Beispiel 2.10

Sei $\Sigma = \{a,b\}$. Die Sprache

$$L_{even} := \{ w \in \{a,b\}^* : |w| \text{ ist gerade} \}$$

wird durch folgenden EMSO[σ_Σ]-Satz φ_{even} beschrieben:

$$\varphi_{even} := \exists X \Big(\forall x \,(\min(x) \rightarrow \neg X(x)) \wedge \forall x\, (\max(x) \rightarrow X(x)) \wedge \forall x \forall y \,(\mathrm{succ}(x,y) \rightarrow (X(x) \leftrightarrow \neg X(y)))\Big)$$

mit
$$\begin{cases} \min(x) := \forall z\ x \leq z \\ \max(x) := \forall z\ z \leq x \\ \mathrm{succ}(x,y) := x \leq y \wedge \neg x = y \wedge \forall z\ (z \leq x \vee y \leq z) \end{cases}$$

diese Abkürzungen werden wir später immer wieder nutzen.

(Idee: "X enthält genau die "geraden" Positionen eines Worts, in dem die Formel ausgewertet wird, und die letzte Position des Worts gehört zu X").

Theorem 2.11 (Der Satz von Büchi)

Sei Σ ein endliches (nicht-leeres) Alphabet und sei $L \subseteq \Sigma^*$. Die folgenden Aussagen sind äquivalent:

(a) L ist regulär (dh wird von einem deterministischen endlichen Automaten erkannt)

(b) L ist EMSO-definierbar.

(c) L ist MSO-definierbar.

Beweis:

Wir beweisen die Richtung (a) $\Rightarrow$ (b) im folgenden Lemma:

Lemma 2.12

Jede reguläre Sprache $L \subseteq \Sigma^+$ ist EMSO-definierbar.

Beweis:

Sei $L \subseteq \Sigma^+$ regulär. D.h. L wird von einem deterministischen endlichen Automaten $\mathcal{A} = (Q, \Sigma, q_0, \delta, F)$ erkannt.

OBdA sei $Q = \{0, 1, \dots, m\}$ für ein $m \in \mathbb{N}_{\geq 1}$,

$q_0 = 0$,

$F \subseteq Q$,

$\delta : Q \times \Sigma \to Q$.

Idee: Eingabewort $\quad w = w_1 \, w_2 \, w_3 \, w_4 \cdots w_n$

Lauf von $\mathcal{A}$: $\quad q_0 \; q_1 \; q_2 \; q_3 \; q_4 \cdots q_n$

Für jeden Zustand $q \in Q$ "rate" die Menge Z_q der Positionen i von w, s.d. gilt: direkt nach dem Lesen des Buchstaben w_i ist $\mathcal{A}$ im Zustand q.

Wir konstruieren einen EMSO[σ_Σ]-Satz $\Phi_{\mathcal{A}}$, s.d. f.a. $w \in \Sigma^+$ gilt: $\mathcal{A}$ akzeptiert w $\iff$ $\mathfrak{A}_w \models \Phi_{\mathcal{A}}$.

Dann wählen wir $\Phi_\mathcal{A}$ wie folgt:

$$\Phi_\mathcal{A} := \exists Z_0 \dots \exists Z_m \Big(\varphi_{\text{Zustand}} \wedge \varphi_{\text{start}} \wedge \varphi_{\text{schritt}} \wedge \varphi_{\text{akzeptiere}} \Big)$$

mit

- $\underline{\varphi_{\text{Zustand}}}$: besagt, dass jede Position x zu genau <u>einer</u> der Mengen $Z_0, \dots, Z_m$ gehört:

$$\forall x \bigvee_{i=0}^{m} \Big(Z_i(x) \wedge \bigwedge_{j \neq i} \neg Z_j(x) \Big)$$

- $\underline{\varphi_{\text{start}}}$: besagt, dass der Automat im Startzustand q_0 startet und dass an Position 1 der Zustand $\delta(q_0, w_1)$ steht (wobei w_1 der Buchstabe an Pos 1 ist):

$$\forall x \Big(\underbrace{\min(x)}_{\forall y\, x \leq y} \rightarrow \bigwedge_{a \in \Sigma} \big(P_a(x) \rightarrow Z_{\delta(q_0, a)}(x) \big) \Big)$$

- $\underline{\varphi_{\text{schritt}}}$: besagt, dass von einem Schritt zum nächsten die Überführungsfunktion δ beachtet wird:

$$\forall x \forall y \Big(succ(x, y) \rightarrow \bigwedge_{\substack{a \in \Sigma, \\ q \in Q}} \big((Z_q(x) \wedge P_a(y)) \rightarrow Z_{\delta(q, a)}(y) \big) \Big)$$

- $\underline{\varphi_{\text{akzeptiere}}}$ besagt, dass der letzte Zustand akzeptierend ist: $\forall x \big(\max(x) \rightarrow \bigvee_{q \in F} Z_q(x) \big)$.

<u>Klar:</u> $\Phi_{\mathcal{A}}$ ist ein $EMSO[\sigma_\Sigma]$-Satz.

Man kann leicht nachprüfen, dass f.a. $w \in \Sigma^+$ gilt:
$\mathfrak{A}_w \models \Phi_{\mathcal{A}} \iff \mathcal{A}$ akzeptiert w.

Somit gilt: $\Phi_{\mathcal{A}}$ beschreibt die von $\mathcal{A}$ akzeptierte reguläre Sprache L.

$\square_{\text{Lemma 2.12}}$

Die Richtung (b) $\Rightarrow$ (c) von Theorem 2.11 gilt trivialerweise (da $EMSO[\sigma_\Sigma] \subseteq MSO[\sigma_\Sigma]$).

Zum Beweis der Richtung (c) $\Rightarrow$ (a) von Theorem 2.11 bringen wir einen gegebenen $MSO[\sigma_\Sigma]$-Satz zunächst in eine für die Übersetzung in einen endlichen Automaten besonders geeignete Form.

Dazu nutzen wir die folgenden $MSO[\sigma_\Sigma]$-Formeln:

- $\text{Sing}(X) := \exists y \left(X(y) \wedge \forall z \left(X(z) \rightarrow z = y \right) \right)$
  besagt: $|X| = 1$ ("X is a singleton set")
- $\text{le}(X,Y) := \forall x \forall y \left(\left(X(x) \wedge Y(y) \right) \rightarrow x \leq y \right)$
  besagt: "Kein Element von Y liegt links von einem Element in X"
- $\text{sub}(X,Y) := \forall x \left(X(x) \rightarrow Y(x) \right)$ besagt "$X \subseteq Y$"
- $\text{symb}_a(X) := \forall x \left(X(x) \rightarrow P_a(x) \right)$, f.a. $a \in \Sigma$
  besagt: "alle Positionen in X tragen den Buchstaben a"

Eine gegebene $MSO[\sigma_\Sigma]$-Formel φ transformieren wir nun in eine "äquivalente" $MSO[\sigma_\Sigma]$-Formel φ^*, indem wir jede Individuenvariable x durch eine <u>neue</u> Mengenvariable V_x ersetzen und atomare Formeln durch geeignete Kombinationen der Formeln singl, le, sub, $symb_a$ ersetzen:

- Für $\varphi = x \leq y$ ist $\varphi^* := le(V_x, V_y)$
- Für $\varphi = x = y$ ist $\varphi^* := le(V_x, V_y) \land le(V_y, V_x)$
- Für $\varphi = P_a(x)$ ist $\varphi^* := symb_a(V_x)$
- Für $\varphi = Z(x)$ ist $\varphi^* := sub(V_x, Z)$
- Für $\varphi = \exists x \psi$ ist $\varphi^* := \exists V_x (singl(V_x) \land \psi^*)$
- Für $\varphi = \forall x \psi$ ist $\varphi^* := \forall V_x (singl(V_x) \to \psi^*)$
- Für $\varphi = \neg\psi$ ist $\varphi^* := \neg \psi^*$
- Für $\varphi = (\varphi_1 \square \varphi_2)$ mit $\square \in \{\land, \lor, \to, \leftrightarrow\}$ ist
  $\varphi^* := (\varphi_1^* \square \varphi_2^*)$
- Für $\varphi = QZ\psi$ mit $Q \in \{\exists, \forall\}$ und $Z \in Var_2$ ist
  $\varphi^* := QZ\psi^*$

<u>Lemma 2.13</u>
Für jeden $MSO[\sigma_\Sigma]$-Satz φ gilt: φ^* beschreibt dieselbe Sprache wie φ.

<u>Beweis:</u> Übung.

Als Nächstes konstruieren wir induktiv für jede Teilformel von φ^* einen "äquivalenten" endlichen Automaten. Formeln der Form

$\mathrm{singl}(X)$, $\mathrm{le}(X,Y)$, $\mathrm{sub}(X,Y)$, $\mathrm{symb}_a(X)$

behandeln wir dabei als "atomar".

Insbes: Die Teilformeln, die wir betrachten müssen, haben keine freien Individuenvariablen, wohl aber freie Mengenvariablen. Belegungen dieser Mengenvariablen repräsentieren wir durch ein größeres Alphabet:

Seien $X_1, \ldots, X_k$ (mit $k \in \mathbb{N}$) die Mengenvariablen, die in φ^* vorkommen. Sei

$$\Sigma' := \Sigma \times \{0,1\}^k .$$

Ein Wort $w = w_1 \cdots w_n \in \Sigma^+$ (mit $w_i \in \Sigma$) und Belegungen $X_1^w, \ldots, X_k^w \subseteq \{1, \ldots, n\}$ der Mengenvariablen $X_1, \ldots, X_k$ in w repräsentieren wir durch das Wort $u_{w, \overline{X}^w} = u_1 \cdots u_n \in (\Sigma')^+$ mit

$$u_i := (w_i, b_{1,i}, \ldots, b_{k,i}) \in \Sigma' \text{ mit}$$

$$b_{j,i} := \begin{cases} 1 & \text{falls } i \in X_j^w \\ 0 & \text{falls } i \notin X_j^w \end{cases} \qquad \text{f.a. } j \in \{1, \ldots, k\} \text{ und } i \in \{1, \ldots, n\}.$$

Umgekehrt repräsentiert jedes Wort

$u = u_1 \cdots u_n \in (\Sigma')^+$

ein Wort $w = w_1 \cdots w_n \in \Sigma^+$ und Belegungen $X_1^w, \ldots, X_k^w \subseteq \{1, \ldots, n\}$ wie folgt:

Ist ~~für~~ $i \in \{1 \ldots n\}$ $u_i = (\sigma_i, b_{i1}, \ldots, b_{ik})$, so ist $w_i = \sigma_i$ f.a. $i \in \{1 \ldots n\}$ und

$X_j^w = \{ i \in \{1 \ldots n\} : b_{ij} = 1 \}$ f.a. $j \in \{1 .. k\}$.

Setze $\mathfrak{A}'_u := (\mathfrak{A}_w, X_1^w, \ldots, X_k^w)$

Eine Teilformel ψ von φ^* <u>definiert</u> die Sprache

$L'(\psi) := \{ u \in (\Sigma')^+ : \mathfrak{A}'_u \models \psi \}$.

<u>Lemma 2.14</u>

Sei $k \in \mathbb{N}$, $\Sigma' := \Sigma \times \{0,1\}^k$, seien $i, j \in \{1, \ldots, k\}$, sei $a \in \Sigma$.

Es gibt nichtdeterministische endliche Automaten

$A_{sing(x_i)}$, $A_{le(x_i,x_j)}$, $A_{sub(x_i,x_j)}$, $A_{symb_a(x_i)}$,

so dass gilt:

<u>(a)</u> $A_{sing(x_i)}$ erkennt die Sprache $L'(sing(x_i)) \subseteq (\Sigma')^+$

<u>(b)</u> $A_{le(x_i,x_j)}$ erkennt die Sprache $L'(le(x_i,x_j)) \subseteq (\Sigma')^+$

<u>(c)</u> $A_{sub(x_i,x_j)}$ erkennt die Sprache $L'(sub(x_i,x_j)) \subseteq (\Sigma')^+$

<u>(d)</u> $A_{symb_a(x_i)}$ erkennt die Sprache $L'(symb_a(x_i)) \subseteq (\Sigma')^+$

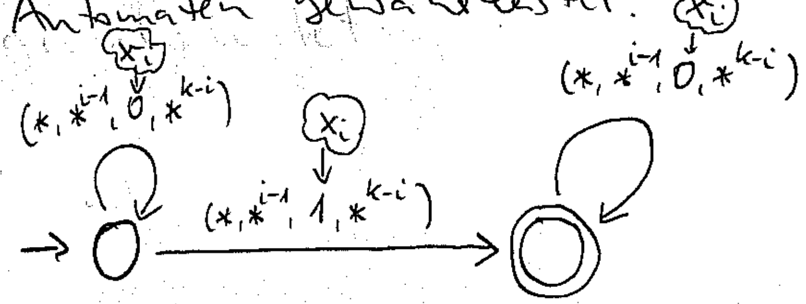
Beweis:

(a) Der Automat $A_{sing(X_i)}$ soll ein Wort $u = u_1 \cdots u_n \in (\Sigma')^+$ genau dann akzeptieren, wenn gilt: es gibt genau eine Position $p \in \{1, \dots, n\}$, so dass der Buchstabe

$$u_p = (w_p, b_{p,1}, \dots, b_{p,k}) \in \Sigma \times \{0,1\}^k$$

an der zu X_i gehörigen Komponente $b_{p,i}$ eine 1 hat.

Dies wird durch den folgenden nichtdeterministischen Automaten gewährleistet:

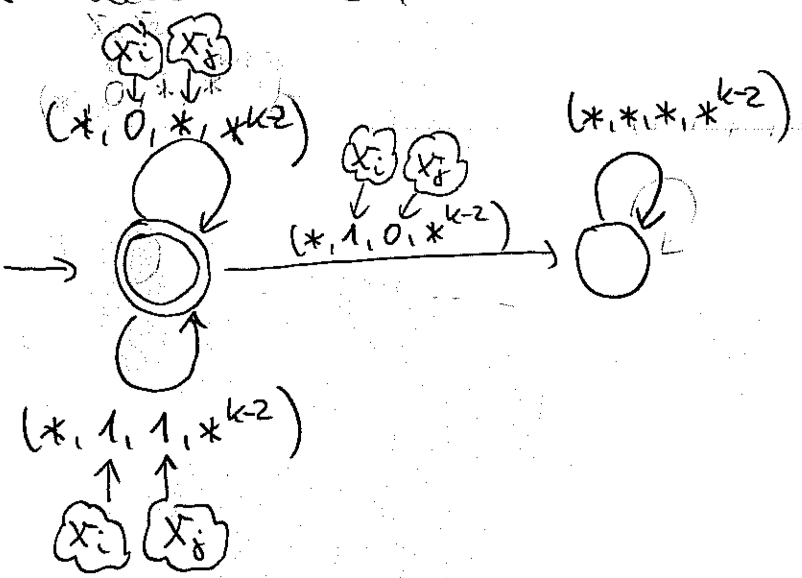


wobei ein "$*$" bedeutet, dass in der entsprechenden Komponente jeder mögliche Wert stehen kann.

(b) Übung

(c) $A_{sub(X_i, X_j)}$ soll ein Wort genau dann akzeptieren, wenn an jeder Position p für den dortigen Buchstaben $u_p = (w_p, b_{p,1}, \dots, b_{p,k}) \in \Sigma \times \{0,1\}^k$ gilt: Falls $b_{p,i} = 1$, so auch $b_{p,j} = 1$

Dies wird durch den folgenden Automaten gewährleistet: (OBdA betrachten wir hier den Fall, dass $i=1$ und $j=2$ ist:



(d) Übung. $\square$

<u>Lemma 2.15</u>

Sei φ ein $MSO[\sigma_\Sigma]$-Satz, seien $X_1, \dots, X_k$ die in φ^* vorkommenden Mengenvariablen und sei $\Sigma' := \Sigma \times \{0,1\}^k$. Für jede Teilformel ψ von φ^* (wobei Formeln der Form $Sing(X_i)$, $le(X_i, X_j)$, $succ(X_i, X_j)$, $symb_a(X_i)$ als "atomar" betrachtet werden) gibt es einen nichtdeterministischen endlichen Automaten A_ψ, so dass gilt:

A_ψ erkennt die Sprache $L'(\psi) \subseteq (\Sigma')^+$.

Beweis:

Per Induktion nach dem Aufbau von ψ.
Der Induktionsanfang für "atomares" ψ wird durch Lemma 2.14 gewährleistet.

Induktionsschritt:

Fall 1: $\psi = \neg\chi$

Gemäß Induktionsannahme gibt es einen Automaten A_χ, der die Sprache $L'(\chi)$ erkennt.
klar: $L'(\psi) = \{ u \in (\Sigma')^+ : u \notin L'(\chi)\}$, dh
$u \in L'(\psi) \iff u \notin L'(\chi)$, f.a. $u \in (\Sigma')^+$
Dh wir erhalten den gesuchten Automaten A_ψ, indem wir A_χ zunächst deterministisch machen ($\nearrow$ Potenzmengenkonstruktion) und dann akzeptierende Zustände zu nichtakzeptierenden Zuständen machen und umgekehrt.

Fall 2: $\psi = (\psi_1 \vee \psi_2)$

Gemäß Induktionsannahme gibt es Automaten A_{ψ_1} und A_{ψ_2}, die die Sprachen $L'(\psi_1)$ und $L'(\psi_2)$ erkennen. OBdA sind deren Zustandsmengen disjunkt. Den nichtdeterministischen Automaten A_ψ erhalten wir, indem wir die Automaten A_{ψ_1} und A_{ψ_2} vereinigen und deren Startzustände miteinander identifizieren.

D.h.: Ist $A_{\psi_i} = (Q_i, \Sigma', q_{0,i}, \Delta_i, F_i)$ mit

$\Delta_i \subseteq Q_i \times \Sigma_i \times Q_i$ und $Q_1 \cap Q_2 = \emptyset$,

So ist $A_\psi = (Q, \Sigma', q_0, \Delta, F)$ mit

- $Q = Q_1 \dot\cup Q_2 \dot\cup \{q_0\}$
- $F = F_1 \cup F_2 \cup M$ mit $M = \begin{cases} \{q_0\} & \text{falls } q_{0,1} \in F_1 \text{ oder } q_{0,2} \in F_2 \\ \emptyset & \text{sonst} \end{cases}$
- $\Delta = \Delta_1 \cup \Delta_2 \cup \{ (q_0, \sigma', q) : (q_{0,1}, \sigma', q) \in \Delta_1 \text{ oder } (q_{0,2}, \sigma', q) \in \Delta_2 \}$.

<u>Fall 3:</u> $\psi = (\psi_1 \square \psi_2)$ mit $\square \in \{\wedge, \rightarrow, \leftrightarrow\}$:

Durch geeignete Kombination der Fälle 1 und 2 (da z.B. $(\psi_1 \rightarrow \psi_2)$ äquivalent ist zu $(\neg\psi_1 \vee \psi_2)$).

<u>Fall 4:</u> $\psi = \exists X_i \chi$ mit $i \in \{1, \dots, k\}$

Gemäß Induktionsannahme gibt es bereits einen Automaten A_χ, der die Sprache $L'(\chi)$ erkennt.

<u>Klar:</u> Für jedes $u = u_1 \cdots u_n \in (\Sigma')^+$ mit

$u_p = (w_p, b_{p,1}, \dots, b_{p,k}) \in \Sigma \times \{0,1\}^k$ f.a. $p \in \{1 \dots n\}$

gilt:

(△) $\begin{cases} u \in L'(\psi) \iff \text{Es gibt Werte } \tilde b_1 \cdots \tilde b_n \in \{0,1\}^n\text{, die} \\ \text{für die "}X_i\text{"-Komponente eingesetzt werden können, s.d. für} \\ \tilde u = \tilde u_1 \cdots \tilde u_n \text{ mit } \tilde u_p = (w_p, b_{p,1}, \dots, b_{p,i-1}, \tilde b_p, b_{p,i+1}, \dots, b_{p,n}) \text{ f.a.} \end{cases}$

$p \in \{1, \dots, n\}$ gilt: $\quad \tilde{u} \in L'(\chi)$. (dh: $\tilde{u}$ wird von A_χ akzeptiert)

Somit wird die Sprache $L'(\psi)$ durch den nichtdeterministischen Automaten A_ψ erkannt, der aus A_χ entsteht, indem in der Beschriftung jedes Pfeils der graphischen Darstellung von A_χ die X_i-Komponente durch das "Wildcard"-Symbol $*$ ersetzt wird. D.h.: Ist

$$A_\chi = (Q, \Sigma', q_0, \Delta, F) \quad , \text{ so ist}$$

$$A_\psi = (Q, \Sigma', q_0, \tilde{\Delta}, F) \quad \text{mit}$$

$$\tilde{\Delta} = \{ (q, \tilde{\sigma}, q') : \begin{array}{l} (q, \sigma, q') \in \Delta \text{ und} \\ \tilde{\sigma} = (\tilde{a}, \tilde{b}_1, \dots, \tilde{b}_k) \in \Sigma \times \{0,1\}^k \\ \text{entsteht aus } \sigma = (a, b_1, \dots, b_k) \in \Sigma \times \{0,1\}^k, \\ \text{indem der Wert } b_i \text{ durch einen} \\ \text{beliebigen Wert } \tilde{b}_i \in \{0,1\} \text{ ersetzt} \\ \text{wird} \} \end{array}$$

Man kann leicht nachprüfen, dass für alle Worte $u \in (\Sigma')^+$ gilt: A_ψ akzeptiert $u \iff$ (Δ) gilt.

Falls 5: $\psi = \forall X_i\, \chi \quad$ mit $i \in \{1, \dots, k\}$:

Durch Kombination der Fälle 4 und 1, da $\forall X_i \chi$ äquivalent ist zu $\neg \exists X_i \neg \chi$.

Beachte: Dies sind alle Fälle, die auftreten können.

$\square$ Lemma 2.15

Wir können nun die Richtung (c) $\Rightarrow$ (a) von Theorem 2.11 beweisen:

Lemma 2.16:

Für jeden $MSO[\sigma_\Sigma]$-Satz φ gibt es einen nichtdeterministischen endlichen Automaten A_φ, der die von φ beschriebene Sprache erkennt.

Beweis:

Wir nutzen Lemma 2.13 und 2.15, um die Formel φ^* und den Automaten A_{φ^*} zu konstruieren, der die Sprache $L'(\varphi^*) \subseteq (\Sigma')^+$ erkennt, wobei $\Sigma' = \Sigma \times \{0,1\}^k$ ist und k so gewählt ist, dass $X_1, \dots, X_k$ die in φ^* vorkommenden Mengenvariablen sind.

Dh: A_{φ^*} akzeptiert ein Wort $u = u_1 \dots u_n \in (\Sigma')^+$ genau dann, wenn $\mathfrak{A}'_u = (\mathfrak{A}_w, X_1^w, \dots, X_k^w)$ die Formel φ^* erfüllt. Da φ^* keine freien Variablen hat, gilt: $\mathfrak{A}'_u \models \varphi^* \iff \mathfrak{A}_w \models \varphi^*$ (dh: die jeweilige Belegung der Mengenvariablen $X_1, \dots, X_k$ ist irrelevant).

Somit erhalten wir einen nichtdeterministischen Automaten A_φ, der genau diejenigen Worte $w \in \Sigma^+$ akzeptiert, für die $\mathfrak{A}_w \models \varphi^*$ gilt, indem wir in der graphischen Darstellung von A_{φ^*} in der Beschriftung jedes Pfeils die Komponenten für $X_1, \dots, X_k$ weglassen. D.h.:

Ist $A_{\varphi^*} = (Q, \Sigma', q_0, \Delta, F)$, so ist

$$A_\varphi = (Q, \Sigma, q_0, \tilde{\Delta}, F) \quad \text{mit}$$

$$\tilde{\Delta} = \{ (q, \sigma, q') : \text{ es gibt } b_1, \dots, b_k \in \{0,1\}, \text{ s.d. } (q, (\sigma, b_1, \dots, b_k), q') \in \Delta \}$$

Man kann leicht nachprüfen, dass für alle Worte $w \in \Sigma^+$ gilt: A_φ akzeptiert $w \iff \mathfrak{M}_w \models \varphi$.

Insbes.: Die von φ beschriebene Sprache ist regulär

$\square$ Lemma 2.16

Dies beendet den Beweis des Satzes von Büchi

$\square$ Theorem 2.11

<u>Beachte</u>: Der Satz von Büchi besagt insbesondere, dass <u>existentielle</u> monadische Logik zweiter Stufe auf <u>Worten</u> dieselbe Ausdrucksstärke besitzt wie die "volle" monadische Logik zweiter Stufe (und genau die regulären Sprachen beschreiben kann).

Aus dem Gebiet der "formalen Sprachen" sind viele Methoden bekannt, mit denen man nachweisen kann, dass bestimmte Sprachen nicht regulär sind (z.B.: Pumping-Lemma, Satz von Myhill und Nerode, Abschlusseigenschaften ..., siehe die Vorlesung "Theoretische Informatik 2").
Insbes. wissen wir, dass die Sprache

$$L_{a^n b^n} := \{ a^n b^n : n \in \mathbb{N}_{\geq 1} \}$$

nicht regulär ist.
Gemäß Satz von Büchi kann $L_{a^n b^n}$ also auch nicht durch einen MSO-Satz beschrieben werden. Durch Anwenden einer "logischen Reduktion" können wir daraus folgern, dass z.B. bestimmte Graphen-Eigenschaften nicht in MSO beschrieben werden können:

Satz 2.17 ("Hamiltonkreis ist nicht MSO-definierbar")
Sei $\sigma_{Graph} = \{E\}$ für ein 2-stelliges Relationssymbol E.
Es gibt keinen MSO[σ_{Graph}]-Satz φ, so dass für jeden endlichen Graphen G gilt:

$$G \models \varphi \iff G \text{ besitzt einen Hamiltonkreis.}$$

Beweis: Durch Widerspruch.

Angenommen, φ_{Ham} ist ein $MSO[\sigma_{Graph}]$-Satz, s.d. für jeden endlichen Graphen G gilt:

$G \models \varphi_{Ham} \iff G$ besitzt einen Hamiltonkreis.

Ziel: Modifiziere φ_{Ham} zu einem $MSO[\sigma_\Sigma]$-Satz $\varphi_{a^nb^n}$ (für $\Sigma = \{a, b\}$), der die Sprache $L_{a^nb^n} = \{ a^n b^n : n \geq 1\}$ beschreibt.

Idee: Einem Wort $w \in \{a,b\}^+$ ordnen wir den Graphen $G_w = (V_w, E_w)$ zu mit

- Knotenmenge V_w : die Menge der Positionen des Worts w
- Kantenmenge E_w: zwischen zwei Positionen x und y gibt es genau dann eine Kante, wenn die beiden Positionen unterschiedliche Buchstaben tragen. D.h.:

$$E(x,y) \iff \big(P_a(x) \wedge P_b(y)\big) \vee \big(P_b(x) \wedge P_a(y)\big)$$

Beachte: G_w ist ein vollständiger bipartiter Graph, dessen Knotenmenge in $P_a^w \,\dot\cup\, P_b^w$ partitioniert ist.

Insbes.: G_w besitzt einen Hamiltonkreis

$\iff$ $|P_a^w| = |P_b^w|$, dh w besitzt genau so viele a's wie b's.

Daher wird $L_{a^n b^n}$ durch folgenden MSO$[\sigma_2]$-Satz $\varphi_{a^n b^n}$ beschrieben:

$$\varphi_{a^n b^n} := \forall x \forall y \Big(\big(P_a(x) \wedge P_b(y) \big) \to x \leq y \Big) \wedge$$

$$\varphi'_{Ham} ,$$

wobei φ'_{Ham} aus φ_{Ham} entsteht, indem jedes Atom der Form $E(x,y)$ ersetzt wird durch die Formel $\Big(\big(P_a(x) \wedge P_b(y) \big) \vee \big(P_b(x) \wedge P_a(y) \big) \Big)$.

Dies ist ein Widerspruch, da gemäß Satz von Büchi dann $L_{a^n b^n}$ regulär sein müsste. $\square$