Endliche Automaten und die durch sie erkennbaren Sprachen

Felix Holzwarth
felix.holzwarth @student.uni-tuebingen.de

Universität Tübingen

Abstract. In der folgenden Ausarbeitung geht es um endliche Automaten und die durch sie erkennbaren Sprachen. Diese Sprachen werden als recocnisable bezeichnet. Zu Beginn sind die endlichen Automaten definiert und einige Eigenschaften dieser dargestellt. Anschließend ist das Pumping Lemma eingeführt. Dieses ist eine Methode um für eine gegebene Sprache nachzuweisen, dass diese nicht regulär sprich recocnasible also durch endliche Automaten erkennbar ist. Dabei ist anzumerken dass das Pumping Lemma nicht die regulären Sprachen definiert, es also Sprachen gibt die das Pumping Lemma erfüllen, jedoch nicht regulär sind.
Der letzte Teil widmet sich den deterministischen endlichen Automaten und einigen Eigenschaften dieser Automaten. Es wird gezeigt das diese äquivalent zu den endlichen Automaten sind. Anders ausgedrückt bedeutet dies, dass sich jeder endliche Automat determinisieren lässt, ohne das sich die durch ihn erkannte Sprache ändert.

1 Einleitung

Die regulären Sprachen lassen sich durch unterschiedliche Formalismen beschreiben. Ein Zugang zu ihnen sind z.B. die Regulären Ausdrücke. Im Folgenden nähern wir uns den regulären Sprachen mithilfe der endlichen Automaten. Diese haben einige Eigenschaften die ebenfalls Vorgestellt sind, eine besondere Eigenschaft ist z.B die graphische Darstellbarkeit dieser Automaten

2 Endliche Automaten

2.1 Definiton

Ein endlicher Automat ist ein 5-Tupel $\mathcal{A} = (Q,\ A,\ E,\ I,F)$

- Q : endliche Menge von Zuständen
- A : Alphabet aus Zeichen
- E : Teilmenge von Q × A × Q (Transitionen)
- I : Teilmenge von Q (Initiale Zustände)
- F : Teilmenge von Q (Endzustände)

Ein endlicher Automat ist durch diese 5 endliche Mengen beschrieben. Zu Transitionen ist anzumerken das diese Aus dem Kreuzprodukt aus einem Zustand aus den Zuständen Q, einem Zeichen aus dem Alphabet A und einem Zustand aus der Menge der Zustände Q gebildet werden.

Daran lässt sich deren Charakter erkennen, eine Transition beschreibt etwas das mit zwei Zuständen und einem Zeichen zu tun hat. Sie beschreiben gerade den Übergang aus einem Zustand unter einem Zeichen in einen nächsten (evtl. auch den selben) Zustand. Dies wird in kürze durch an einem Beispiel verdeutlicht.

2.2 Graphische Darstellung

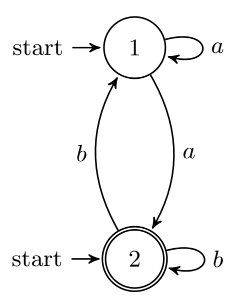

Fig. 1: Beispiel zur graph. Darstellung

Die endlichen Automaten lassen sich auf eine geschickte Art graphisch Darstellen. Dabei geht man wie folgt vor:

- Die Knoten (Kreis) repräsentieren die Zustände des Automaten
- Die Kanten repräsentieren die Transitionen
- Initiale Zustände sind durch eingehenden Pfeilen markiert
- Endzustände sind durch doppelte Kreise markiert

In 1 Ist ein Beispiel zur graphischen Darstellung eines endlichen Automaten gegeben. Dieser ist durch das 5 Tupel an Mengen definiert.

Sei $\mathcal{A} = (Q, A, E, I, F)$ mit $Q = \{1, 2\}$, $I = \{1, 2\}$, $F = \{2\}$, $A = \{a, b\}$ und $E = \{(1, a, 1), (2, b, 1), (1, a, 2), (2, b, 2)\}$

Die Zustandsmenge Q besteht aus den Zuständen 1 und 2, die Menge der Initialen Zustände besteht aus den Zuständen 1 und 2, die Menge der Endzustände besteht nur aus dem Zustand 2, das Alphabet besteht aus den Zeichen a und b und die Menge der Transitionen besteht aus den Übergängen aus Zustand 1 unter einem a zurück in den Zustand 1, aus dem Zustand 2 unter einem b in den Zustand 1, aus Zustand 1 unter einem a in den Zustand 2 und aus dem Zustand 2 unter einem b zurück in den Zustand 2.

2.3 Eigenschaften endlicher Automaten

Nun sind einige Eigenschaften endlicher Automaten betrachtet. Diese Begrifflichkeiten und Eigenschaften erleichtern im folgenden über bestimmte Themen im Zusammenhang mit den endlichen Automaten zu sprechen.

2 Transitionen (p, a, q) und (p', a', q') heißen **aufeinanderfolgend** wenn $q = p'$. Der Zustand in den der Automat unter dem ersten Zeichen gegenangen ist muss mit dem ersten Zustand der betrachteten Transition übereinstimmen.

Ein **Pfad** im Automat $\mathcal{A}$ ist eine endliche Folge aus aufeinanderfolgenden Transitionen. Ein Pfad kann auf 2 verschiedene Arten notiert werden.

Die eine Art benutzt die Tupel in der die einzelnen Transitionen definiert sind direkt:

$c = (q_0, a_1, q_1),(q_1, a_2, q_2),\ ....\ ,\ (q_{n-1}, a_n, q_n)$

Die andere Art ist verkürzt, hierbei spart man sich den einen Zustand der in beiden Transitionen gleich sein muss doppelt zu notieren:

$c : q_0 \xrightarrow{a_1} q_1 \cdots q_{n-1} \xrightarrow{a_n} q_n$

Der Zustand q_0 heißt **Ursprung** und q_n heißt **Ende**, $a_1 \cdots a_n$ ist das **Label** und n die **Länge** des Pfades

Nun betrachten wir einige Eigenschaften von Pfaden selbst. Ein **Pfad** in $\mathcal{A}$ heißt **initial** wenn sein *Ursprung* ein initiale Zustand ist und **final** wenn sein *Ende* ein Endzustand ist

Ein **Pfad** heißt **akzeptierend** wenn er *initial* und *final* ist

Pfade und deren Eigenschaften in einem Automaten sind im folgenden Beispiel nocheinmal verdeutlicht

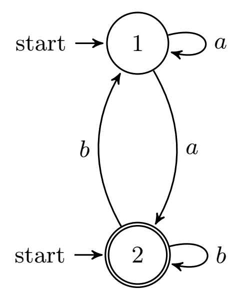

Fig. 2: Vorheriger Automat

Der Pfad in 2 $c_1 : 1 \xrightarrow{a} 2 \xrightarrow{b} 2 \xrightarrow{b} 1 \xrightarrow{a} 2 \xrightarrow{b} 2$ ist akzeptierend da er initial und final ist, sprich sein Urpsrung ist ein initialer Zustand und sein Ende ein Endzustand.
Hingegen ist der Pfad in 2 $c_2 : 1 \xrightarrow{a} 2 \xrightarrow{b} 2 \xrightarrow{b} 1 \xrightarrow{a} 2 \xrightarrow{b} 1$ nicht akzeptierend da er nicht final ist, er also nicht in einem Endzustand endet. Es fällt auf das beide Pfade das selbe Label besitzten. Ein Label kann in einem endlichen Automaten mehrere Pfade definieren.

Ein Zustand q heißt **erreichbar** wenn es einen *initialen Pfad* gibt der in q *endet*. Ein Zustand q ist **coerreichbar** wenn ein *finaler Pfad* existiert der in Zustand q startet.

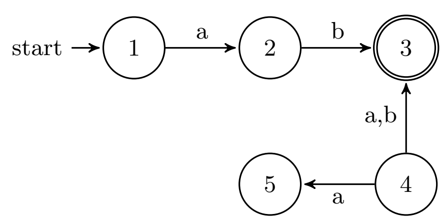

Fig. 3: Automat mit erreichbaren und coerreichbaren Zuständen

Der Zustand 2 oder der Zustand 3 in 3 sind *erreichbar* da in beiden Fällen ein initialer Pfad existiert, der jeweils in den Zuständen endet. Der Zustand 4 in 3 ist *coerreichbar* da ein finaler Pfad existiert der in 4 startet. Zustand 5 in 3ist weder *erreichbar* noch *coerreichbar*

Nun können wir die akzeptierte Sprache $\mathcal{L}$ eines Automaten definieren. Ein Wort w über A wird durch den Automaten $\mathcal{A}$ *akzeptiert*, wenn w das *Label* mindestens eines *akzeptierendes Pfades* ist.

Die **Sprache die von $\mathcal{A}$ erkannt wird** ist Menge aller Wörter die durch $\mathcal{A}$ erkannt werden. Ausgedrückt durch $\mathcal{L}(\mathcal{A})$. Eine Sprache $\mathcal{L}$ heißt **recognisable** wenn sie durch einen endlichen Automaten $\mathcal{A}$ erkannt wird so dass : $\mathcal{L} = \mathcal{L}(\mathcal{A})$. Zwei Automaten sind **äquivalent** wenn sie die gleiche *Sprache* $\mathcal{L}$ erkennen.

3 Das Pumping Lemma

Eine Möglichkeit nachzuweisen das eine Sprache nicht regulär ist, sie also nicht durch einen endlichen Automaten erkannt werden kann, ist das Pumping Lemma.

3.1 Definition

Sei $\mathcal{L}$ eine *recognisable* Sprache. Dann $\exists n > 0 : \forall$ Wörter $u \in \mathcal{L}$ mit $|u| \geq n$ gilt dass sie sich faktorsieren lassen als $u = xyz$, mit $x, y, z \in A^*$, $|xy| \leq n$, $|y| \geq 1$ und $\forall k \geq 0$ $xy^k z \in \mathcal{L}$

In Worten ausgedrückt bedeuted dies, dass für jede reguläre Sprache $\mathcal{L}$ ein n existiert, so dass für jedes Wort w aus $\mathcal{L}$ der Länge $|w| \geq n$ gilt, das es eine Zerlegung in Teilworte $w = xyz$ hat die die obigen drei Eigenschaften erfüllen:

- Der Betrag $|xy|$ des Teilwortes xy ist $\leq n$
- Das Teilwort y ist nicht das leere Wort, es besteht aus mind. aus einem Zeichen
- $\forall k \geq 0 : xy^k z \in \mathcal{L}$

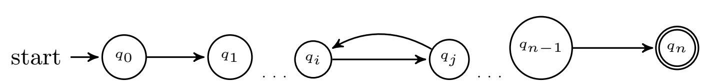

Fig. 4: Idee hinter dem Pumping Lemma

Die Idee hinter dem Pumping Lemma, (auch Schleifensatz) ist in 4 verdeutlicht. Da jeder Automat $\mathcal{A}$ nur endlich viele Zustände n Besitzt, gilt also ab einem Bestimmten n das die Anzahl der verschiedenen Zustände beschreibt, das ein Wort aus der Sprache $\mathcal{L}$ dessen Betrag größer als n ist, mind. $n+1$ Zustände Benötigt. Dies Bedeuted dass das Wort mind. einen Zustand doppelt benutzen muss und dadurch eine Schleife in den Zuständen bildet.

3.2 Beweis des Pumping Lemma

Sei $\mathcal{A} = (Q, A, E, I, F)$ ein Automat mit n Zuständen also $|Q| = n$ der $\mathcal{L}$ erkennt. Nach dem Pumping Lemma existiert ein n so das für Alle Worte der länge mindestens n die oben genannten Eigenschaften des Pumping Lemmas gelten.

$$\text{Sei } u = a_1, \cdots, a_r \text{ ein Wort} \in \mathcal{L} \text{ mit } r \geq n.$$

Sei $c : q_0 \xrightarrow{a_1} q_1 \cdots q_{r-1} \xrightarrow{a_r} q_r$ ein *akzeptierender* Pfad der mit u *gelabelt* ist.

$$\text{Da } r \geq n \; \exists \; i, \; j : i < j \leq n \text{ so dass } q_i = q_j$$

Aus diesem Grund ist das Teilwort $a_{i+1} \cdots a_j$ das *Label* einer Schleife um den Zustand q_i

Ist nun $x = a_1 \cdots a_i$, $y = a_{i+1} \cdots a_j$, $z = a_{j+1} \cdots a_r$ dann ist $|xy| \leq n$ und $\forall k \geq 0$ gilt $xy^k z \in \mathcal{L}$, da $xy^k z$ das *label* eines *akzeptierenden* Pfades ist. qed

3.3 Beispiel $\{a^n b^n | n \geq 0\}$

Sei $\mathcal{L} = \{a^n b^n | n \geq 0\}$ Angenommen $\mathcal{L}$ sei duch einen endl. Automaten *erkennbar*.

Nach Pumping lemma $\exists n > 0$ so dass jedes Wort w aus $\mathcal{L}$ der Länge $\geq n$ w eine Zerlegung in Teilworte in $w = xyz$ hat für die die Eigenschaften des Pumping Lemmas gelten. Sei n gleich dem n aus $a^n b^n$. Damit hat das Wort $a^n b^n$ die Länge $2n$. Es folgt auch dass das Teilwort xy nur $\acute{a}s$ enthalten kann. Nach dem Pumping Lemma muss nun eine Zerlegung existieren so dass jedes Wort geschrieben werden kann als: $a^n b^n = xyz$ mit x, y, $z \in A^*$, $|xy| \leq n$, $|y| \geq 1$ und $xy^2 z \in \mathcal{L}$. (nur schwaches Kriterium des Pumping lemma)

Da $|xy| \leq n$, ist das Wort xy ein präfix von a^n und es gilt $x = a^r$, $y = a^s$, $z = a^t b^n$ mit $\underline{r + s + t = n}$, weiter gilt nach Vorraussetzung $s \neq 0$

Betrachten wir $xy^2 z = a^r a^{2s} a^t b^n$, gilt da $s \neq 0$ $\underline{r + 2s + t \neq n}$ und somit $xy^2 z \notin \mathcal{L}$. Damit erhalten wir einen Widerspruch zu anfänglichen Annahme. qed

3.4 Bemerkung zum Pumping Lemma

Durch das Pumping Lemma sind nicht die regulären Sprachen und damit auch die Sprachen welche durch endliche Automaten erkannt werden, definiert. Es gilt also ist eine Sprache regulär, dann erfüllt sie das Pumping Lemma. Es gilt aber nicht das jede Sprache die das Pumping Lemma erfüllt, eine Reguläre Sprache ist.In vielen Fällen ist das Pumping Lemma jedoch ein guter Weg um bei einer Sprache zu zeigen das sie nicht regulär und damit durch einen endlichen Automaten akzeptierbar ist.

4 Deterministische endliche Automaten

Die bisher betrachteten Automaten haben einen Nachteil. Wenn der Automat das Wort akzeptiert, dann wissen wir dass das dem Automaten gegebene Wort sich in der Sprache des Automaten befindet. Beim Ablehnen wissen wir nicht ob es nicht doch einen akzeptierenden Pfad im Automaten gibt, der im letzten Lauf des Automaten nicht genommen wurde. An dieser Stelle wird das Konzept der deterministischen Automaten Sinnvoll. Bei dieser Form eines Automaten können wir dem Ablehnen so wie dem Akzeptieren vetrauen.

4.1 Definition

Ein Automat $\mathcal{A} = (Q,A,E,I,F)$ ist **deterministisch** wenn:

- $|I| = 1$
- $\forall q \in Q \ \forall a \in A : \exists$ genau ein $q' : q \xrightarrow{a} q' \in E$
- Der Zustand q' wird nun mit $q \cdot a$ notiert
- Ist q_- der initial Zustand schreiben wir $\mathcal{A} = (Q,A,\cdot,q_-,F)$

Im deterministischen Fall, darf die Menge der initialen Zustände nur einen Zustand enthalten. Es fällt auf das die Zuordnung zu einem Zustand q und einem Zeichen a jetzt eindeutig bestimmt ist. Daher beschreiben die Transitionen durch diese eindeutigkeit jetzt eine Funktion. An diese Punkt befindet sich auch der determinismus, es darf nun keine Transitionen zu dem selben Zustand geben, die unter dem selben Zeichen a in zwei verschiedene Zustände führen.

4.2 Beispiel zu einem deterministischen Automaten

In 5 ist ein deterministischer Automat gegeben. Aus jedem Zustand ist bei diesen Automaten eindeutig in welchen Zustand er unter einem Zeichen überführt wird.

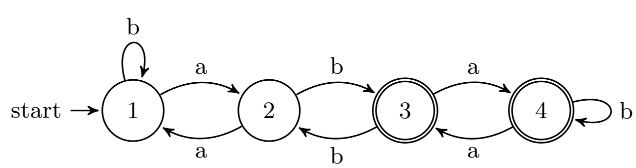

Fig. 5: Beispiel zu einem deter. Automat

4.3 Äquivalenz zu endl. Automaten

Ein sehr wichtiges Ergebniss im Zusammenhang mit den endlichen und deterministischen Automaten die Äquivalenz dieser Beiden Konzepte. Dies bedeuted, dass sich jeder endliche Automat determinisieren lässt ohne das sich bei seiner akzeptierten Sprache etwas ändert. In dem vorgeführten Beweis wird die Methode der Potenzmengen Konstruktion verwendet. Diese Konstruktion soll zuerst an einem Beispiel vorgeführt werden, anschließend wird die Aussage bewiesen. Bevor wir das Beispiel betrachten, folgen kurz die Definitionen der neuen Mengen des deterministischen Automaten:

Sei $\mathcal{A} = (Q,A,E,I,F)$ der endliche Automat von dem wir zu Beginn ausgehen. D($\mathcal{A}$) = ($\mathcal{P}$(Q),A,$\cdot$,I,$\mathcal{F}$) *($\mathcal{P}(Q)$ ist die Potenzmenge von Q)* ist die determinisierte Version von $\mathcal{A}$.

$$\mathcal{F} = \{P \subseteq Q | P \cap F \neq \emptyset\}$$

$$\forall\ P \in \mathcal{P}(Q),\ \forall\ a \in A : P \cdot a = \{q \in Q \mid \exists\ p \in P : (p, a, q) \in E\}$$

Die neuen Endzustände $\mathcal{F}$ sind diejenigen Elemente P aus der Potenzmenge der Zustände $\mathcal{P}(Q)$ die unter dem Schnitt der Mengen mit der alten Menge der Endzuständen F eine nicht leere Menge ergeben. Dies bedeuted der neue Endzustand bestehend aus Elementen der alten Zustandsmenge Q enthält mindestens einen Zustand der sich auch in der Menge der alten Endzustände befunden hat.

Die neue Menge der Transitionen für die Elemente P (welche selbst eine Menge alter Zustände p darstellen), unter einem Zeichen a , sind diejenigen Zustände q für die es in der alten Menge der Transitionen jeweils eine Transition unter einem p unter a in einen Zustand q gibt.
Die neuen Zustände aus der Potenzmenge $\mathcal{P}(Q)$ sind so zu interpretieren das wenn der neue deterministische Automat sich in einem Zustand $\{q_1, q_2\}$ befindet, befand sich der alte Automat in Zustand q_1 oder Zustand q_2.

4.4 Beispiel Potenzmengen Konstruktion

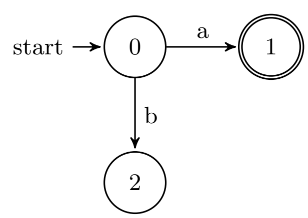

Fig. 6: Nichtdeter. Automat

Der in 6 gegebene Automat hat die Zustandsmenge $Q = \{0, 1, 2\}$, die Potenzmenge davon ist gegeben durch $\mathcal{P}(Q) = \{\{\}, \{0\}, \{1\}, \{2\}, \{0, 1\}, \{1, 2\}, \{0, 2\}, \{0, 1, 2\}\}$.

Der determinisierte Automat D($\mathcal{A}$) in abb. 7 besteht nun aus den Zuständen aus der Potenzmenge $\mathcal{P}(Q)$. Ausgehend von Zustand der Menge $\{0\}$ überprüfen wir in welche alten Zustände q in Q wir aus dem Zustand 0 unter einem Zeichen a gelangen. Dies ist die Menge aus dem Zustand $\{1\}$. Ähnlich gilt für Zustand $\{0\}$ unter dem Zeichen b, das wir in den Zustand aus der Menge $\{2\}$ gelangen. Aus den Zustand der Mengen $\{1\}$ und $\{2\}$ gelangen wir jeweils unter den Zeichen a und b in den Zustand der leeren Menge $\{\}$. Dies stellt den Totzustand dar. Unter den Zeichen a und b verbleiben wir in diesem. Nun haben wir Alle erreichbaren Zustände überprüft. In der Potenzmenge existieren jedoch noch weitere Zustände. Diese verhalten sich in diesem Beispiel Alle gleich, daher ist nur einer von ihnen exemplarisch dargestellt und Besprochen. Aus dem Zustand der Menge $\{1, 2\}$ erreichen wir unter einem Zeichen a den Zustand aus der Menge $\{1\}$ und unter einem Zeichen b den Zustand der Menge $\{2\}$.

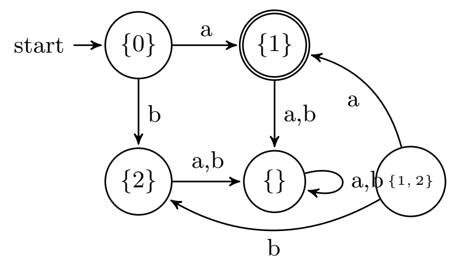

Fig. 7: Determinisierter Automat aus 6

Beweis der Äquivalenz

In dem Beweis zeigen wir einmal die Richtung, dass der endliche Automat ein Wort akzeptiert und beweisen dann, dass dieses Wort auch durch den determinisierten Automaten akzeptiert wird. Umgekehrt wird dann gezeigt, dass wenn der determinisierte Automat ein Wort erkennt, dies auch der endliche Automat akzeptiert.

Sei $u = a_1 \cdots a_n$ das Label eines akzeptierenden Pfades des endlichen Automaten $\mathcal{A}$. Diesen können wir als

$$C : q_0 \xrightarrow{a_1} q_1 \cdots q_{n-1} \xrightarrow{a_n} q_n \text{ notieren.}$$

Das Label u definiert ebenfalls einen Pfad in dem determinisierten Automaten $\mathcal{P}(Q)$:

$$\mathcal{C} : \mathrm{I} = P_0 \xrightarrow{a_1} P_1 \cdots P_{n-1} \xrightarrow{a_n} P_n$$

Hierbei sind die P_i Zustände aus der Potenzmenge $\mathcal{P}(Q)$. Mithilfe einer Induktion über i kann man zeigen, das für $0 \leq i \leq n$, $\underline{q_i \in P_i}$, das also der Pfad in der determinisierten Version ebenfalls ein akzeptierender Pfad ist, da jeder einzelne alte Zustand q_i nun in der neuen Zustandsmenge P_i, bestehend aus evtl. mehreren alten Zuständen, enthalten ist.

Induktions Anfang: Da c ein akzeptierender Pfad ist (muss in initialem Zustand beginnen) gilt $q_0 \in I = P_0$

Induktions Annahme: $q_{i-1} \in P_{i-1}$

Induktions Schluss: Da $q_{i-1} \xrightarrow{a_i} q_i$ eine Transition ist, gilt:

$$q_i \in \underbrace{P_{i-1} \cdot a_i}_{P_{i-1} \cdot a_i = \{q \in Q \mid \exists p \in P_{i-1} : (p,a,q) \in E\}} = P_i$$

Im Induktionschluss benutzen wir zum einen das $q_{i-1} \in P_{i-1}$ gilt und das $q_{i-1} \xrightarrow{a_i} q_i$. Nun benutzen wir die Eigenschaft der Definition der Übergänge unter einem Zeichen a_i eines Zustandes P_i. Dies sind Alle q für die es aus einem Zustand $p_i \in P_i$ in der alten Menge der Transition E gerade eine Transition unter p_i unter dem Zeichen a_i in den Zustand q gab, daher gilt $q_i \in P_i$

Für $i = n$ erhalten wir auf diese weise $q_n \in P_n$ und da c ein akzeptierender Pfad ist gilt $q_n \in F$ (F sind alten Entzustände)

Es folgt also das P_n ein Endzustand aus F enthält, und daher gilt $P_n \in \mathcal{F}$ Daher akzeptiert D($\mathcal{A}$) das Wort u.

Sei andersherum $u = a_1 \cdots a_n$ ein Wort das durch determinisierten Automaten D($\mathcal{A}$) akzeptiert wird und dadurch einen akzeptierenden Pfad definiert.

Da P_n ein Endzustand ist existiert ein $q_n \in P_n \cap F$ der den Endzustand des Pfades in $\mathcal{A}$ definiert.

Nun können wir für $i = n, n-1, ..., 1$ ein Element $q_{i-1} \in P_{i-1}$ auswählen und damit eine passende Transition $q_{i-1} \xrightarrow{a_i} q_i$ in $\mathcal{A}$ bauen. Dies funktioniert da wir den Pfad durch die Potenzmengenzustände zuvor so ausgewählt haben das evtl nicht jeder alte Zustand q_{i-1} infrage kommt, dass es jedoch mindestens einen gibt der die Bedingung erfüllt

Da $q_0 \in I$ und $q_n \in F$ ist der Pfad $q_0 \xrightarrow{a_1} q_1 \cdots q_{n-1} \xrightarrow{a_n} q_n$ ein akzeptierender Pfad in $\mathcal{A}$. qed

4.5 Eigenschaften deterministischer endlicher Automaten

Ein Automat $\mathcal{A} = (\mathrm{Q},\mathrm{A},\cdot,q_-,\mathrm{F})$ ist **vollständig** wenn gilt

$$\forall q \in Q \wedge \forall a \in A \; \exists \text{ mindestens ein } q' : q \xrightarrow{a} q'$$

Ein Automat ist **erreichbar / coerreichbar** wenn Alle seine Zustände *erreichbar / coerreichbar* sind

Ein Automat ist **trim**, wenn alle seine Zustände *erreichbar* und *coerreichbar* sind.

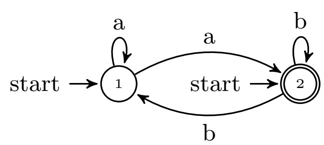

Fig. 8: Beispiel eines nicht vollständigen Automaten

Der in 8 dargestellte endliche Automat ist nicht vollständig, da es aus Zustand 1 keine Transition für das Zeichen b gibt. Der in 9 dargestellte Automat

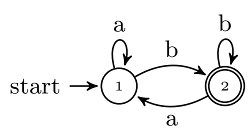

Fig. 9: Beispiel eines vollständigen Automaten

ist vollständig, da es für alle Zeichen und jeden Zustand eine Transition gibt. Der in 10 gegebene Automat ist nur erreichbar da ein initialer Pfad zu allen Endzuständen existiert, jedoch kein finaler Pfad aus Zustand 1 existiert.

Der in 11 gegeben Automat ist coerreichbar da für alle Zustände gilt, dass sie finale Pfade zu initialen Zustand sind.

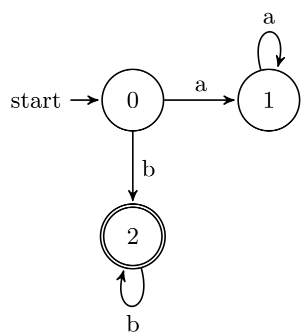

Fig. 10: Beispiel eines erreichbaren Automaten

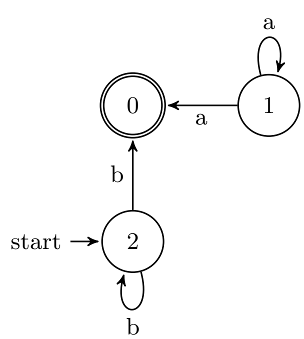

Fig. 11: Beispiel eines coerreichbaren Automaten

5 Zusamenfassung

Endliche Automaten erkennen die als recocnisable bezeichneten Sprachen. Sie bieten einen alternativen Zugang zu den regulären Sprächen über das Modell der Automaten. Jeder endliche Automat lässt sich mit maximal exponentiell viel mehr Aufwand in eine deterministische Variante umsetzen. Die durch den Automaten erkannte Sprache ändert sich dabei nicht.