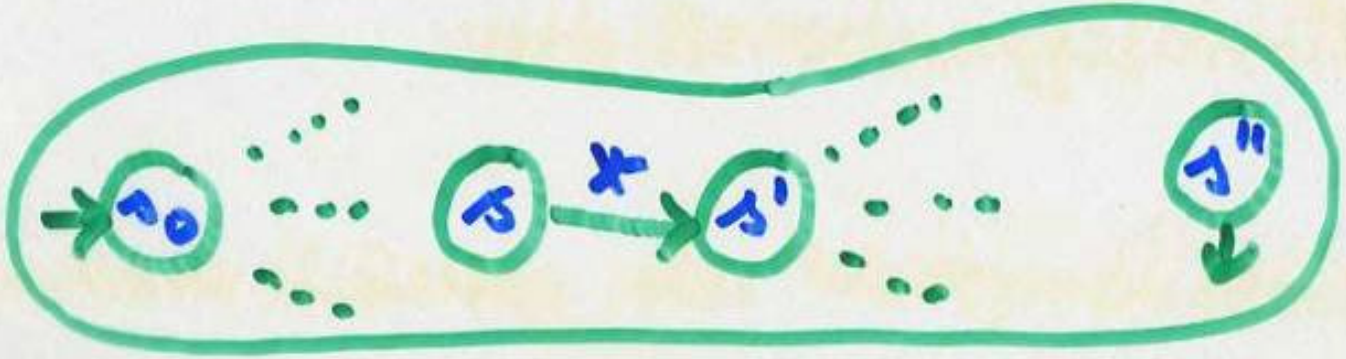
endliche Automaten

$$A = (Z, I, \delta, s_0, F)$$

- Z: endlich viele Zustände
- I: Eingaben
- δ: Zustandsüberführung
- s_0: Anfangszustand
- F: Endzustände



$s'' \in F$

$s' \in \delta(s, x)$

deterministisch vs. nichtdeterministisch

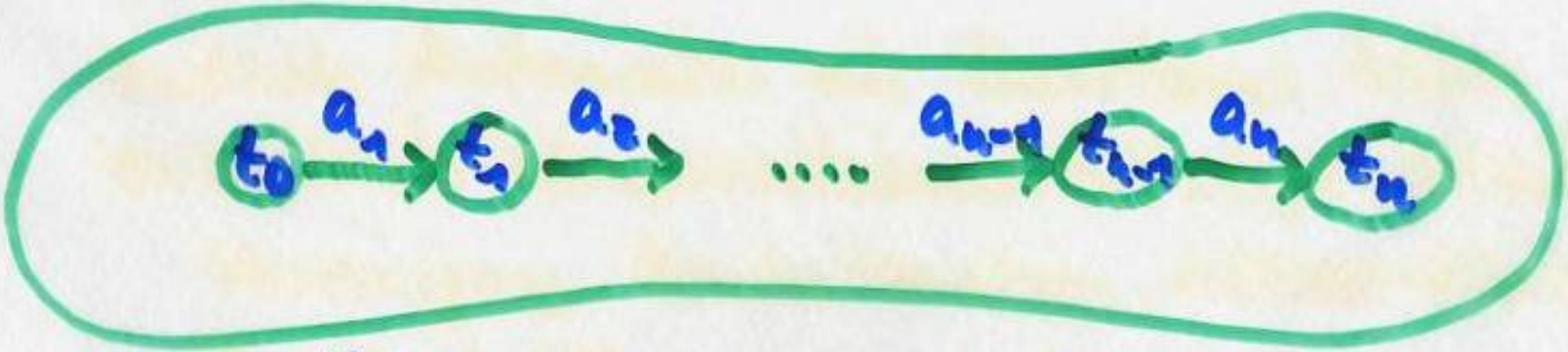
$d(s,x) \in Z$ | $d(s,x) \subseteq Z$

fortgesetzte Zustandsüberführung

$d^*: Z \times I \to Z$ Abb.: | $d^* \subseteq Z \times I \times Z$ Relation:

$d^*(s,\lambda) = s$ | $d^*(s,\lambda) = \{s\}$

$d^*(s,wx) = d(d^*(s,w),x)$ | $d^*(s,wx) = \bigcup_{t \in d^*(s,w)} d(t,x)$



$t_n \in d^*(t_0, a_1 \cdots a_n)$ gdw. $t_i \in d(t_{i-1}, a_i)$

deterministisch vs. nichtdeterministisch

erkannte Sprache $L(A) =$

$\{w \in I^* \mid d^*(s_0, w) \in F\}$ $\{w \in I^* \mid d^*(s_0, w) \cap F \neq \emptyset\}$

- $w \in L(A)$ o. $w \notin L(A)$ in $\text{length}(w)+1$ Berechnungsschritten entscheidbar

folgt so nicht aus der Definition

$\mathcal{P}(A) = (\mathcal{P}(Z), I, D, s_0, F_{\mathcal{P}}) \leftarrow \boxed{\text{power}} \leftarrow A = (Z, I, d, s_0, F)$

$D(S, x) = \bigcup_{s \in S} d(s, x)$ $\{S \subseteq Z \mid S \cap F \neq \emptyset\}$

Theorem: $L(A) = L(\mathcal{P}(A))$

Wortproblem für endliche Automaten

- entscheide algorithmisch $w \in L(A)$ oder $w \notin L(A)$
- wenn A nicht deterministisch, bilde $\mathcal{P}(A)$; löse für deterministisches A bzw. $\mathcal{P}(A)$ das Wortproblem
- $\text{length}(w)+1$ viele Berechnungsschritte (Konstruktion von $\mathcal{P}(A)$ hängt nicht von w ab)
- Lineare Lösung des Wortproblems
- praktisches Problem: $\mathcal{P}(A)$ hat exponentiell viele Zustände

Modularisierung & Kompositionalität

- große DV-Systeme werden aus kleineren Komponenten aufgebaut, die selbst aus Unterkomponenten bestehen können usw.
  - welche „Bausteine" eignen sich?
  - wie können sie zusammengesetzt werden?
  - wie wirkt sich Komponente auf Ganzes aus?

- endliche Automaten als Bausteine haben exzellente Kompositionalitätseigenschaften
  - Vereinigung, Konkatenation, Kleene-Hülle
  - Durchschnitt