmetik

Wir unterscheiden nun 3 Fälle:

- Fall 1:  $x$  kommt in  $s$  vor und  $m = 1$
- Fall 2:  $x$  kommt in  $s$  vor und  $m > 1$
- Fall 3:  $x$  kommt in  $s$  nicht vor.

**Fall 1:**  $G = (s = 0 \wedge t > 0)$  mit  $s = p_m x^m + \dots + p_1 x + p_0$ ,  $m \geq 1$ ,  
 $p_i \in \mathbb{Z}[y_1, \dots, y_n]$

**Notation:** Für  $k \geq 0$  sei  $(\#x : G) = k$  eine neue Formel mit:  
Für alle  $a_1, \dots, a_n \in \mathbb{R}$  gilt  $\mathbb{R}_{[y_1/a_1, \dots, y_n/a_n]} \models (\#x : G) = k$  g.d.w.

$$|\{a \in \mathbb{R} \mid \mathbb{R}_{[x/a, y_1/a_1, \dots, y_n/a_n]} \models G\}| = k.$$

**Beachte:**  $\exists x G$  ist in  $\mathbb{R}$  äquivalent zu

$$(\#x : G) = 1 \vee (\#x : G) = 2 \vee \dots \vee (\#x : G) = m$$

**Neues Ziel:** Finde eine quantorenfreie Formel, welche in  $\mathbb{R}$  äquivalent ist zu  $(\#x : G) = k$ .

Unser Hilfsmittel sind sogenannte **Sturmfolgen**.

Für  $\bar{a} = (a_1, \dots, a_n) \in (\mathbb{R} \setminus \{0\})^n$  sei

$$\text{Var}(\bar{a}) = |\{i < n \mid a_i a_{i+1} < 0\}|$$

(Anzahl der Vorzeichenwechsel).

Für  $\bar{a} \in \mathbb{R}^n$  sei  $\text{Var}(\bar{a}) = \text{Var}(\bar{b})$ , wobei  $\bar{b}$  aus  $\bar{a}$  durch Löschen aller Nullen entsteht.

Für  $\bar{f} = (f_1, \dots, f_n) \in (\mathbb{R}[x])^n$  und  $a \in \mathbb{R}$  sei

$$\text{Var}_a(\bar{f}) = \text{Var}(f_1(a), \dots, f_n(a)).$$

# Entscheidbarkeit der reellen Arithmetik

Seien  $f, g \in \mathbb{R}[x]$ . Definiere die Polynome  $h_0(x), \dots, h_n(x)$  eindeutig wie folgt (**Euklidischer Algorithmus**):

$$\begin{aligned}h_0(x) &= f(x), & h_1(x) &= g(x) \\h_0(x) &= q_1(x)h_1(x) - h_2(x) & \deg(h_2) &< \deg(h_1) \\h_1(x) &= q_2(x)h_2(x) - h_3(x) & \deg(h_3) &< \deg(h_2) \\&\vdots \\h_{n-1}(x) &= q_n(x)h_n(x)\end{aligned}$$

Dann gilt:

- $h_n(x) = \text{ggT}(f, g)$
- Für alle  $0 \leq i \leq n$  ist das Polynom  $h_n(x)$  ein Teiler von  $h_i(x)$ .

Dann ist  $[f, g] = (h_0(x), h_1(x), \dots, h_n(x))$  die **Sturmfolge** von  $f$  und  $g$ .

# Entscheidbarkeit der reellen Arithmetik

Die **gekürzte Sturmfolge** von  $f$  und  $g$  ist

$$\left( \frac{h_0(x)}{h_n(x)}, \frac{h_1(x)}{h_n(x)}, \dots, \frac{h_{n-1}(x)}{h_n(x)}, 1 \right).$$

Für eine quantorenfreie Formel  $H$  mit der einzigen freien Variablen  $x$  und  $a, b \in \mathbb{R}$  mit  $a < b$  sei

$$(\#x : H)_a^b = |\{c \in (a, b) \mid \mathbb{R}_{[x/c]} \models H\}|.$$

Mit  $f'(x)$  bezeichnen wir die **formale Ableitung** des Polynoms  $f(x) \in \mathbb{R}[x]$ .

## Satz von Sturm und Tarski

Seien  $f, g \in \mathbb{R}[x]$ ,  $\text{ggT}(f, g) = \text{ggT}(f, f') = 1$ ,  $a, b \in \mathbb{R}$ ,  $a < b$ ,  $f(a) \neq 0 \neq f(b)$  (insbesondere  $f \neq 0$ ). Dann gilt

$$(\#x : f(x) = 0 \wedge g(x) > 0)_a^b - (\#x : f(x) = 0 \wedge g(x) < 0)_a^b = \text{Var}_a([f, f'g]) - \text{Var}_b([f, f'g]).$$

# Entscheidbarkeit der reellen Arithmetik

Für den Beweis des Satzes von Sturm und Tarski benötigen wir zwei Lemmata.

## Lemma A

Seien  $f, g \in \mathbb{R}[x]$ ,  $a, b \in \mathbb{R}$ ,  $a < b$ , und  $\forall c \in [a, b] : f(c) \neq 0$ . Dann gilt  $\text{Var}_a([f, g]) = \text{Var}_b([f, g])$ .

**Beweis von Lemma A:** Sei

$$[f, g] = S = (h_0, h_1, \dots, h_s)$$

und sei

$$S' = (h'_0, h'_1, \dots, h'_s)$$

die gekürzte Sturmfolge, d.h.  $h'_s = 1$  und  $h'_i = \frac{h_i}{h_s}$ .

Sei  $N = \{c \in [a, b] \mid \exists 0 \leq i \leq s : h'_i(c) = 0\}$ .

Dann ist  $N$  endlich.

Sei  $[a', b'] \subseteq [a, b]$  ein Intervall mit  $|N \cap [a', b']| \leq 1$ .

Es genügt es folgende Behauptung zu zeigen:

$$\text{Var}_{a'}([f, g]) = \text{Var}_{b'}([f, g]).$$

**Fall 1:** Kein  $h'_i$  hat eine Nullstelle in  $[a', b']$ .

Nach dem Zwischenwertsatz gilt  $h'_i(a') \cdot h'_i(b') > 0$  für alle  $0 \leq i \leq s$ .

Also gilt  $\text{Var}_{a'}(S') = \text{Var}_{b'}(S')$ .

Wegen  $h_s(a') \neq 0 \neq h_s(b')$  (denn  $f(a') \neq 0 \neq f(b')$  und  $h_s = \text{ggT}(f, g)$  ist Teiler von  $f$ ) folgt

$$\text{Var}_{a'}(S) = \text{Var}_{a'}(S') = \text{Var}_{b'}(S') = \text{Var}_{b'}(S).$$



# Entscheidbarkeit der reellen Arithmetik

**Fall 2:** Mindestens ein  $h'_i$  hat eine Nullstelle  $c \in [a', b']$ , d.h.  
 $N \cap [a', b'] = \{c\}$ .

Wegen  $h'_s = 1$  und der Annahme, dass  $f = h_0$  keine Nullstelle in  $[a, b]$  hat (damit hat auch  $h'_0$  keine Nullstelle in  $[a, b]$ ) gilt  $1 \leq i \leq s - 1$ .

Es gilt  $h'_{i-1}(c) = q_i(c)h'_i(c) - h'_{i+1}(c) = -h'_{i+1}(c)$ .

Würde  $h'_{i+1}(c) = 0 = h'_i(c)$  gelten, so wäre  $h'_j(c) = 0$  für alle  $j \geq i$ , was  $h'_s = 1$  widerspricht.

Also gilt  $h'_{i+1}(c) \neq 0$  und damit

$$h'_{i-1}(c)h'_{i+1}(c) = -(h'_{i+1}(c))^2 < 0,$$

d.h.  $h'_{i-1}(c)$  und  $h'_{i+1}(c)$  haben verschiedene Vorzeichen.

Da  $h'_{i-1}$  und  $h'_{i+1}$  keine Nullstelle in  $[a', b']$  haben, gilt

$$h'_{i-1}(a')h'_{i+1}(a') < 0 \quad \text{und} \quad h'_{i-1}(b')h'_{i+1}(b') < 0.$$

Es folgt:

$$\begin{aligned}\text{Var}_{a'}(S') &= \text{Var}(h'_0(a'), \dots, h'_{i-1}(a'), h'_i(a'), h'_{i+1}(a'), \dots, h'_s(a')) \\ &= \text{Var}(h'_0(a'), \dots, h'_{i-1}(a'), h'_{i+1}(a'), \dots, h'_s(a')) \\ \text{Var}_{b'}(S') &= \text{Var}(h'_0(b'), \dots, h'_{i-1}(b'), h'_i(b'), h'_{i+1}(b'), \dots, h'_s(b')) \\ &= \text{Var}(h'_0(b'), \dots, h'_{i-1}(b'), h'_{i+1}(b'), \dots, h'_s(b'))\end{aligned}$$

Auf diese Weise können wir für alle  $j$  mit  $h'_j(c) = 0$  die Einträge  $h'_j(a')$  und  $h'_j(b')$  eliminieren.

Wir erhalten somit

$$\begin{aligned}\text{Var}_{a'}(S') &= \text{Var}(g_0(a'), \dots, g_t(a')) \\ \text{Var}_{b'}(S') &= \text{Var}(g_0(b'), \dots, g_t(b'))\end{aligned}$$

wobei die Polynome  $g_0, \dots, g_t$  keine Nullstelle in  $[a', b']$  haben.

Also gilt:

$$\begin{aligned}\text{Var}_{a'}(S) = \text{Var}_{a'}(S') &= \text{Var}(g_0(a'), \dots, g_t(a')) \\ &= \text{Var}(g_0(b'), \dots, g_t(b')) = \text{Var}_{b'}(S') = \text{Var}_{b'}(S)\end{aligned}$$



## Lemma B

Seien  $f, g \in \mathbb{R}[x]$ ,  $\text{ggT}(f, g) = \text{ggT}(f, f') = 1$ ,  $a, b, c \in \mathbb{R}$ ,  $a < c < b$ ,  $f(c) = 0$ ,  $\forall d \in [a, b] \setminus \{c\} : f(d) \neq 0$ . Dann gilt

$$\text{Var}_a([f, f'g]) - \text{Var}_b([f, f'g]) = \begin{cases} 1 & \text{falls } g(c) > 0 \\ -1 & \text{falls } g(c) < 0 \end{cases}$$

## Beweis von Lemma B:

Wegen  $\text{ggT}(f, g) = \text{ggT}(f, f') = 1$  haben  $f$  und  $g$  keine gemeinsamen Nullstellen, und  $f$  hat keine mehrfache Nullstelle.

Insbesondere gilt  $g(c) \neq 0$  und es gibt ein Polynom  $h(x)$  mit  $f(x) = (x - c) \cdot h(x)$ ,  $h(c) \neq 0$ .

Also gilt  $f'g = (x - c) \cdot \underbrace{(h^2g + (x - c)hh'g)}_{u(x)}$ .

Sei  $[f, f'g] = (f, f'g, h_2, \dots, h_s)$  mit  $s \geq 1$ .

Gelte  $g(c) > 0$  (der Fall  $g(c) < 0$  kann analog analysiert werden).

Es gilt  $u(c) = (h(c))^2g(c) > 0$  und  $f'g(c) \neq 0$ .

Also gibt es  $a' < b'$  mit  $a \leq a' < c$ ,  $c < b' \leq b$  und  $\forall x \in [a', b'] : u(x) > 0$  und  $f'g(x) \neq 0$ .

Es folgt  $f(a') \cdot f'g(a') < 0$  und  $f(b') \cdot f'g(b') > 0$ .

# Entscheidbarkeit der reellen Arithmetik

Wir erhalten damit im Fall  $s \geq 2$ :

$$\begin{aligned} \text{Var}_a([f, f'g]) &\stackrel{\text{Lemma A}}{=} \text{Var}_{a'}([f, f'g]) \\ &= 1 + \text{Var}_{a'}([f'g, h_2]) \\ &\stackrel{\text{Lemma A}}{=} 1 + \text{Var}_{b'}([f'g, h_2]) \\ &= 1 + \text{Var}_{b'}([f, f'g]) \\ &\stackrel{\text{Lemma A}}{=} 1 + \text{Var}_b([f, f'g]) \end{aligned}$$

Im Fall  $s = 1$  (d.h.  $[f, f'g] = (f, f'g)$ ) gilt

$$\begin{aligned} \text{Var}_a([f, f'g]) &\stackrel{\text{Lemma A}}{=} \text{Var}_{a'}([f, f'g]) \\ &= 1 \\ &= 1 + \text{Var}_{b'}([f, f'g]) \\ &\stackrel{\text{Lemma A}}{=} 1 + \text{Var}_b([f, f'g]) \end{aligned}$$

## Beweis des Satzes von Tarski und Sturm:

Seien  $f, g \in \mathbb{R}[x]$ ,  $\text{ggT}(f, g) = \text{ggT}(f, f') = 1$ ,  $a, b \in \mathbb{R}$ ,  $a < b$ ,  
 $f(a) \neq 0 \neq f(b)$ .

Sei  $N = \{c \in (a, b) \mid f(c) = 0\}$  (endlich).

Falls  $N = \emptyset$  gilt wegen Lemma A:

$$(\#x : f(x) = 0 \wedge g(x) > 0)_a^b - (\#x : f(x) = 0 \wedge g(x) < 0)_a^b = 0 = \\ \text{Var}_a([f, f'g]) - \text{Var}_b([f, f'g]).$$

Sei nun  $N = \{c_1, c_2, \dots, c_n\}$  mit  $n \geq 1$ .

Wähle Punkte  $a = a_0 < c_1 < a_1 < c_2 < a_2 < \dots < a_{n-1} < c_n < a_n = b$ .

Dann gilt wegen Lemma B für alle  $1 \leq i \leq n$ :

$$\text{Var}_{a_{i-1}}([f, f'g]) - \text{Var}_{a_i}([f, f'g]) = \begin{cases} 1 & \text{falls } g(c_i) > 0 \\ -1 & \text{falls } g(c_i) < 0 \end{cases}$$

Aufsummieren ergibt

$$\text{Var}_a([f, f'g]) - \text{Var}_b([f, f'g]) = (\#x : f(x) = 0 \wedge g(x) > 0)_a^b - (\#x : f(x) = 0 \wedge g(x) < 0)_a^b$$

Dies beendet den Beweis des Satzes von Tarski und Sturm. □

## Korollar aus dem Satz von Tarski und Sturm

Seien  $f, g \in \mathbb{R}[x]$ ,  $\text{ggT}(f, g) = \text{ggT}(f, f') = 1$ ,  $a, b \in \mathbb{R}$ ,  $a < b$ ,  $f(a) \neq 0 \neq f(b)$  (insbesondere  $f \neq 0$ ). Dann gilt

$$\begin{aligned} & \#x(f(x) = 0 \wedge g(x) > 0)_a^b \\ &= \frac{1}{2}(\text{Var}_a([f, f'g]) - \text{Var}_b([f, f'g]) + \text{Var}_a([f, f']) - \text{Var}_b([f, f'])). \end{aligned}$$

**Beweis:** Nach dem Satz von Tarski und Sturm gilt:

$$\begin{aligned} & (\#x : f(x) = 0 \wedge g(x) > 0)_a^b - (\#x : f(x) = 0 \wedge g(x) < 0)_a^b \\ &= \text{Var}_a([f, f'g]) - \text{Var}_b([f, f'g]). \end{aligned}$$



sowie (da  $f$  und  $g$  keine gemeinsame Nullstelle haben)

$$\begin{aligned} & (\#x : f(x) = 0 \wedge g(x) > 0)_a^b + (\#x : f(x) = 0 \wedge g(x) < 0)_a^b \\ &= (\#x : f(x) = 0)_a^b = \\ &= (\#x : f(x) = 0 \wedge 1 > 0)_a^b - (\#x : f(x) = 0 \wedge 1 < 0)_a^b \\ &= \text{Var}_a([f, f']) - \text{Var}_b([f, f']). \end{aligned}$$

Addieren der beiden Gleichungen ergibt:

$$\begin{aligned} & 2 \cdot (\#x : f(x) = 0 \wedge g(x) > 0)_a^b \\ &= \text{Var}_a([f, f'g]) - \text{Var}_b([f, f'g]) + \text{Var}_a([f, f']) - \text{Var}_b([f, f']) \end{aligned}$$



Wir brauchen auch noch Cauchys Schranke für die Nullstellen eines Polynoms.

## Lemma (Cauchys Schranke für Nullstellen eines Polynoms)

Sei  $f(x) = a_m x^m + \dots + a_1 x + a_0 \in \mathbb{R}[x]$ ,  $a_m \neq 0$ . Dann liegen alle Nullstellen von  $f$  im Intervall  $(-c, c)$  mit

$$c = 1 + \frac{\max\{|a_0|, \dots, |a_{m-1}|\}}{|a_m|}.$$

### Beweis:

Indem wir  $f(x)$  durch das Polynom  $x^m + \frac{a_{m-1}}{a_m} x^{m-1} + \dots + \frac{a_1}{a_m} x + \frac{a_0}{a_m}$  ersetzen, genügt es, Cauchys Schranke für den Fall  $a_m = 1$  zu zeigen.

# Entscheidbarkeit der reellen Arithmetik

Sei also  $f(x) = x^m + a_{m-1}x^{m-1} + \dots + a_1x + a_0$  und

$$h = \max\{|a_i| \mid 0 \leq i \leq m-1\}.$$

Gelte  $f(\alpha) = \alpha^m + a_{m-1}\alpha^{m-1} + \dots + a_1\alpha + a_0 = 0$ , d.h.

$$\alpha^m = -a_{m-1}\alpha^{m-1} - \dots - a_1\alpha - a_0.$$

Zu zeigen:  $|\alpha| < 1 + h$ .

Wenn  $|\alpha| \leq 1$  gilt, gilt auch  $|\alpha| < 1 + h$  (falls  $h = 0$  muss  $\alpha = 0$  gelten).

Gelte also  $|\alpha| > 1$ .

Es folgt:

$$\begin{aligned} |\alpha|^m &\leq |a_{m-1}| \cdot |\alpha|^{m-1} + \dots + |a_1| \cdot |\alpha| + |a_0| \\ &\leq h \cdot (|\alpha|^{m-1} + \dots + |\alpha| + 1) \\ &= h \cdot \frac{|\alpha|^m - 1}{|\alpha| - 1} \end{aligned}$$

Da  $|\alpha| > 1$  gilt, folgt:

$$|\alpha| - 1 \leq h \cdot \frac{|\alpha|^m - 1}{|\alpha|^m} < h$$



Wir kommen nun zurück zu Fall 1.

Erinnerung: Wir müssen eine quantorenfreie arithmetische Formel für  $(\#x : s = 0 \wedge t > 0) = k$  finden, wobei  $s = p_m x^m + \dots + p_1 x + p_0$ ,  $m \geq 1$ ,  $p_i \in \mathbb{Z}[y_1, \dots, y_n]$

Wir können uns auf den Fall beschränken, dass  $s = y_m x^m + \dots + y_1 x + y_0$  und  $t = z_l x^l + \dots + z_1 x + z_0$  gilt.

Wir müssen dann für  $(\#x : s = 0 \wedge t > 0) = k$  eine quantorenfreie arithmetische Formel in den freien Variablen  $y_0, \dots, y_m, z_0, \dots, z_l$  finden.

Wegen Cauchys Schranke genügt es für  $(\#x : s = 0 \wedge t > 0)_y^z = k$  eine quantorenfreie arithmetische Formel in den freien Variablen  $y, z, y_0, \dots, y_m, z_0, \dots, z_l$  zu finden.

# Entscheidbarkeit der reellen Arithmetik

In dieser Formel können wir dann  $y$  durch  $-\frac{|y_m| + \max\{|y_0|, \dots, |y_{m-1}|\}}{|y_m|}$  und  $z$  durch  $\frac{|y_m| + \max\{|y_0|, \dots, |y_{m-1}|\}}{|y_m|}$  ersetzen.

Anwendungen von  $|\cdot|$  und  $\max$  können durch eine große Fallunterscheidung ersetzt werden.

Ebenso muss der Fall  $y_m = 0$  durch eine Fallunterscheidung abgefangen werden.

Anwendungen von  $\frac{\cdot}{|y_m|}$  (im Fall  $y_m \neq 0$ ) können durch Multiplizieren mit genügend großen Potenzen von  $y_m$  eliminiert werden.

Wir konstruieren eine quantorenfreie arithmetische Formel in den freien Variablen  $y, z, y_0, \dots, y_m, z_0, \dots, z_l$  für  $(\#x : s = 0 \wedge t > 0)_y^z = k$  mittels des Satzes von Sturm und Tarski (Korollar hiervon).

# Entscheidbarkeit der reellen Arithmetik

Problem: Wie stellen wir die Vorbedingung  $\text{ggT}(s, t) = \text{ggT}(s, s') = 1$  sicher?

Beachte hierzu:

- $(\#x : s = 0 \wedge t > 0)_y^z = k$  gdw.  $(\#x : s/\text{ggT}(s, t) = 0 \wedge t > 0)_y^z = k$ .
- $(\#x : s = 0 \wedge t > 0)_y^z = k$  gdw.  $(\#x : s/\text{ggT}(s, s') = 0 \wedge t > 0)_y^z = k$ .

Wir können daher  $s$  zunächst durch  $r = s/\text{ggT}(s, t)$  ersetzen, und dann  $r$  durch  $r/\text{ggT}(r, r')$  ersetzen.

Betrachten wir die Berechnung von  $s/\text{ggT}(s, t)$  genauer.

Die Koeffizienten der  $x$ -Polynome  $s$  und  $t$  sind die Variablen  $y_i, z_j$ .

Hierzu lassen daher den Euklidischen Algorithmus symbolisch mit  $s = y_m x^m + \dots + y_1 x + y_0$  und  $t = z_l x^l + \dots + z_1 x + z_0$  laufen.

Dabei werden die Koeffizienten  $y_m, \dots, y_1, y_0$  durch  $z_l$  geteilt. Wir müssen daher den Fall  $z_l = 0$  abfangen.

# Entscheidbarkeit der reellen Arithmetik

**Beispiel:**  $m = 2$ ,  $l = 1$ , d.h.

$$s(x) = y_2x^2 + y_1x + y_0 \text{ und } t(x) = z_1x + z_0$$

Dann gilt  $(\exists x : s = 0 \wedge t > 0)_y^z = k$  g.d.w.

$$(z_1 \neq 0 \wedge (\exists x : z_1^2 \cdot s = 0 \wedge t > 0)_y^z = k) \vee \\ (z_1 = 0 \wedge (\exists x : s = 0 \wedge z_0 > 0)_y^z = k).$$

Division mit Rest:

$$\begin{array}{r} (y_2z_1^2x^2 + y_1z_1^2x + y_0z_1^2) : (z_1x + z_0) = y_2z_1x + (y_1z_1 - y_2z_0) \\ - \underline{(y_2z_1^2x^2 + y_2z_1z_0x)} \\ ((y_1z_1^2 - y_2z_1z_0)x + y_0z_1^2) \\ - \underline{((y_1z_1^2 - y_2z_1z_0)x + (y_1z_1z_0 - y_2z_0^2))} \\ y_0z_1^2 - y_1z_1z_0 + y_2z_0^2 \text{ (Rest)} \end{array}$$



# Entscheidbarkeit der reellen Arithmetik

Also:

- Wenn  $z_1 \neq 0 \neq y_0 z_1^2 - y_1 z_1 z_0 + y_2 z_0^2$ , dann gilt  $\text{ggT}(z_1^2 s, t) = 1$  und wir machen mit der symbolischen Berechnung von  $\text{ggT}(z_1^2 s, z_1^2 s')$  weiter.
- Wenn  $z_1 \neq 0 = y_0 z_1^2 - y_1 z_1 z_0 + y_2 z_0^2$ , dann gilt  $\text{ggT}(z_1^2 s, t) = t = (z_1 x + z_0)$  und  $\frac{z_1^2 s}{t} = y_2 z_1 x + (y_1 z_1 - y_2 z_0)$ .  
Ausserdem gilt  $(z_1 \neq 0 \wedge (\#x : z_1^2 \cdot s = 0 \wedge t > 0))_y^z = k$  g.d.w.  
 $(z_1 \neq 0 \wedge (\#x : y_2 z_1 x + y_1 z_1 - y_2 z_0 = 0 \wedge t > 0))_y^z = k$ .

Es gilt noch nicht unbedingt  $\text{ggT}(y_2 z_1 x + y_1 z_1 - y_2 z_0, t) = 1$ , aber der  $x$ -Grad von  $y_2 z_1 x + y_1 z_1 - y_2 z_0$  ist kleiner als der  $x$ -Grad von  $z_1^1 s$ .

Wir können also wie oben weiter machen.

Auf die gleiche Weise kann die Voraussetzung  $\text{ggT}(s, s') = 1$  sichergestellt werden.

# Entscheidbarkeit der reellen Arithmetik

Unter der Voraussetzung  $\text{ggT}(s, t) = \text{ggT}(s, s') = 1$  (und  $s(y) \neq 0 \neq s(z)$ ) ist  $(\exists x : s = 0 \wedge t > 0)_y = k$  äquivalent zu

$$\text{Var}_y([s, s't]) - \text{Var}_z([s, s't]) + \text{Var}_y([s, s']) - \text{Var}_z([s, s']) = 2k$$

Dies kann als eine Boolesche Kombination von Aussagen der Gestalt  $\text{Var}_y([s, s't]) = i_1$ ,  $\text{Var}_z([s, s't]) = i_2$ ,  $\text{Var}_y([s, s']) = i_3$  und  $\text{Var}_z([s, s']) = i_4$  geschrieben werden.

Eine Aussage  $\text{Var}_y([s, s't]) = i$  (analog für die anderen Polynome) kann schließlich durch eine quantorenfreie Formel ausgedrückt werden.

Hierzu lassen wir wieder symbolisch den Euklidischen Algorithmus für  $s$  und  $s't$  laufen und berechnen so die Sturmfolge symbolisch.

Dies beendet die Behandlung von Fall 1.

# Entscheidbarkeit der reellen Arithmetik

**Fall 2:**  $G = (s = 0 \wedge \bigwedge_{i=1}^m t_i > 0)$ ,  $m \geq 1$ , und  $x$  kommt in  $s$  vor.

Induktion über  $m$ :

IA:  $m = 1$ . Siehe Fall 1.

IS:  $m \geq 2$ .

Sei  $G' = (s = 0 \wedge \bigwedge_{i=1}^{m-2} t_i > 0)$ .

$$\begin{aligned} & \#x(G' \wedge t_{m-1} > 0 \wedge t_m > 0) + \\ & \#x(G' \wedge t_{m-1} > 0 \wedge t_m < 0) = \#x(G' \wedge t_{m-1} t_m^2 > 0) \end{aligned} \quad (1)$$

$$\begin{aligned} & \#x(G' \wedge t_{m-1} > 0 \wedge t_m > 0) + \\ & \#x(G' \wedge t_{m-1} < 0 \wedge t_m > 0) = \#x(G' \wedge t_{m-1}^2 t_m > 0) \end{aligned} \quad (2)$$

$$\begin{aligned} & \#x(G' \wedge t_{m-1} > 0 \wedge t_m < 0) + \\ & \#x(G' \wedge t_{m-1} < 0 \wedge t_m > 0) = \#x(G' \wedge t_{m-1} t_m < 0) \end{aligned} \quad (3)$$

(1) + (2) - (3) ergibt:

$$\begin{aligned}2 \cdot \#xG &= 2 \cdot \#x \cdot (G' \wedge t_{m-1} > 0 \wedge t_m > 0) \\ &= \#x(G' \wedge t_{m-1}t_m^2 > 0) + \\ &\quad \#x(G' \wedge t_{m-1}^2t_m > 0) - \\ &\quad \#x(G' \wedge -t_{m-1}t_m > 0)\end{aligned}$$

**Fall 3:**  $G = (s = 0 \wedge \bigwedge_{i=1}^m t_i > 0)$ ,  $m \geq 1$ , und  $x$  kommt in  $s$  nicht vor.

Dann ist  $\exists x G$  äquivalent zu

$$s = 0 \wedge \exists x \bigwedge_{i=1}^m t_i > 0.$$

Sei  $H = \bigwedge_{i=1}^m t_i > 0$ .

Sei  $t = t_1 t_2 \cdots t_m$ .

Behauptung:  $\exists x H$  ist äquivalent in  $\mathbb{R}$  zu

$$\exists x_0 \forall x \leq x_0 : H \vee \exists x_0 \forall x \geq x_0 : H \vee \exists x (t'(x) = 0 \wedge H). \quad (4)$$

Die Implikation (4)  $\Rightarrow \exists x H$  ist klar.

Gelte nun  $\mathbb{R} \models \exists x H$ .

Hieraus folgt:

$$\mathbb{R} \models \exists x_0 \forall x \leq x_0 : H \vee \exists x_0 \forall x \geq x_0 : H \vee \\ \exists x_1 \exists x \exists x_2 (x_1 < x < x_2 \wedge \neg H[x/x_1] \wedge H \wedge \neg H[x/x_2])$$

# Entscheidbarkeit der reellen Arithmetik

Angenommen es gilt

$$\mathbb{R} \models \exists x_1 \exists x \exists x_2 (x_1 < x < x_2 \wedge \neg H[x/x_1] \wedge H \wedge \neg H[x/x_2])$$

Dann gibt es  $x'_1 < x'_2$  mit  $t(x'_1) = 0 = t(x'_2)$  und  $t_i(y) > 0$  für alle  $y \in (x'_1, x'_2)$  und  $1 \leq i \leq m$  (insbesondere  $t(y) > 0$  für alle  $y \in (x'_1, x'_2)$ ).

Aus dem Satz von Rolle folgt, dass ein  $x$  existiert mit  $t'(x) = 0$  und  $t_i(x) > 0$  für alle  $1 \leq i \leq m$ , d.h.

$$\mathbb{R} \models \exists x_0 \forall x \leq x_0 : H \vee \exists x_0 \forall x \geq x_0 : H \vee \exists x (t'(x) = 0 \wedge H).$$

Dies zeigt die Behauptung.

Es genügt also, eine quantorenfreie Formel für

$$\exists x_0 \forall x \leq x_0 : H \vee \exists x_0 \forall x \geq x_0 : H \vee \exists x (t'(x) = 0 \wedge H)$$

anzugeben.

# Entscheidbarkeit der reellen Arithmetik

Für die Formeln  $\exists x_0 \forall x \leq x_0 H$  und  $\exists x_0 \forall x \geq x_0 H$  kann man leicht quantorenfreie Formeln angeben.

Beachte hierzu: Für ein Polynom  $a_n x^n + \dots + a_1 x + a_0$  mit  $a_n \neq 0$  gilt  $\exists x_0 \forall x \leq x_0 (a_n x^n + \dots + a_1 x + a_0 > 0)$  genau dann, wenn einer der beiden folgenden Fälle gilt:

- $n$  gerade und  $a_n > 0$
- $n$  ungerade und  $a_n < 0$

Die Formel  $\exists x (t'(x) = 0 \wedge H)$  kann über Fall 1 bzw. 2 behandelt werden, falls  $x$  in  $t'(x)$  vorkommt.

Beachte: Kommt  $x$  in  $t'(x)$  nicht vor, so hat  $t(x) = t_1(x) \cdot \dots \cdot t_m(x)$  einen  $x$ -Grad von höchstens 1.

Dies bedeutet, dass  $x$  in höchstens einem  $t_i$  vorkommt; sei dies o.B.d.A.  $t_1$ .

Ausserdem hat  $t_1(x)$  einen  $x$ -Grad von höchstens 1, d.h.

$t_1 = p_1(y_1, \dots, y_n) \cdot x + p_0(y_1, \dots, y_n)$ ,  $t_i = p_i(y_1, \dots, y_n)$  für  $2 \leq i \leq m$ .

Damit ist  $\exists x (t'(x) = 0 \wedge H) = \exists x (t'(x) = 0 \wedge \bigwedge_{i=1}^m t_i > 0)$  äquivalent zu der quantorenfreien Formel

$$p_1(y_1, \dots, y_n) = 0 \wedge p_0(y_1, \dots, y_n) > 0 \wedge \bigwedge_{i=2}^m p_i(y_1, \dots, y_n) > 0$$

Dies beendet den Beweis des Satzes von Tarski. □



# Entscheidbarkeit der reellen Arithmetik

Aus dem Satz von Tarski folgt sofort, dass es keine arithmetische Formel  $\varphi(x)$  mit der einzigen freien Variablen  $x$  gibt, so dass für alle  $r \in \mathbb{R}$  gilt:

$$(\mathbb{R}, +, \cdot)_{[x/r]} \models \varphi(x) \Leftrightarrow r \in \mathbb{N}$$

Denn gäbe es solch ein Formel  $\varphi(x)$  so würde mit Gödels Satz ( $\text{Th}(\mathbb{N}, +, \cdot)$  unentscheidbar) sofort die Unentscheidbarkeit von  $\text{Th}(\mathbb{R}, +, \cdot)$  folgen.

Erstaunlicherweise hat Julia Robinson 1949 solch ein Formel für  $\mathbb{Q}$  anstatt  $\mathbb{R}$  angegeben.

## Satz (Robinson 1949)

Es existiert eine arithmetische Formel  $\varphi(x)$  mit der einzigen freien Variablen  $x$  gibt, so dass für alle rationalen Zahlen  $r \in \mathbb{Q}$  gilt:

$$(\mathbb{Q}, +, \cdot)_{[x/r]} \models \varphi(x) \Leftrightarrow r \in \mathbb{N}$$

Konsequenz:  $\text{Th}(\mathbb{Q}, +, \cdot)$  ist unentscheidbar.

**Monadische Logik 2. Stufe** (kurz **MSO** für **monadic second order**) ist eine Erweiterung der Prädikatenlogik (welche auch als Logik 1. Stufe bezeichnet wird), bei der über Teilmengen des Universums quantifiziert werden darf.

Wir fixieren hierzu zwei Mengen von Variablen:

- Variablen 1. Stufe:  $\text{Var}_1 = \{x_1, x_2, x_3, \dots\}$
- Variablen 2. Stufe:  $\text{Var}_2 = \{X_1, X_2, X_3, \dots\}$

Es gelte  $\text{Var}_1 \cap \text{Var}_2 = \emptyset$ .

Variablen aus  $\text{Var}_1$  bezeichnen wir im folgenden mit  $x, y, z, x', x_0, \dots$ , während wir Variablen aus  $\text{Var}_2$  mit  $X, Y, Z, X', X_0, \dots$  bezeichnen.

Wie in der Prädikatenlogik haben wir Prädikatensymbole  $P_i^k$  ( $k$ -stellig) und Funktionssymbole  $f_i^k$  ( $k$ -stellig).

Terme sind dann wieder genau wie in der Prädikatenlogik definiert.

Die Menge MSO aller **MSO-Formeln** ist die kleinste Menge mit:

- Wenn  $t_1, t_2$  Terme sind und  $X \in \text{Var}_2$ , dann sind  $(t_1 = t_2), (t_1 \in X) \in \text{MSO}$ .
- Wenn  $t_1, t_2, \dots, t_k$  Terme sind, und  $P$  ein  $k$ -stelliges Prädikatensymbol ist, dann ist  $P(t_1, \dots, t_k) \in \text{MSO}$ .
- Wenn  $F, G \in \text{MSO}$ , dann auch  $\neg F, F \wedge G, F \vee G \in \text{MSO}$ .
- Wenn  $F \in \text{MSO}$  und  $x \in \text{Var}_1, X \in \text{Var}_2$  dann  $\exists xF, \exists XF, \forall xF, \forall XF \in \text{MSO}$ .

Die Menge  $\text{free}(F) \subseteq \text{Var}_1 \cup \text{Var}_2$  aller in  $F \in \text{MSO}$  **freien Variablen** ist definiert wie üblich.

Für  $F \in \text{MSO}$  schreiben wir auch  $F(x_1, \dots, x_n, X_1, \dots, X_m)$  um  $\text{free}(F) = \{x_1, \dots, x_n, X_1, \dots, X_m\}$  auszudrücken.

Ein Formel  $F \in \text{MSO}$  ist ein **MSO-Satz**, falls  $\text{free}(F) = \emptyset$  gilt.

## Monadische Logik 2. Stufe

Eine **Struktur** ist nun ein Paar  $\mathcal{A} = (U_{\mathcal{A}}, I_{\mathcal{A}})$ , wobei  $U_{\mathcal{A}}$  eine nicht-leere Menge (das Universum) ist und  $I_{\mathcal{A}}$  eine partiell definierte Abbildung, die

- jedem  $k$ -stelligen Prädikatensymbol  $P$  aus dem Definitionsbereich von  $I_{\mathcal{A}}$  eine  $k$ -stellige Relation  $I_{\mathcal{A}}(P) \subseteq U_{\mathcal{A}}^k$  zuordnet,
- jedem  $k$ -stelligen Funktionssymbol  $f$  aus dem Definitionsbereich von  $I_{\mathcal{A}}$  eine  $k$ -stellige Funktion  $I_{\mathcal{A}}(f) : U_{\mathcal{A}}^k \rightarrow U_{\mathcal{A}}$  zuordnet,
- jeder Variablen  $x \in \text{Var}_1$  aus dem Definitionsbereich von  $I_{\mathcal{A}}$  ein Element  $I_{\mathcal{A}}(x) \in U_{\mathcal{A}}$  zuordnet, und
- jeder Variablen  $X \in \text{Var}_2$  aus dem Definitionsbereich von  $I_{\mathcal{A}}$  eine Teilmenge  $I_{\mathcal{A}}(X) \subseteq U_{\mathcal{A}}$  zuordnet.

Die Struktur  $\mathcal{A}$  heißt passend zur Formel  $F \in \text{MSO}$ , falls  $I_{\mathcal{A}}$  auf allen in  $F$  vorkommenden Prädikatsymbolen, Funktionssymbolen und freien Variablen definiert ist.

## Monadische Logik 2. Stufe

Wenn  $\mathcal{A}$  passend zu  $F$  ist, schreiben wir  $\mathcal{A} \models F$  genau dann, wenn einer der folgenden Fälle gilt:

- $F = (t_1 = t_2)$  und  $\mathcal{A}(t_1) = \mathcal{A}(t_2)$
- $F = (t \in X)$  und  $\mathcal{A}(t) \in I_{\mathcal{A}}(X)$
- $F = P(t_1, \dots, t_k)$  und  $(\mathcal{A}(t_1), \dots, \mathcal{A}(t_k)) \in I_{\mathcal{A}}(P)$
- $F = \neg G$  und  $\mathcal{A} \not\models G$
- $F = G \wedge H$  und  $(\mathcal{A} \models G \text{ und } \mathcal{A} \models H)$
- $F = G \vee H$  und  $(\mathcal{A} \models G \text{ oder } \mathcal{A} \models H)$
- $F = \exists x G$  und es gibt ein  $a \in U_{\mathcal{A}}$  mit  $\mathcal{A}_{[x/a]} \models G$
- $F = \forall x G$  und für alle  $a \in U_{\mathcal{A}}$  gilt  $\mathcal{A}_{[x/a]} \models G$
- $F = \exists X G$  und es gibt ein  $B \subseteq U_{\mathcal{A}}$  mit  $\mathcal{A}_{[X/B]} \models G$
- $F = \forall X G$  und für alle  $B \subseteq U_{\mathcal{A}}$  gilt  $\mathcal{A}_{[X/B]} \models G$ .

## Konventionen:

- Im folgenden bezeichnen wir die Interpretation  $I_{\mathcal{A}}(P)$  eines Symbols  $P$  ebenfalls mit dem Symbol  $P$ .
- Eine Struktur  $\mathcal{A} = (U_{\mathcal{A}}, I_{\mathcal{A}})$  mit  $\text{dom}(I_{\mathcal{A}}) = \{P_1, \dots, P_n, f_1, \dots, f_m\}$  schreiben wir auch als  $(U_{\mathcal{A}}, I_{\mathcal{A}}(P_1), \dots, I_{\mathcal{A}}(P_n), I_{\mathcal{A}}(f_1), \dots, I_{\mathcal{A}}(f_m))$  oder kurz  $(U_{\mathcal{A}}, P_1, \dots, P_n, f_1, \dots, f_m)$ .
- Für eine MSO-Formel  $F = F(x_1, \dots, x_n, X_1, \dots, X_m)$  und  $a_1, \dots, a_n \in U_{\mathcal{A}}, A_1, \dots, A_m \subseteq U_{\mathcal{A}}$  schreiben wir auch  $\mathcal{A} \models F(a_1, \dots, a_n, A_1, \dots, A_m)$  anstatt  $\mathcal{A}_{[x_1/a_1, \dots, x_n/a_n, X_1/A_1, \dots, X_m/A_m]} \models F$ .

Die **MSO-Theorie** einer Struktur  $\mathcal{A}$  ist die Menge aller MSO-Sätze  $F$  mit  $\mathcal{A} \models F$ .

Ein Beispiel für eine nützliche MSO-Formel:

Sei  $\theta(x, y)$  eine MSO-Formel.

Die folgende Formel  $\text{reach}_\theta(x, y)$  drückt aus, dass in der durch  $\theta(x, y)$  definierten binären Relation ein Pfad von  $x$  nach  $y$  existiert:

$$\text{reach}_\theta(x, y) = \forall X \left( (x \in X \wedge \forall y \forall z (\theta(y, z) \wedge y \in X \rightarrow z \in X)) \rightarrow y \in X \right)$$

In Worten: Es gibt einen  $\theta$ -Pfad von  $x$  nach  $y$  genau dann, wenn jede Teilmenge  $X$ , die  $x$  enthält und unter der durch  $\theta$  definierten Relation abgeschlossen ist ( $\forall y \forall z (\theta(y, z) \wedge y \in X \rightarrow z \in X)$ ), auch  $y$  enthält.

# MSO-definierbare Sprachen

Wir wollen MSO-Sätze benutzen, um (formale) Sprachen zu definieren.

Hierzu müssen wir zunächst Wörter als Strukturen repräsentieren.

Sei  $\Sigma$  ein endliches Alphabet im folgenden.

Ein nicht-leeres Wort  $w = a_1 a_2 \cdots a_n$  ( $n \geq 1$ ,  $a_i \in \Sigma$ ) identifizieren wir mit der Struktur

$$\mathcal{A}_w = (\{1, 2, \dots, n\}, <, (P_a)_{a \in \Sigma}),$$

wobei gilt:

- $<$  ist die gewöhnliche Ordnung auf  $\{1, 2, \dots, n\}$
- $P_a$  ist die einstellige Relation  $P_a = \{i \mid 1 \leq i \leq n, a_i = a\}$

Im folgenden schreiben wir einfach  $w$  für  $\mathcal{A}_w$ .

Eine Sprache  $L \subseteq \Sigma^+$  ist **MSO**-definierbar, falls ein MSO-Satz  $F$  existiert mit  $L = \{w \in \Sigma^+ \mid w \models F\}$ .



# MSO-definierbare Sprachen

Für die folgenden Beispiele sei das Alphabet  $\Sigma = \{a, b\}$ .

**Beispiel 1:** Der MSO-Satz

$$\exists x \exists y \exists z (\forall u (x \leq u \wedge u \leq z) \wedge P_a(x) \wedge P_b(y) \wedge P_a(z))$$

definiert die Sprache  $a\Sigma^*b\Sigma^*a$ .

Hier ist  $x \leq u$  eine Abkürzung für  $x < u \vee x = u$ .

**Beispiel 2:** Der MSO-Satz

$$\exists X (\exists x \exists y (\forall u (x \leq u \wedge u \leq y) \wedge x \in X \wedge \neg y \in X) \wedge \forall x \forall y (y = x + 1 \rightarrow (x \in X \leftrightarrow y \notin X)))$$

definiert die Sprache  $\{w \in \{a, b\}^* \mid |w| \text{ ist gerade}\}$ .

Hier ist  $y = x + 1$  eine Abkürzung für die Formel  $x < y \wedge \forall z (x \leq z \leq y \rightarrow (x = z \vee y = z))$ .

## Satz (Büchi, Elgot 1958 und Trachtenbrot 1958)

Eine Sprache  $L$  ist MSO-definierbar genau dann, wenn sie regulär ist.

### Beweis:

1. Sei  $L \subseteq \Sigma^*$  regulär. Wir zeigen, dass  $L$  MSO-definierbar ist.

Sei  $A = (Q, \Sigma, \delta, q_0, F)$  ein deterministischer endlicher Automat mit  $L(A) = L$ , wobei

- $Q$  die endliche Menge der Zustände ist,
- $\delta : Q \times \Sigma \rightarrow Q$  die Überföhrungsfunktion ist,
- $q_0 \in Q$  der Anfangszustand ist, und
- $F \subseteq Q$  die Menge der Endzustände ist.

# Der Satz von Büchi

O.B.d.A. sei  $Q = \{1, \dots, n\}$ .

Dann definiert der folgende MSO-Satz die Sprache  $L = L(A)$ :

$$\exists X_1 \exists X_2 \cdots \exists X_n$$

$$\bigvee_{p \neq q} X_p \cap X_q = \emptyset \wedge \forall x \bigvee_{q \in Q} x \in X_q \wedge$$

$$\exists x (\forall y (x \leq y) \wedge \bigvee_{a \in \Sigma} (P_a(x) \wedge x \in X_{\delta(q_0, a)})) \wedge$$

$$\exists x (\forall y (y \leq x) \wedge \bigvee_{q \in F} x \in X_q) \wedge$$

$$\forall x \forall y (y = x + 1 \rightarrow \bigvee_{q \in Q} \bigvee_{a \in \Sigma} (x \in X_q \wedge P_a(y) \wedge y \in X_{\delta(q, a)}))$$

Hierbei ist  $X_p \cap X_q = \emptyset$  eine Abkürzung für  $\neg \exists x (x \in X_p \wedge x \in X_q)$ .

2. Sei  $L \subseteq \Sigma^*$  MSO-definierbar. Wir zeigen, dass  $L$  regulär ist.

Sei  $V \subseteq \text{Var}_1 \cup \text{Var}_2$  eine endliche Menge von Variablen.

Ein nicht-leeres Wort

$$w = (a_1, V_1)(a_2, V_2) \cdots (a_k, V_k) \in (\Sigma \times 2^V)^*$$

( $k \geq 1$ ,  $a_i \in \Sigma$ ,  $V_k \subseteq V$ ) nennen wir **gültig** falls es für jede Variable  $x \in V \cap \text{Var}_1$  genau ein  $1 \leq i \leq k$  mit  $x \in V_i$  gibt.

Für solch ein gültiges Wort  $w$  definieren wir die Abbildung  $f_w : V \rightarrow \{1, \dots, k\} \cup 2^{\{1, \dots, k\}}$  durch:

- $f_w(x) = i$  falls  $x \in V_i \cap \text{Var}_1$ .
- $f_w(X) = \{i \mid X \in V_i\}$  für  $X \in V \cap \text{Var}_2$ .

Weiter identifizieren wir ein gültiges Wort  $w = (a_1, V_1)(a_2, V_2) \cdots (a_k, V_k)$  mit der Struktur  $\mathcal{A}_w = (\{1, \dots, k\}, I_w)$ , wobei

- $I_w(x) = f_w(x)$  für  $x \in V \cap \text{Var}_1$ ,
- $I_w(X) = f_w(X)$ , für  $X \in V \cap \text{Var}_2$ ,
- und  $I_w$  auf den Symbolen  $<, P_a$  ( $a \in \Sigma$ ) genau so definiert ist wie die Struktur  $\mathcal{A}_v$  für  $v = a_1 a_2 \cdots a_k$ .

Für eine MSO-Formel  $F$  mit  $\text{free}(F)$  sei  $L(F)$  die Menge aller nicht-leeren gültigen Wörter  $w$  über dem Alphabet  $\Sigma \times 2^{\text{free}(F)}$  mit  $w \models F$ .

**Beweisstrategie:** Wir konstruieren für jede Formel  $F$  durch Induktion über den Aufbau von  $F$  einen endlichen Automaten  $A_F$  für die Sprache  $L(F)$ .

# Der Satz von Büchi

Zunächst kann man für jede endliche Variablenmenge  $V \subseteq \text{Var}_1 \cup \text{Var}_2$  einen Automaten  $A_V$  konstruieren, der genau die gültigen Wörter aus  $(\Sigma \times 2^V)^*$  akzeptiert.

1. Fall:  $F = (x = y)$ . Konstruiere  $A_F$  so, dass

$$L(A_F) = (\Sigma \times \{\emptyset\})^*(\Sigma \times \{x, y\})(\Sigma \times \{\emptyset\})^*.$$

2. Fall:  $F = (x < y)$ . Konstruiere  $A_F$  so, dass

$$L(A_F) = (\Sigma \times \{\emptyset\})^*(\Sigma \times \{x\})(\Sigma \times \{\emptyset\})^*(\Sigma \times \{y\})(\Sigma \times \{\emptyset\})^*.$$

3. Fall:  $F = P_a(x)$ . Konstruiere  $A_F$  so, dass

$$L(A_F) = (\Sigma \times \{\emptyset\})^*(a, \{x\})(\Sigma \times \{\emptyset\})^*.$$

4. Fall:  $F = (x \in X)$ . Konstruiere  $A_F$  so, dass

$$L(A_F) = (\Sigma \times \{\emptyset, \{X\}\})^*(\Sigma \times \{x, X\})(\Sigma \times \{\emptyset, \{X\}\})^*.$$

# Der Satz von Büchi

5. Fall:  $F = \neg G$ . Sei  $V = \text{free}(G)$ . Konstruiere  $A_F$  so, dass

$$L(A_F) = L(A_V) \setminus L(A_G).$$

6. Fall:  $F = G \vee H$ .

Sei  $V_G = \text{free}(G)$ ,  $V_H = \text{free}(H)$  und  $V = \text{free}(F) = V_G \cup V_H$ .

Definiere Homomorphismen  $g : (\Sigma \times 2^V)^* \rightarrow (\Sigma \times 2^{V_G})^*$  und  $h : (\Sigma \times 2^V)^* \rightarrow (\Sigma \times 2^{V_H})^*$  durch

$$g(a, S) = (a, S \cap V_G),$$

$$h(a, S) = (a, S \cap V_H).$$

Konstruiere nun Automaten  $A'_G$  und  $A'_H$ , so dass

$$L(A'_G) = L(A_V) \cap g^{-1}(L(A_G)),$$

$$L(A'_H) = L(A_V) \cap h^{-1}(L(A_H)).$$

Der Automat  $A_F$  wird dann so konstruiert, dass  $L(A_F) = L(A'_G) \cup L(A'_H)$ .

# Der Satz von Büchi

7. Fall:  $F = \exists x G$ .

Sei  $V = \text{free}(G)$  und damit  $\text{free}(F) = V \setminus \{x\}$ .

Definiere den Homomorphismen  $f : (\Sigma \times 2^V)^* \rightarrow (\Sigma \times 2^{V \setminus \{x\}})^*$  durch

$$f(a, S) = (a, S \setminus \{x\}).$$

Konstruiere dann den Automaten  $A_F$  so, dass  $L(A_F) = f(L(A_G))$ .

8. Fall:  $F = \exists X G$ .

Sei  $V = \text{free}(G)$  und damit  $\text{free}(F) = V \setminus \{X\}$ .

Definiere den Homomorphismen  $f : (\Sigma \times 2^V)^* \rightarrow (\Sigma \times 2^{V \setminus \{X\}})^*$  durch

$$f(a, S) = (a, S \setminus \{X\}).$$

Konstruiere dann den Automaten  $A_F$  so, dass  $L(A_F) = f(L(A_G))$ .

Dies beendet den Beweis des Satzes von Büchi. □