Theoretische Informatik HS24

Nicolas Wehrli

Übungsstunde 03

8. Oktober 2024

ETH Zürich
nwehrl@ethz.ch

Heute

1. Feedback zur Serie
2. Endliche Automaten - Einführung
3. Beweise für Nichtregularität
    - Theorie für Nichtregularitätsbeweise

Feedback zur Serie

Feedback

- Falsche Annahme zur Kolmogorov Komplexität von natürlichen Zahlen

$$K(n) \neq \lceil \log_2(n+1) \rceil + c$$

- Ihr müsst Pascal Programme schreiben!
- Kolmogorov Komplexität ist definiert über Pascal-Programme die keinen Input nehmen.

Feedback

Sei Σ ein Alphabet und $x \in \Sigma^*$. Wir sagen, dass ein Algorithmus A das Wort x **generiert**, falls A für die Eingabe λ die Ausgabe x liefert.

Kolmogorov-Komplexität

Für jedes Wort $x \in (\Sigma_{\text{bool}})^*$ ist die **Kolmogorov-Komplexität** $K(x)$ **des Wortes** x das Minimum der binären Längen, der Pascal-Programme, die x generieren.

$\Longrightarrow$ Kolmogorov Komplexität eines Wortes ist die Länge des kürzesten Programms, dass keinen Input nimmt und das Wort ausgibt!

Endliche Automaten - Einführung

Erster Ansatz zur Modellierung von Algorithmen

Ein (deterministischer) **endlicher Automat (EA)** ist ein Quintupel $M = (Q, \Sigma, \delta, q_0, F)$, wobei

(i) Q eine endliche Menge von **Zuständen** ist,

(ii) Σ ein Alphabet, genannt **Eingabealphabet**, ist,

(iii) $q_0 \in Q$ der Anfangszustand ist,

(iv) $F \subseteq Q$ die **Menge der akzeptierenden Zustände** ist und

(v) $\delta : Q \times \Sigma \to Q$ die **Übergangsfunktion** ist.

Konfigurationen

Eine **Konfiguration** von M ist ein Tupel $(q, w) \in Q \times \Sigma^*$.

- "M befindet sich in einer Konfiguration $(q, w) \in Q \times \Sigma^*$, wenn M im Zustand q ist und noch das Suffix w eines Eingabewortes lesen soll."
- Die Konfiguration $(q_0, x) \in \{q_0\} \times \Sigma^*$ heisst die **Startkonfiguration von M auf x**.
- Jede Konfiguration aus $Q \times \{\lambda\}$ nennt man **Endkonfiguration**.

Ein **Schritt** von M ist eine Relation (auf Konfigurationen) $\underset{M}{\vdash} \subseteq (Q \times \Sigma^*) \times (Q \times \Sigma^*)$, definiert durch

$$(q, w) \underset{M}{\vdash} (p, x) \iff w = ax, a \in \Sigma \text{ und } \delta(q, a) = p.$$

Berechnungen

Eine **Berechnung** C von M ist eine endliche Folge $C = C_0, C_1, \ldots, C_n$ von Konfigurationen, so dass

$$C_i \vdash_M C_{i+1} \text{ für alle } 0 \leq i \leq n-1.$$

C ist die **Berechnung von M auf einer Eingabe** $x \in \Sigma^*$, falls $C_0 = (q_0, x)$ und $C_n \in Q \times \{\lambda\}$ eine Endkonfiguration ist.

Falls $C_n \in F \times \{\lambda\}$, sagen wir, dass C eine **akzeptierende Berechnung** von M auf x ist, und dass M **das Wort** x **akzeptiert**.

Falls $C_n \in (Q \setminus F) \times \{\lambda\}$, sagen wir, dass C eine **verwerfende Berechnung** von M auf x ist, und dass M **das Wort** x **verwirft (nicht akzeptiert)**.

Transitivität von $\vdash_M$ und δ

Sei $M = (Q, \Sigma, \delta, q_0, F)$ ein endlicher Automat. Wir definieren $\vdash_M^*$ als die reflexive und transitive Hülle der Schrittrelation $\vdash_M$ von M; daher ist

$$(q, w) \vdash_M^* (p, u) \iff (q = p \wedge w = u) \text{ oder } \exists k \in \mathbb{N} \setminus \{0\},$$

so dass

(i) $w = a_1 a_2 \ldots a_k u, a_i \in \Sigma$ für $i = 1, 2, \ldots, k$, und

(ii) $\exists r_1, r_2, \ldots, r_{k-1} \in Q$, so dass

$$(q, w) \vdash_M (r_1, a_2 \ldots a_k u) \vdash_M \ldots \vdash_M (r_{k-1}, a_k u) \vdash_M (p, u)$$

Transitivität von $\vdash_M$ und δ

Wir definieren $\hat{\delta} : Q \times \Sigma^* \to Q$ durch:

(i) $\hat{\delta}(q, \lambda) = q$ für alle $q \in Q$ und

(ii) $\hat{\delta}(q, wa) = \delta(\hat{\delta}(q, w), a)$ für alle $a \in \Sigma, w \in \Sigma^*, q \in Q$.

$$\hat{\delta}(q, w) = p \iff (q, w) \vdash_M^* (p, \lambda)$$

Reguläre Sprachen

Die **von** M **akzeptierte Sprache** $L(M)$ ist definiert als

$$\begin{aligned} L(M) &= \{w \in \Sigma^* \mid \text{Berechnung von } M \text{ auf } w \text{ endet in } (p, \lambda) \in F \times \{\lambda\}\} \\ &= \{w \in \Sigma^* \mid (q_0, w) \vdash_M^* (p, \lambda) \wedge p \in F\} \\ &= \{w \in \Sigma^* \mid \hat{\delta}(q_0, w) \in F\} \end{aligned}$$

$\mathcal{L}_{\mathbf{EA}} = \{L(M) \mid M \text{ ist ein EA}\}$ ist die Klasse der Sprachen, die von endlichen Automaten akzeptiert werden.

$\mathcal{L}_{\text{EA}}$ bezeichnet man auch als die **Klasse der regulären Sprachen**, und jede Sprache $L \in \mathcal{L}_{\text{EA}}$ wird **regulär** genannt.

Klassen für alle Zustände im Endlichen Automaten

Für alle $p \in Q$ definieren wir die Klasse

$$\begin{aligned}\mathbf{Kl[p]} &= \{w \in \Sigma^* \mid \hat{\delta}(q_0, w) = p\} \\ &= \{w \in \Sigma^* \mid (q_0, w) \vdash_M^* (p, \lambda)\}\end{aligned}$$

Wir bemerken dann

$$\bigcup_{q \in Q} \mathrm{Kl[q]} = \Sigma^*$$

$$\mathrm{Kl}[q] \cap \mathrm{Kl}[p] = \emptyset, \forall p, q \in Q, p \neq q$$

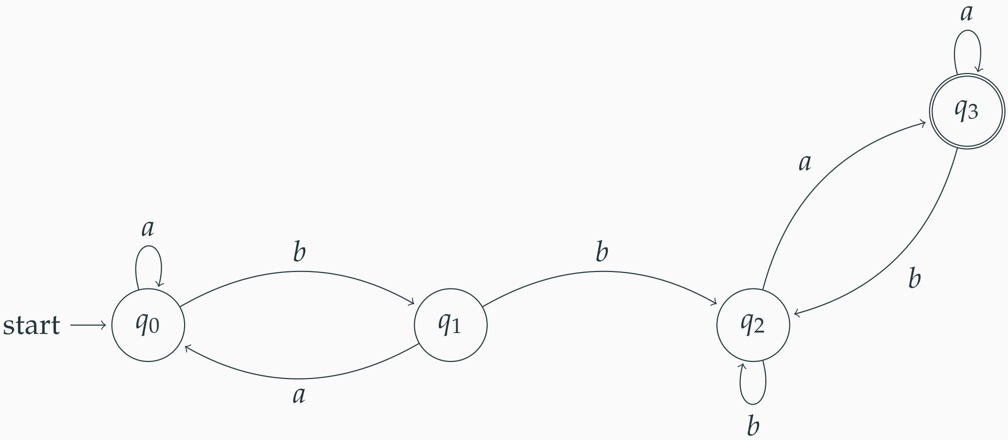
$$L(M) = \bigcup_{q \in F} \mathrm{Kl}[q]$$

EA Konstruktion - Beispielaufgabe

Entwerfen sie für folgende Sprache einen Endlichen Automat und geben Sie eine Beschreibung von $\mathrm{Kl}[q]$ für jeden Zustand $q \in Q$.

$$L_1 = \{xbbya \in \{a, b\}^* \mid x, y \in \{a, b\}^*\}$$

EA Konstruktion - Beispielaufgabe



EA Konstruktion - Beispielaufgabe

Wir beschreiben nun die Klassen für die Zustände q_0, q_1, q_2, q_3:

$\mathrm{Kl}[q_0] = \{wa \in \{a,b\}^* \mid \text{Das Wort } w \text{ enthält nicht die Teilfolge } bb\} \cup \{\lambda\}$

$\mathrm{Kl}[q_1] = \{wb \in \{a,b\}^* \mid \text{Das Wort } w \text{ enthält nicht die Teilfolge } bb\}$

$\mathrm{Kl}[q_3] = \{wa \in \{a,b\}^* \mid \text{Das Wort } w \text{ enthält die Teilfolge } bb\} = L_1$

$\mathrm{Kl}[q_2] = \{a,b\}^* - (\mathrm{Kl}[q_0] \cup \mathrm{Kl}[q_1] \cup \mathrm{Kl}[q_3])$

Produktautomaten - Lemma 3.2

Sei Σ ein Alphabet und seien $M_1 = (Q_1, \Sigma, \delta_1, q_{01}, F_1)$ und $M_2 = (Q_2, \Sigma, \delta_2, q_{02}, F_2)$ zwei EA. Für jede Mengenoperation $\odot \in \{\cup, \cap, -\}$ existiert ein EA M, so dass

$$L(M) = L(M_1) \odot L(M_2).$$

Sei $M = (Q, \Sigma, \delta, q_0, F_\odot)$, wobei

(i) $Q = Q_1 \times Q_2$

(ii) $q_0 = (q_{01}, q_{02})$

(iii) für alle $q \in Q_1, p \in Q_2$ und $a \in Sigma$, $\delta((q,p),a) = (\delta_1(q,a), \delta_2(p,a))$,

(iv) falls $\odot = \cup$, dann ist $F = F_1 \times Q_2 \cup Q_1 \times F_2$
falls $\odot = \cap$, dann ist $F = F_1 \times F_2$, und
falls $\odot = -$, dann ist $F = F_1 \times (Q_2 - F_2)$.

Produktautomat - Beispielaufgabe

Verwenden Sie die Methode des modularen Entwurfs (Konstruktion eines Produktautomaten), um einen endlichen Automaten (in Diagrammdarstellung) für die Sprache

$$L = \{w \in \{a, b\}^* \mid |w|_a = 2 \text{ oder } w = ya\}$$

zu entwerfen. Zeichnen Sie auch jeden der Teilautomaten und geben Sie für die Teilautomaten für jeden Zustand q die Klasse $\mathrm{Kl}[q]$ an.

Produktautomat - Beispielaufgabe

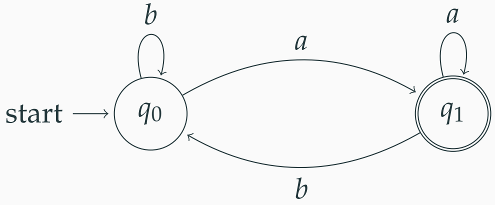
Wir teilen L wie folgt auf:

$$L = L_1 \cup L_2 \text{ wobei gilt:}$$
$$L_1 = \{w \in \{a, b\}^* \mid w = ya\}$$
$$L_2 = \{w \in \{a, b\}^* \mid |w|_a = 2\}$$

Zuerst zeichnen wir die 2 einzelnen Teilautomaten und geben für jeden Zustand q bzw. p die Klasse $\text{Kl}[q]$ respektive $\text{Kl}[p]$ an:

Produktautomat - Beispielaufgabe

erster Teilautomat: $L_1 = \{w \in \{a,b\}^* \mid w = ya\}$

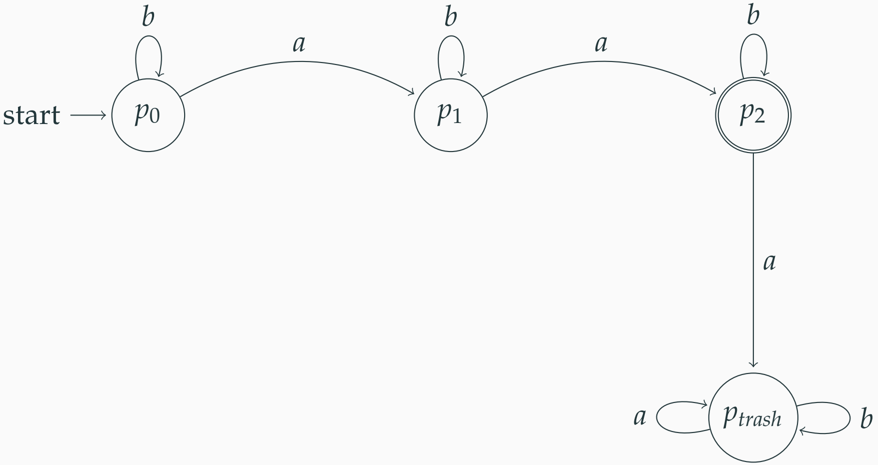


$\mathrm{Kl}[q_0] = \{yb \mid y \in \{a,b\}^*\} \cup \{\lambda\}$

$\mathrm{Kl}[q_1] = \{ya \mid y \in \{a,b\}^*\}$

Produktautomat - Beispielaufgabe

zweiter Teilautomat: $L_2 = \{w \in \{a,b\}^* \mid |w|_a = 2\}$



Produktautomat - Beispielaufgabe

Wir beschreiben nun die Zustände für die Klassen p_0, p_1, p_2, p_{trash}:

$\text{Kl}[p_0] = \{w \in \{a, b\}^* \mid |w|_a = 0\}$

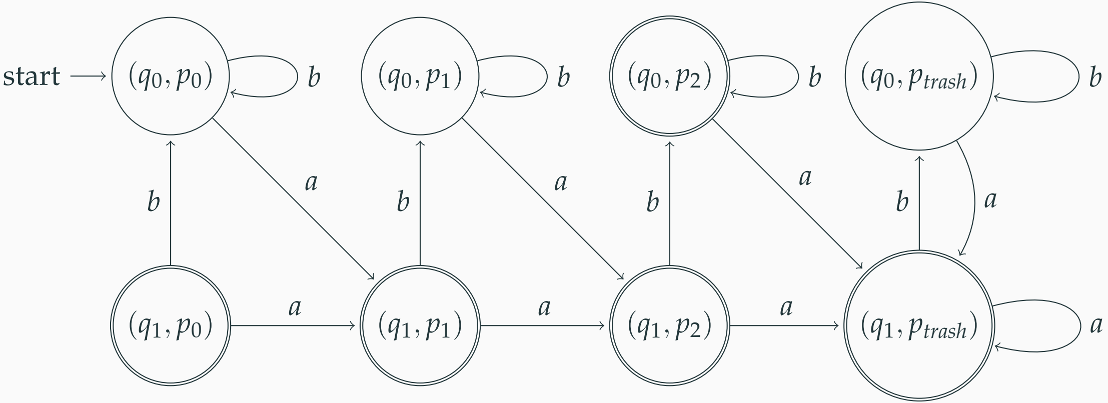
$\text{Kl}[p_1] = \{w \in \{a, b\}^* \mid |w|_a = 1\}$

$\text{Kl}[p_2] = \{w \in \{a, b\}^* \mid |w|_a = 2\}$

$\text{Kl}[p_{trash}] = \{w \in \{a, b\}^* \mid |w|_a > 2\}$

Zum Schluss kombinieren wir diese Teilautomaten zu einem Produktautomaten:

Produktautomat: $L = L_1 \cup L_2$



Beweise für Nichtregularität

Einführung und grundlegende Tipps

i. Wichtiges Unterkapitel. Kommt fast garantiert am Midterm.
ii. Um $L \notin \mathcal{L}_{\mathrm{EA}}$ zu zeigen, genügt es zu beweisen, dass es keinen EA gibt, der L akzeptiert.
iii. Nichtexistenz ist generell sehr schwer zu beweisen, da aber die Klasse der endlichen Automaten sehr eingeschränkt ist, ist dies nicht so schwierig.
iv. Wir führen Widerspruchsbeweise.
v. Es gibt 3 Arten Nichtregularitätsbeweise zu führen (Lemma 3.3, Pumping-Lemma und Kolmogorov-Komplexität).
vi. Ihr müsst alle 3 Methoden können. Ist aber halb so wild.

Theorie für Nichtregularitätsbeweise - Lemma 3.3

Sei $A = (Q, \Sigma, \delta_A, q_0, F)$ ein EA. Seien $x, y \in \Sigma^*, x \neq y$, so dass

$$\hat{\delta}_A(q_0, x) = p = \hat{\delta}_A(q_0, y)$$

für ein $p \in Q$ (also $x, y \in \mathrm{Kl}[p]$). Dann existiert für jedes $z \in \Sigma^*$ ein $r \in Q$, so dass xz und $yz \in \mathrm{Kl}[r]$, also gilt insbesondere

$$xz \in L(A) \iff yz \in L(A)$$

Beweis:

Aus der Existenz der Berechnungen

$(q_0, x) \vdash_A^* (p, \lambda)$ und $(q_0, y) \vdash_A^* (p, \lambda)$ von A folgt die Existenz der Berechnungen auf xz und yz:

$(q_0, xz) \vdash_A^* (p, z)$ und $(q_0, yz) \vdash_A^* (p, z)$ für alle $z \in \Sigma^*$.

Wenn $r = \hat{\delta}_A(p, z)$ ist, dann ist die Berechnung von A auf xz und yz:

$(q_0, xz) \vdash_A^* (p, z) \vdash_A^* (r, \lambda)$ und $(q_0, yz) \vdash_A^* (p, z) \vdash_A^* (r, \lambda)$.

Wenn $r \in F$, dann sind beide Wörter xz und yz in $L(A)$. Falls $r \notin F$, dann sind $xz, yz \notin L(A)$.

■

Bemerkungen

- Von den 3 vorgestellten Methoden, ist diese Methode die einzige, die (unter der richtigen Anwendung) garantiert für jede nichtreguläre Sprache funktioniert.
- Um die Nichtregularität von L zu beweisen, verwenden wir die Endlichkeit von Q und das Pigeonhole-Principle.

Betrachten wir mal eine Beispielaufgabe mit dieser Methode am Paradebeispiel

$$L = \{0^n 1^n \mid n \in \mathbb{N}\}$$

Beispielaufgabe - Lemma 3.3

Nehmen wir zum Widerspruch an L sei regulär.

Dann existiert ein EA $A = (Q, \Sigma, \delta, q_0, F)$ mit $L(A) = L$.

Wir betrachten die Wörter $0^1, \dots, 0^{|Q|+1}$. Per Pigeonhole-Principle existiert O.B.d.A. $i < j$, so dass

$$\hat{\delta}(q_0, 0^i) = \hat{\delta}(q_0, 0^j)$$

Nach Lemma 3.3 gilt

$$0^i z \in L \iff 0^j z \in L$$

für alle $z \in (\Sigma_{\text{bool}})^*$.

Dies führt aber zu einem Widerspruch, weil für $z = 1^i$ das Wort $0^i 1^i \in L$ aber $0^j 1^i \notin L$.

Sei L regulär. Dann existiert eine Konstante $n_0 \in \mathbb{N}$, so dass jedes Wort $w \in \Sigma^*$ mit $|w| \geq n_0$ in drei Teile x, y und z zerlegen lässt, das heisst $w = yxz$, wobei

(i) $|yx| \leq n_0$

(ii) $|x| \geq 1$

(iii) entweder $\{yx^kz \mid k \in \mathbb{N}\} \subseteq L$ oder $\{yx^kz \mid k \in \mathbb{N}\} \cap L = \emptyset$.

Beweis

Sei $L \in \Sigma^*$ regulär. Dann existiert ein EA $A = (Q, \Sigma, \delta_A, q_0, F)$, so dass $L(A) = L$.

Sei $n_0 = |Q|$ und $w \in \Sigma^*$ mit $|w| \geq n_0$. Dann ist $w = w_1 w_2 ... w_{n_0} u$, wobei $w_i \in \Sigma$ für $i = 1, ..., n_0$ und $u \in \Sigma^*$. Betrachten wir die Berechnung auf $w_1 w_2 ... w_{n_0}$:

$$(q_0, w_1 w_2 w_3 ... w_{n_0}) \underset{A}{\vdash} (q_1, w_2 w_3 ... w_{n_0}) \underset{A}{\vdash} ... \underset{A}{\vdash} (q_{n_0 - 1}, w_{n_0}) \underset{A}{\vdash} (q_{n_0}, \lambda)$$

Theorie für Nichtregularitätsbeweise - Pumping Lemma

In dieser Berechnung kommen $n_0 + 1$ Zustände $q_0, q_1, ..., q_{n_0}$ vor. Da $|Q| = n_0$, existieren $i, j \in \{0, 1, ..., n_0\}, i < j$, so dass $q_i = q_j$. Daher haben wir in der Berechnung die Konfigurationen

$$(q_0, w_1 w_2 w_3 ... w_{n_0}) \mathrel{\vdash_A^*} (q_i, w_{i+1} w_{i+2} ... w_{n_0}) \mathrel{\vdash_A^*} (q_i, w_{j+1} ... w_{n_0}) \mathrel{\vdash_A^*} (q_{n_0}, \lambda)$$

Dies impliziert

$$(q_i, w_{i+1} w_{i+2} ... w_j) \mathrel{\vdash_A^*} (q_i, \lambda) \qquad (1)$$

Wir setzen nun $y = w_1 ... w_i$, $x = w_{i+1} ... w_j$ und $z = w_{j+1} ... w_{n_0} u$, so dass $w = yxz$.

Theorie für Nichtregularitätsbeweise - Pumping Lemma

Wir überprüfen nun die Eigenschaften (i),(ii) und (iii):

(i) $yx = w_1 ... w_i w_{i+1} ... w_j$ und daher $|yx| = j \leq n_0$.

(ii) Da $|x| \geq j - i$ und $i < j$, ist $|x| \geq 1$.

(iii) (1) impliziert $(q_i, x^k) \vdash_A^* (q_i, \lambda)$ für alle $k \in \mathbb{N}$. Folglich gilt für alle $k \in \mathbb{N}$:

$$(q_0, yx^k z) \vdash_A^* (q_i, x^k z) \vdash_A^* (q_i, z) \vdash_A^* (\hat{\delta}_A(q_i, z), \lambda)$$

Wir sehen, dass für alle $k \in \mathbb{N}$ die Berechnungen im gleichen Zustand $q_{end} = \hat{\delta}_A(q_i, z)$ enden. Falls also $q_{end} \in F$, akzeptiert A alle Wörter aus $\{yx^k z \mid k \in \mathbb{N}\}$. Falls $q_{end} \notin F$, dann akzeptiert A kein Wort aus $\{yx^k z \mid k \in \mathbb{N}\}$.

$\blacksquare$

Beispielaufgabe - Pumping Lemma

Versuchen wir zu beweisen, dass

$$L_2 = \{wabw^{\mathbf{R}} \mid w \in \{a, b\}^*\}$$

nicht regulär ist.

Wir nehmen zum Widerspruch an, dass L_2 regulär ist.

Das Pumping-Lemma (Lemma 3.4) besagt, dass dann eine Konstante $n_0 \in \mathbb{N}$ existiert, so dass sich jedes Wort $w \in \Sigma*$ mit $|w| \geq n_0$ in drei Teile y, x, und z zerlegen lässt. ($\implies w = yxz$). Wobei folgendes gelten muss:

(i) $|yx| \leq n_0$

(ii) $|x| \geq 1$

(iii) **entweder** $\{yx^kz \mid k \in \mathbb{N}\} \subseteq L_2$ **oder** $\{yx^kz \mid k \in \mathbb{N}\} \cap L_2 = \emptyset$

Beispielaufgabe - Pumping Lemma

Wir wählen $w = a^{n_0}aba^{n_0}$. Es ist leicht zu sehen das $|w| = 2n_0 + 2 \geq n_0$.

Da nach (i), $|yx| \leq n_0$ gelten muss, haben wir $y = a^l$ und $x = a^m$ für beliebige $l, m \in \mathbb{N}, l + m \leq n_0$.
Somit gilt $z = a^{n_0-(l+m)}aba^{n_0}$

Nach (ii) ist $m \geq 1$.

Wir haben also $\{yx^kz \mid k \in \mathbb{N}\} = \{a^{n_0-m+km}aba^{n_0} \mid k \in \mathbb{N}\}$

Beispielaufgabe - Pumping Lemma

Da $yx^1z = a^{n_0}aba^{n_0}$ und

$a^{n_0}aba^{n_0} \in \{a^{n_0-m+km}aba^{n_0} \mid k \in \mathbb{N}\} \wedge a^{n_0}aba^{n_0} \in L_2$ gilt, folgt

$$\{a^{n_0-m+km}aba^{n_0} \mid k \in \mathbb{N}\} \cap L_2 \neq \emptyset$$

Wenn wir nun $k = 0$ wählen und uns daran erinnern, dass $m \geq 1$, erhalten wir folgendes

$$\Rightarrow yx^0z = yz = a^{n_0-m}aba^{n_0} \notin L_2$$

Daraus folgt,

$$\{a^{n_0-m+km}aba^{n_0} \mid k \in \mathbb{N}\} \not\subseteq L_2$$

Somit gilt (iii) nicht.

Dies ist ein Widerspruch! Somit haben wir gezeigt, dass die Sprache $L_2 = \{wabw^{\mathbf{R}} \mid w \in \{a,b\}^*\}$ nicht regulär ist.