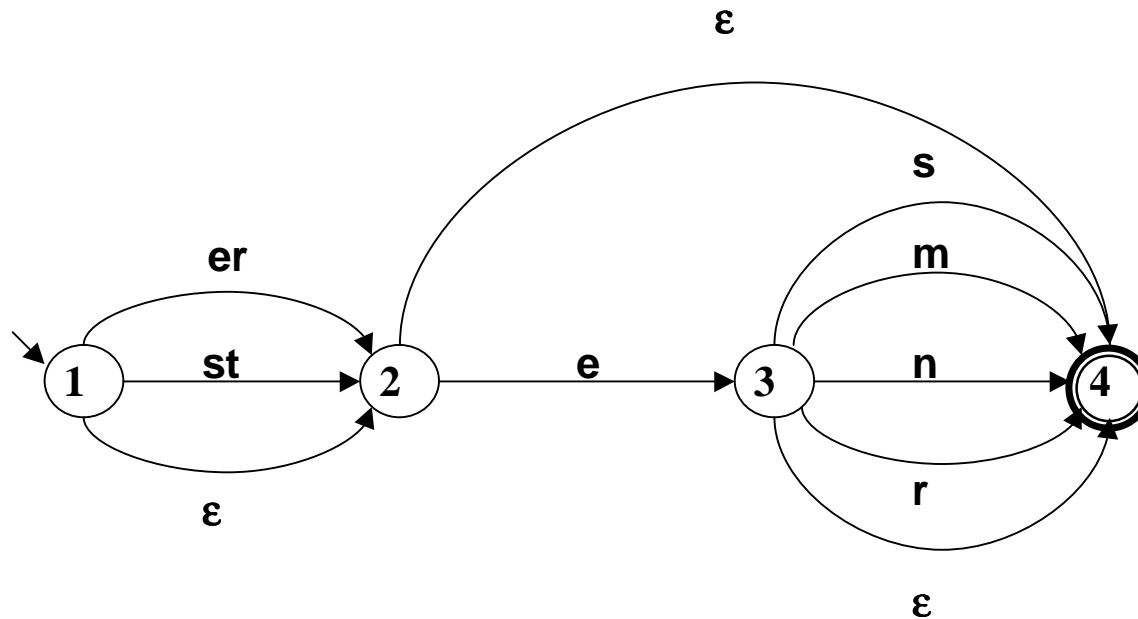


Die NEA-DEA-Überführung

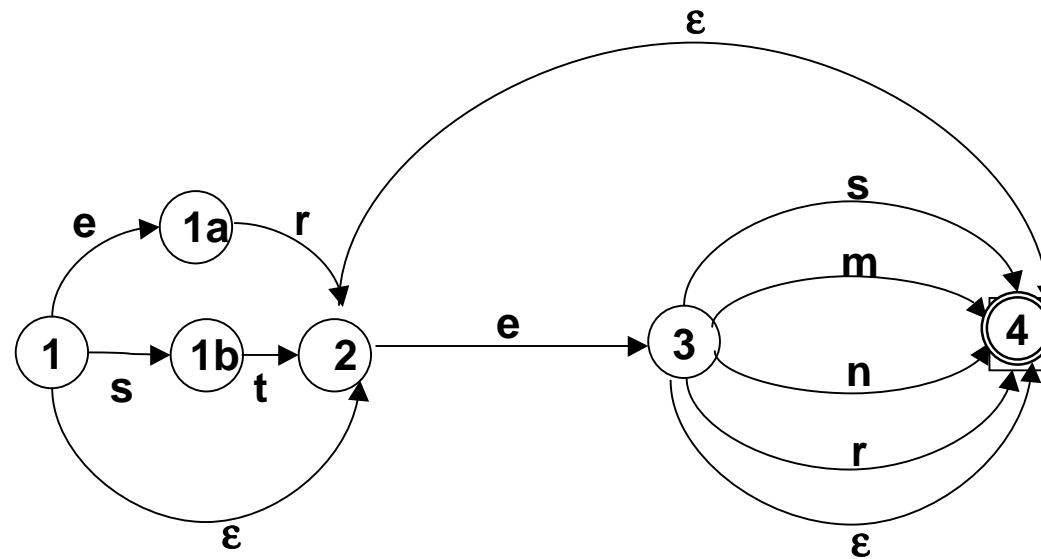
Der Algorithmus zur NEA-DEA-Überführung besteht aus drei Schritten:

1. Beseitigung von Mehrsymbol-Kanten
2. Beseitigung von ε -Kanten
3. Die „Potenz-Automaten“-Konstruktion

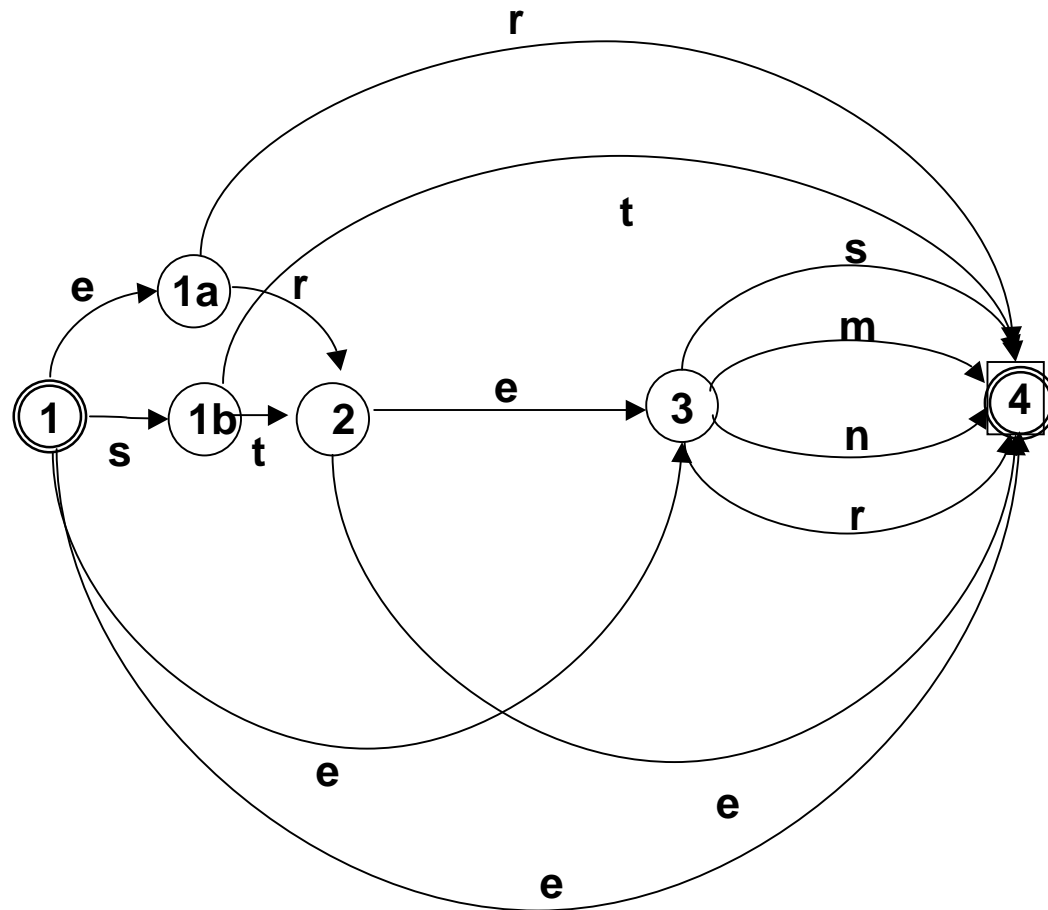
Adjektivendungen: Zustandsdiagramm



Beispiel-Automat nach Schritt 1:



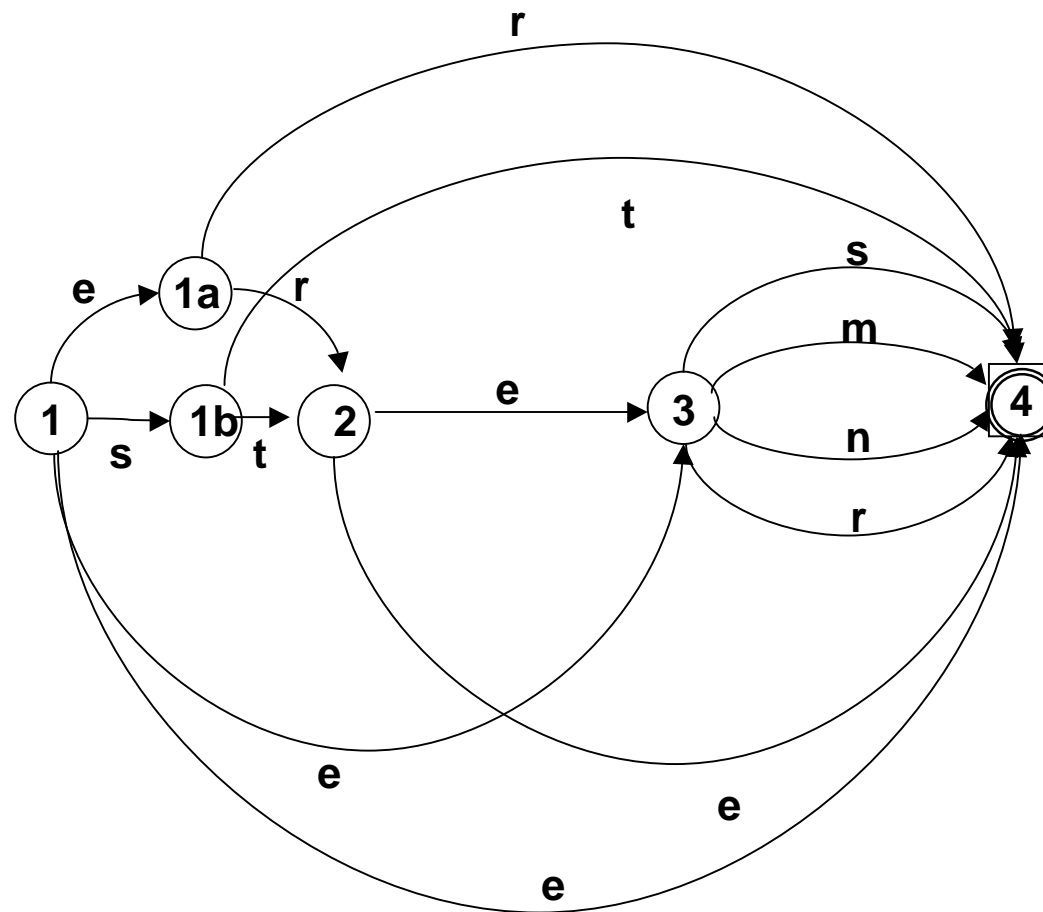
Schritt 2: Beseitigung von ϵ -kanten: Resultat ist „buchstabierender Automat“



Schritt 3: Potenzautomaten-Konstruktion, Vorüberlegung

- Wir haben einen Algorithmus zur Pfadsuche am Beispiel des unbearbeiteten Adjektivendungs-Diagramms kennengelernt: „Tiefensuche mit Backtracking“. Durch die Organisation der Agenda als Stapel/Stack („last in – first out“) wird eine Alternative so weit wie möglich verfolgt; bei endgültigem Scheitern wird das System zurückgesetzt.
- Durch die Organisation der Agenda als Warteschlange (queue), bei der die Aufgaben in der Reihenfolge ihrer Generierung abgearbeitet werden („first in – first out“), erhalten wir Breitensuche. Die alternativen Pfade werden (quasi) parallel verfolgt.

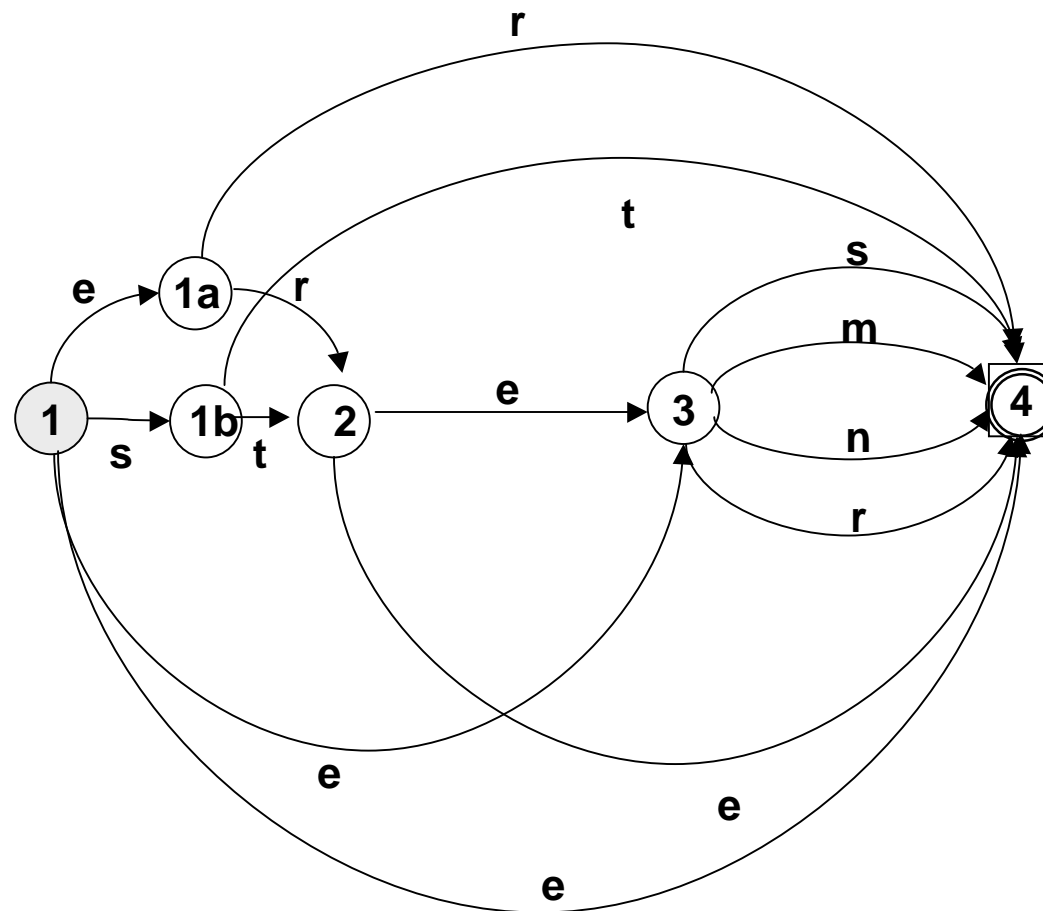
Pfadsuche als Breitensuche



Eingabewort:

Agenda: 1 -- klein eres

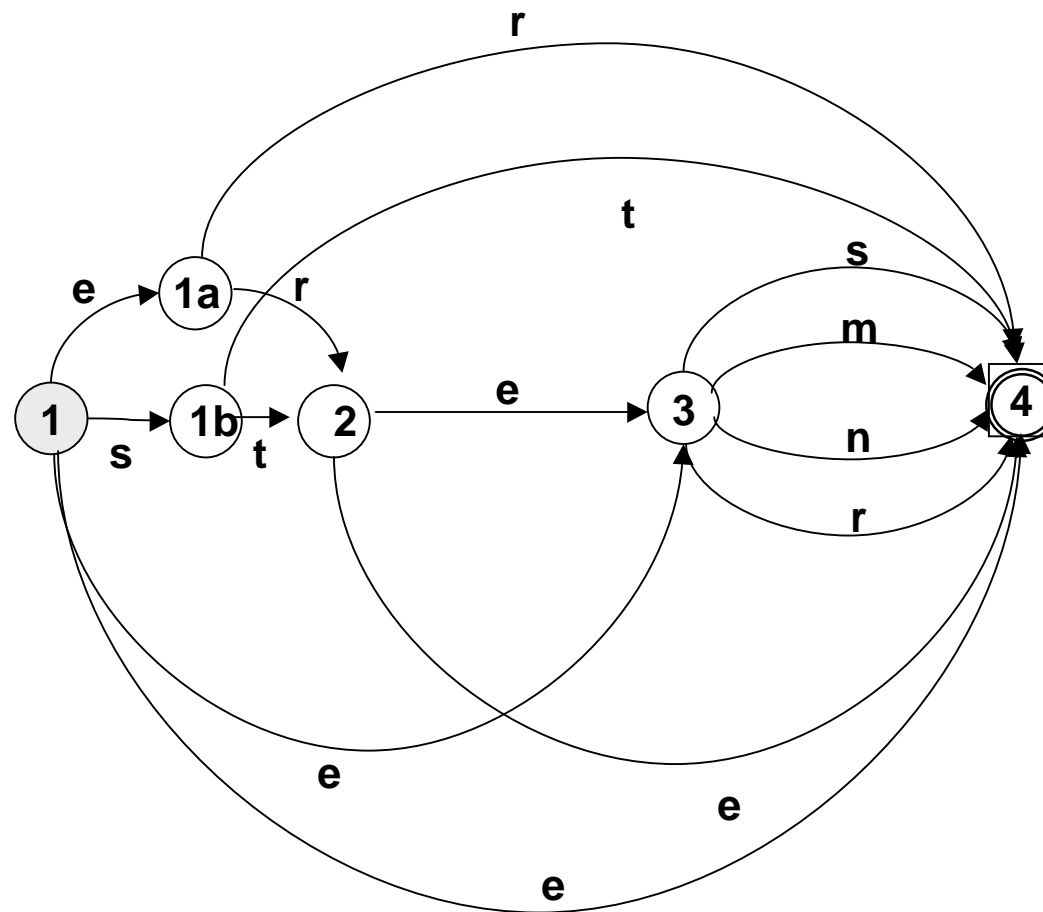
Pfadsuche als Breitensuche



Eingabewort: klein eres

Agenda: _____

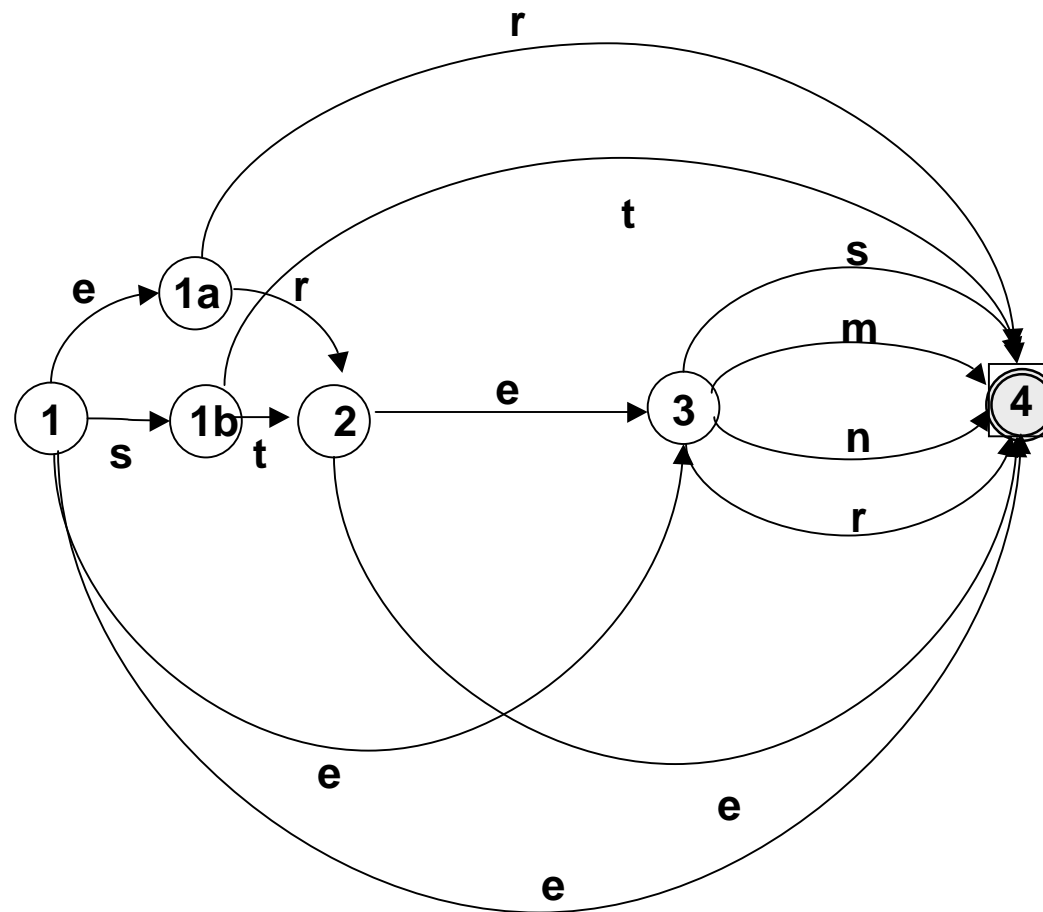
Pfadsuche als Breitensuche



Eingabewort: klein eres

1a -- klein eres
 3 -- klein eres
 Agenda: 4 -- klein eres

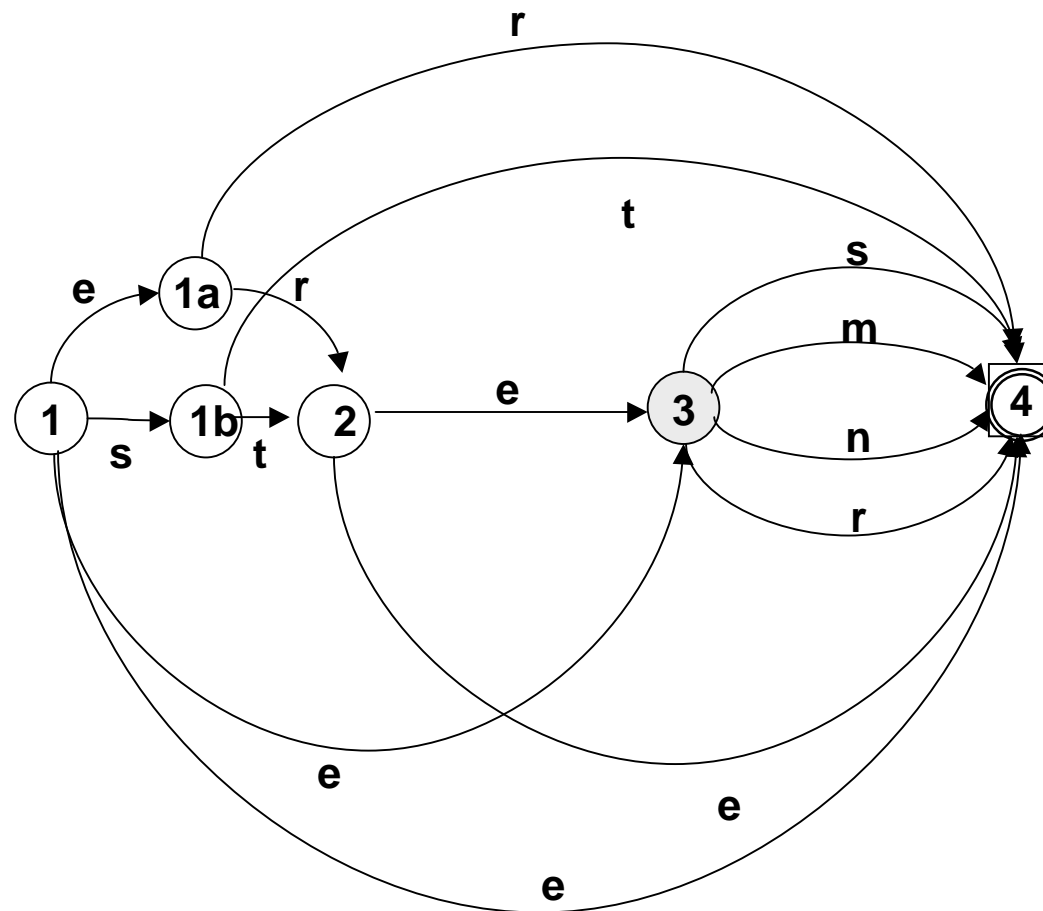
Pfadsuche als Breitensuche



Eingabewort: klein eres

Agenda: 3 -- klein eres

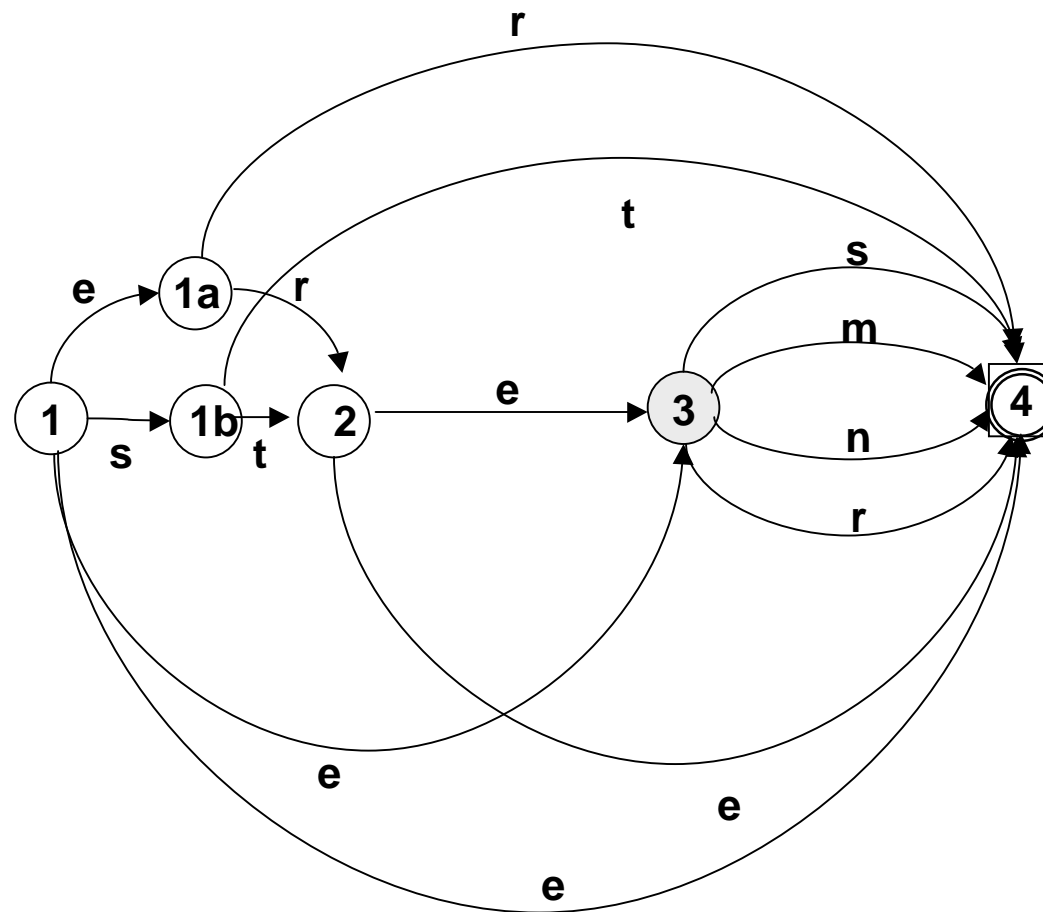
Pfadsuche als Breitensuche



Eingabewort: klein eres

Agenda: 1a -- klein eres

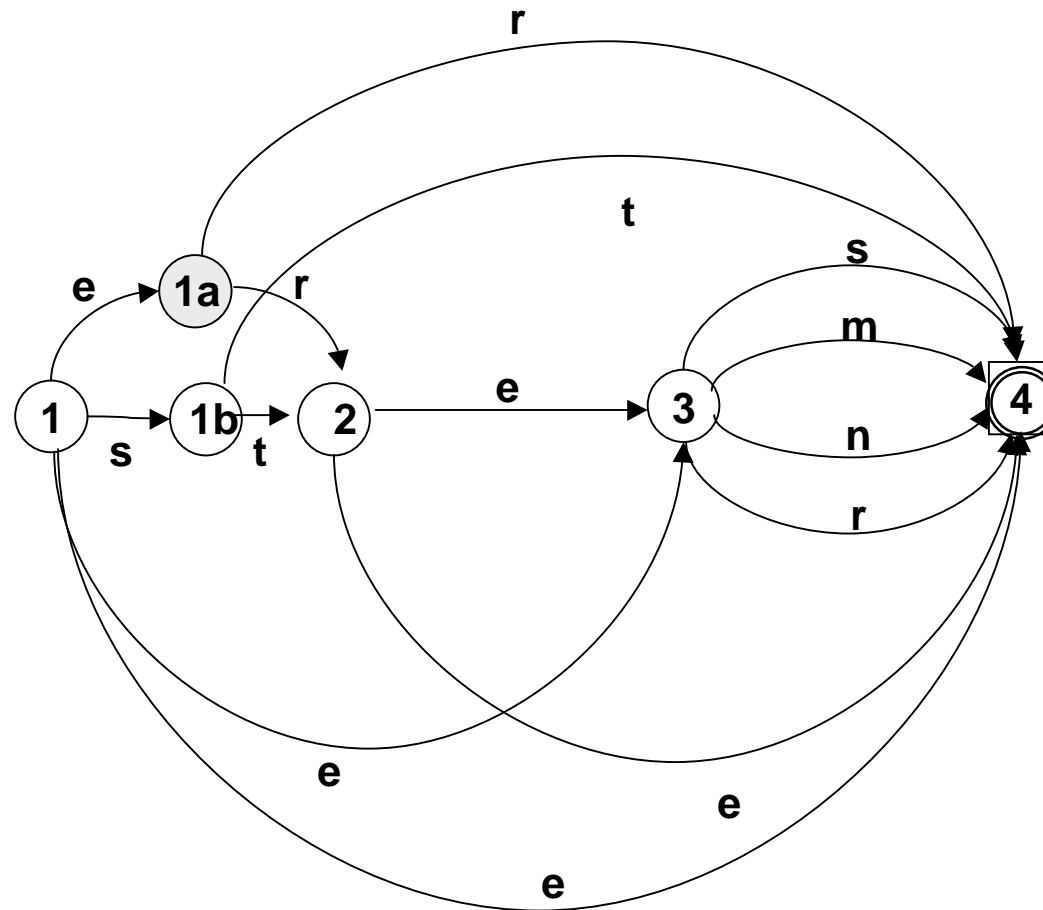
Pfadsuche als Breitensuche



Eingabewort: klein eres

Agenda: 1a -- klein eres

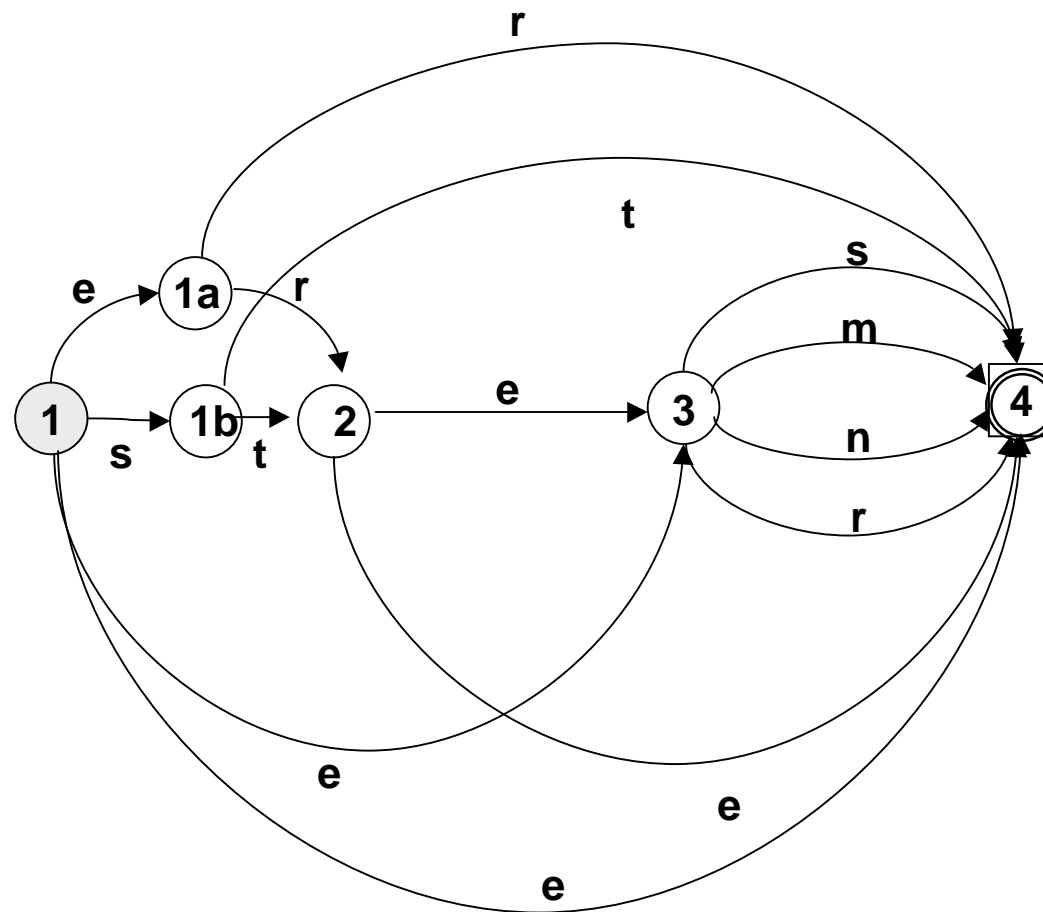
Pfadsuche als Breitensuche



Eingabewort: klein eres

2 -- klein eres
 4 -- klein eres
 Agenda: 4 -- klein eres

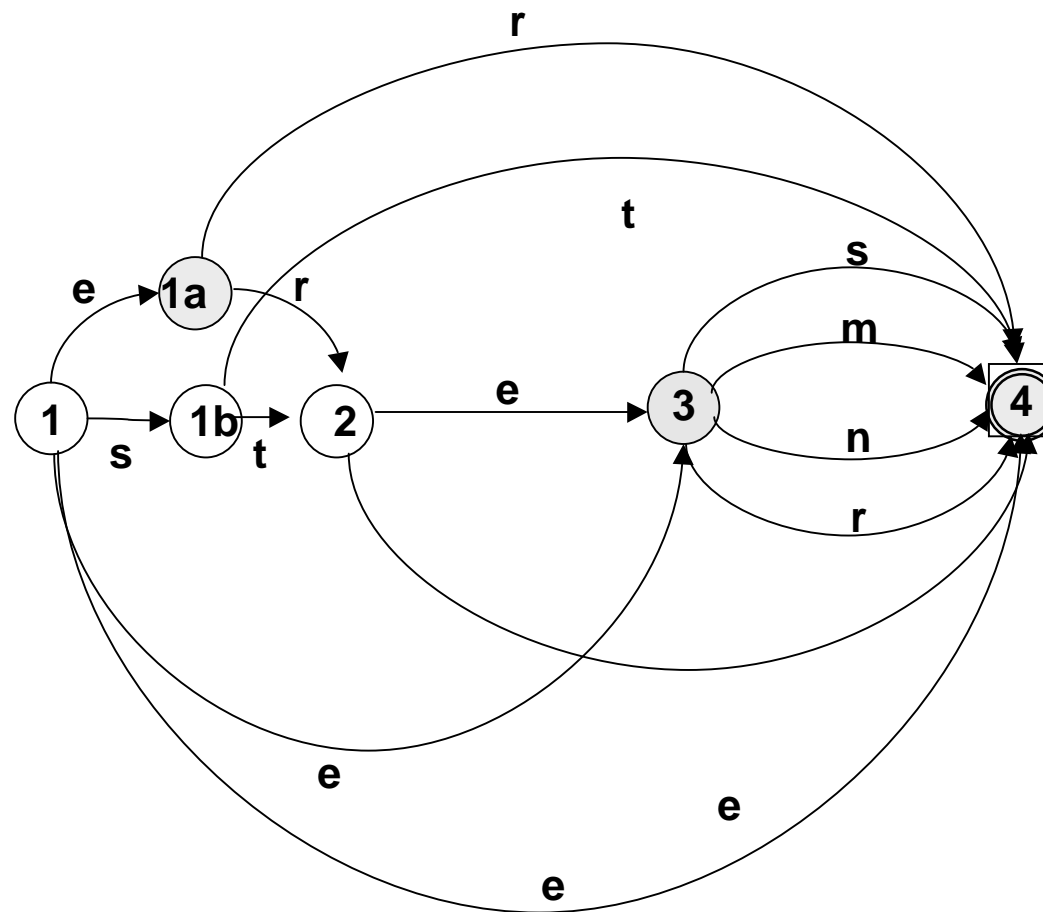
Pfadsuche als Breitensuche



Eingabewort: klein eres

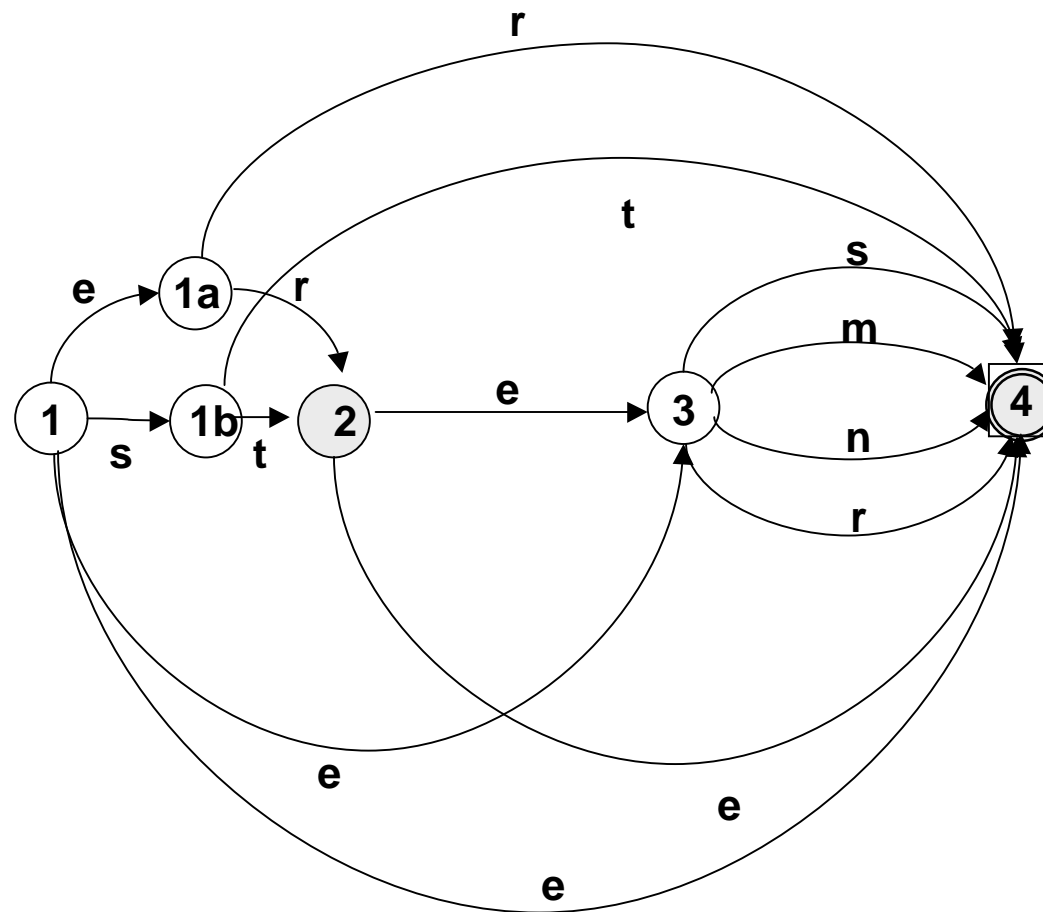
1a -- klein eres
 3 -- klein eres
 Agenda: 4 -- klein eres

Pfadsuche als Breitensuche



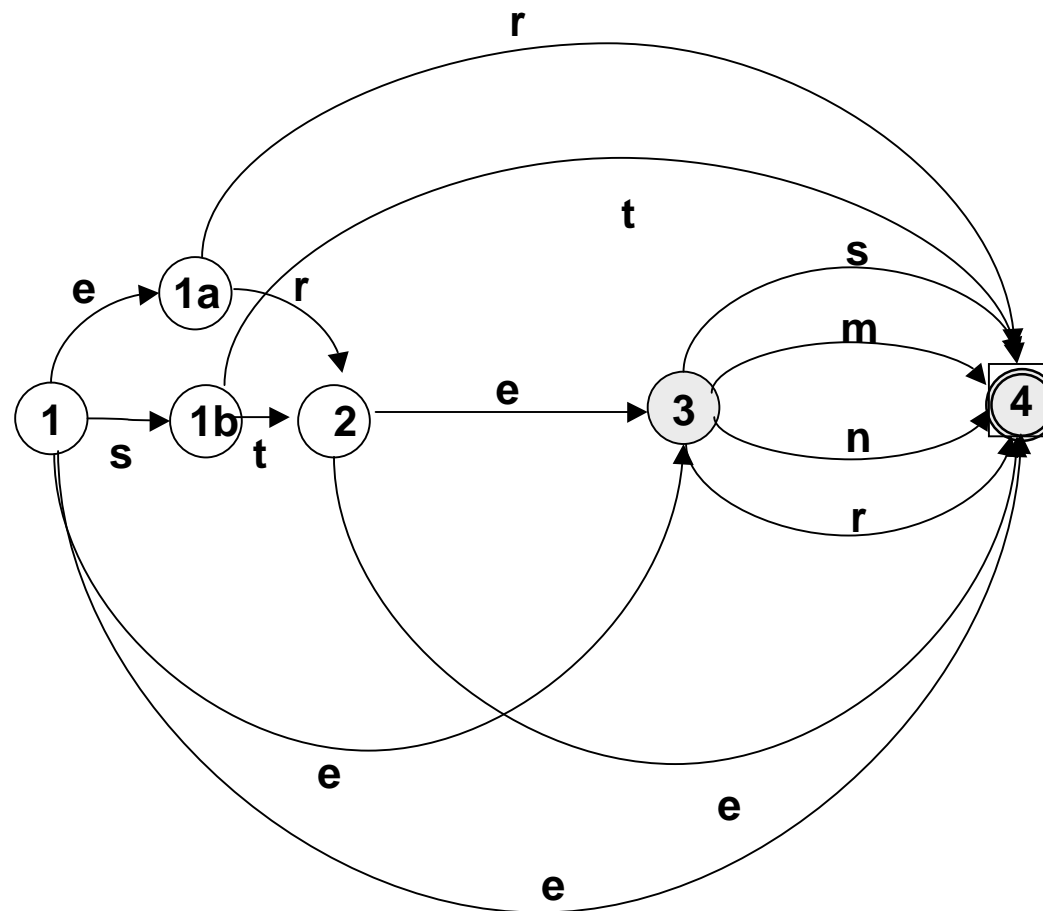
Eingabewort: klein eres

Pfadsuche als Breitensuche



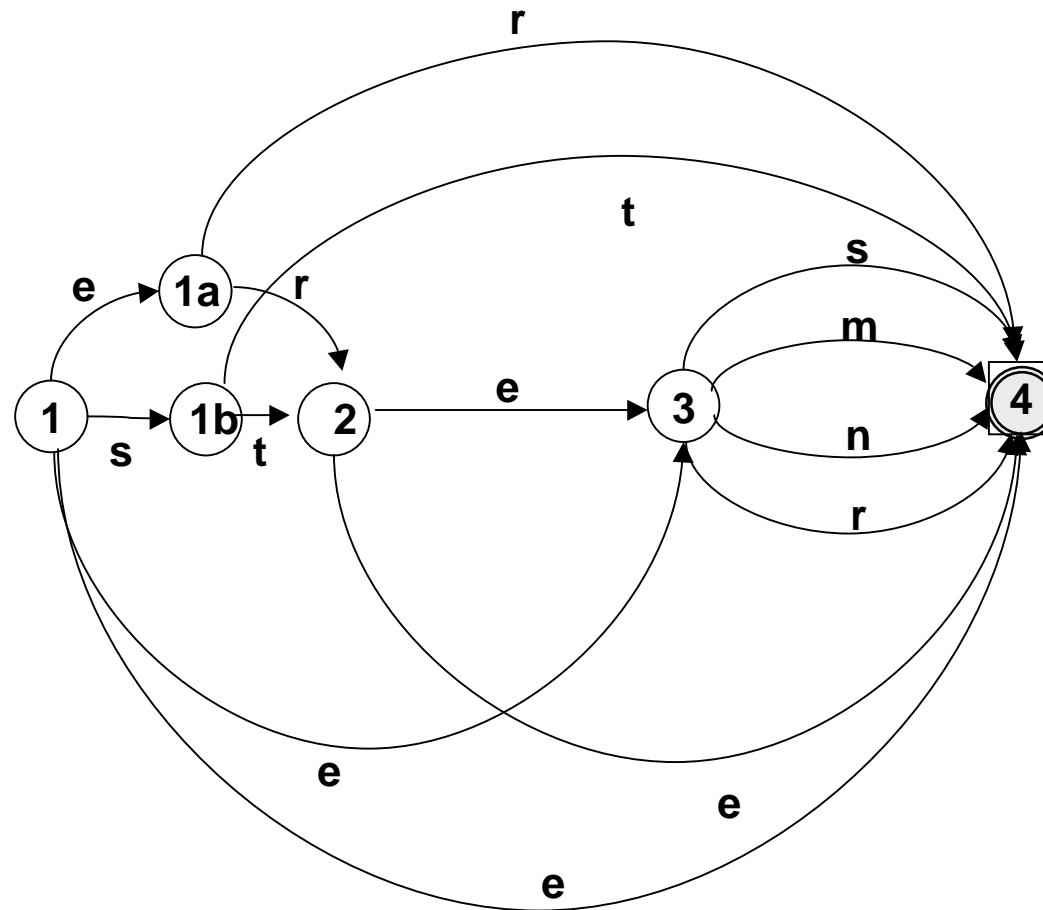
Eingabewort: klein eres

Pfadsuche als Breitensuche



Eingabewort: klein eresu

Pfadsuche als Breitensuche



Eingabewort: klein eres_

Schritt 3: Potenzautomaten-Konstruktion, Vorüberlegung [2]

- Wir können „getaktete“ Breitensuche in einem buchstabierenden NEA so beschreiben:
 - Wir ermitteln alle Zustände, die durch die Abarbeitung des ersten Eingabesymbols vom Startzustand aus erreicht werden können.
 - Wir ermitteln alle Zustände, die durch die Abarbeitung des zweiten Eingabesymbols von einem Zustand dieser Zustandsmenge erreicht werden können, usf.
 - Wenn die Zustandsmenge, die wir auf diese Weise nach Abarbeiten des kompletten Wortes w enthalten, einen Endzustand des NEA enthält, wird w akzeptiert.

Schritt 3: Potenzautomaten-Konstruktion, Vorüberlegung [3]

- Wir können das Suchverfahren selbst mit einem endlichen Automaten beschreiben:
 - Zustände des neuen Automaten lassen sich als Mengen von Zuständen des NEA beschreiben. Am Beispiel: Nach Abarbeiten des ersten Symbols „e“ befindet er sich in dem Zustand, dass es die Zustandsmenge des NEA $\{1a, 2, 4\}$ als mögliche aktuelle Zustände erkannt hat.
 - Wenn die Eingabekette abgearbeitet ist, und der Automat sich in einem Zustand befindet, der einen Endzustand des NEA enthält, ist die Eingabe akzeptiert.
 - Die „möglichen Zustände“ des NEA, die sich durch ein bestimmtes Eingabe-Symbol erreichen lassen, sind eindeutig definiert. Der neue Automat ist also ein DEA.

Schritt 3: Potenzautomaten-Konstruktion: Die Definition

Der Potenzautomat zum buchstabierenden NEA

$A = \langle K, \Sigma, \Delta, s, F \rangle$ ist der DEA $A' = \langle K', \Sigma, \delta, s', F' \rangle$
mit:

- $K' = \wp(K)$ (die Potenzmenge der Zustandsmenge des NEA)
- $s' = \{s\}$
- $\delta(p', a) = \{q \mid \text{es gibt } p \in p' \text{ und } \langle p, a, q \rangle \in D\}$ für
jedes $p' \subseteq K, a \in \Sigma$
- $q' \in F'$ gdw. $q' \cap F \neq \emptyset$

Praktisches Vorgehen

Der Potenzautomat A' zu $A = \langle K, \Sigma, \Delta, s, F \rangle$ hat $2^{|\Delta|}$ Zustände. In der Regel sind viele dieser Zustände unerreichbar (vom Startzustand $\{s\}$ aus) und deshalb funktionslos.

Praktisches Konstruktionsverfahren:

Beginne mit $\{s\}$, berechne die Übergangsfunktion für $\{s\}$, für alle direkt von s erreichbaren Zustände usw., bis keine neuen erreichbaren Zustände hinzukommen.

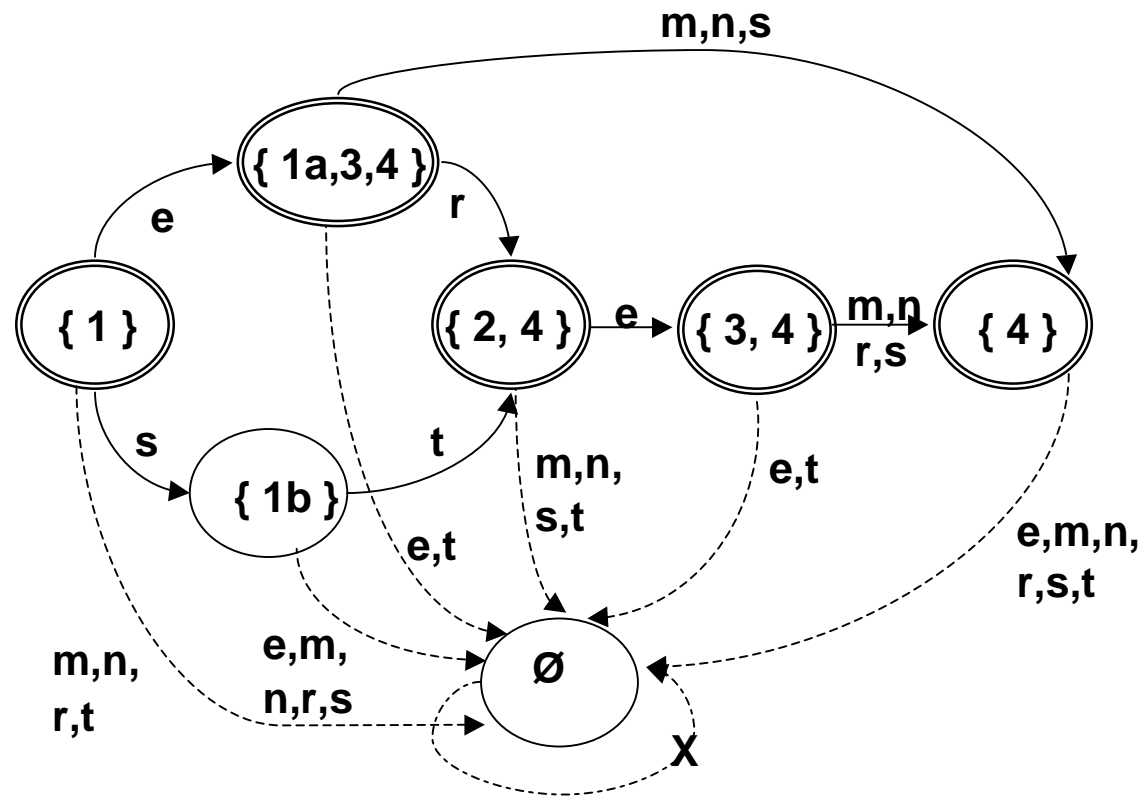
Beispiel: DEA für Adjektiv-Endungen

- Grundlage: der buchstabierende Automat
 $A = \langle \{1, 1a, 1b, 2, 3, 4\}, \{e, m, n, r, s, t\}, \Delta, 1, \{1, 4\} \rangle$,
 Δ wie im Diagramm Folie 42
- Potenzautomat ist $A' = \langle K', \Sigma, \delta, s', F' \rangle$
mit
 $K' = \wp(K)$
 $s' = \{s\}$
 $F' = \{q' \in K' \mid 1 \in q' \text{ oder } 4 \in q'\}$
 δ s. Übergangstabelle nächste Folie

DEA für Adjektiv-Endungen, Übergangstabelle

$\delta:$	e	m	n	r	s	t
$\{1\}$	$\{1a,3,4\}$	\emptyset	\emptyset	\emptyset	$\{1b\}$	\emptyset
$\{1a,3,4\}$	\emptyset	$\{4\}$	$\{4\}$	$\{2,4\}$	$\{4\}$	\emptyset
$\{1b\}$	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset	$\{2,4\}$
$\{2,4\}$	$\{3,4\}$	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset
$\{3,4\}$	\emptyset	$\{4\}$	$\{4\}$	$\{4\}$	$\{4\}$	\emptyset
$\{4\}$	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset
\emptyset	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset

Das Diagramm



Noch einmal: Der DEA für Adjektiv-Endungen

- Die Zustandsmenge des Potenzautomaten A' ist eigentlich $K' = \wp(K)$, er hat in unserem Beispiel also $2^6 = 64$ Zustände. Wie die Übergangstabelle zeigt, sind vom Startzustand $\{1\}$ aus aber nur 7, ohne den „trap state“ \emptyset 6 echte Zustände erreichbar. Die übrigen Zustände können ignoriert werden.

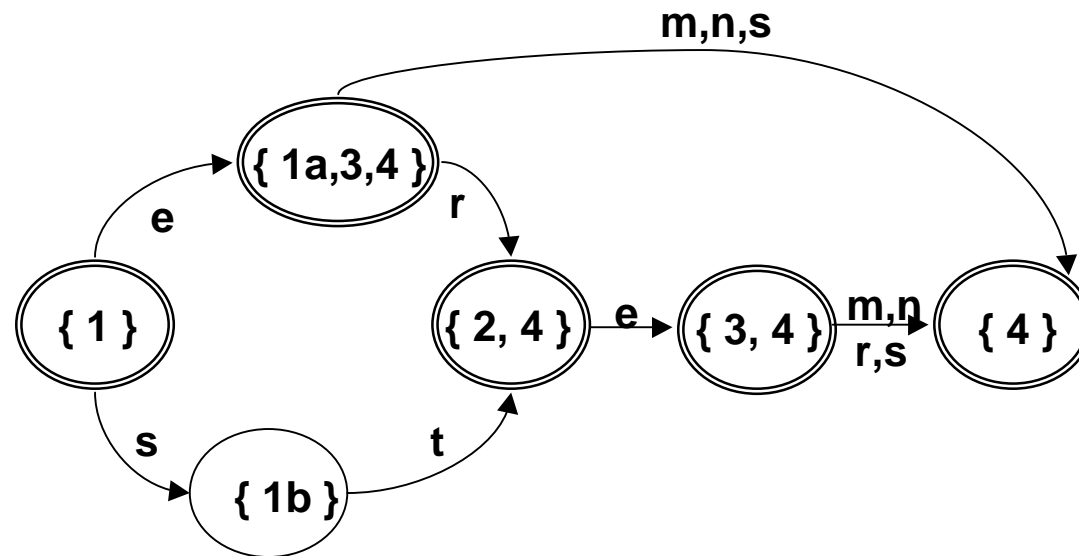
Wir können also, wie im Diagramm, ausgehen von:

$$K' = \{\{1\}, \{1a,3,4\}, \{1b\}, \{2,4\}, \{3,4\}, \{4\}, \emptyset\} \quad \text{und}$$

$$F' = \{\{1\}, \{1a,3,4\}, \{2,4\}, \{3,4\}, \{4\}\}$$

- Im Diagramm können wir außerdem noch, per Konvention, den Zustand \emptyset und alle hinführenden Kanten unterschlagen, und erhalten dann das vereinfachte Diagramm auf der folgenden Folie.

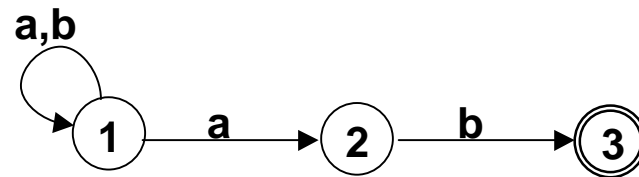
Das Diagramm, vereinfacht



Potenzautomatenkonstruktion, Beispiel 2

NEA $A = \langle \{1,2,3\}, \{a,b\}, \Delta, 1, \{3\} \rangle$

Δ gegeben durch:



DEA

$$A' = \langle \wp(\{1,2,3\}), \{a,b\}, \delta, \{1\}, F' \rangle$$

$$F' = \{\{3\}, \{1,3\}, \{2,3\}, \{1,2,3\}\}$$

Potenzautomatenkonstruktion, Beispiel 2: Die Übergangstabelle

q	$\delta(q, a)$	$\delta(q, b)$
{1}	{1,2}	{1}
{2}	\emptyset	{3}
{3}	\emptyset	\emptyset
{1,2}	{1,2}	{1,3}
{1,3}	{1,2}	{1}
{2,3}	\emptyset	{3}
{1,2,3}	{1,2}	{1,3}
\emptyset	\emptyset	\emptyset

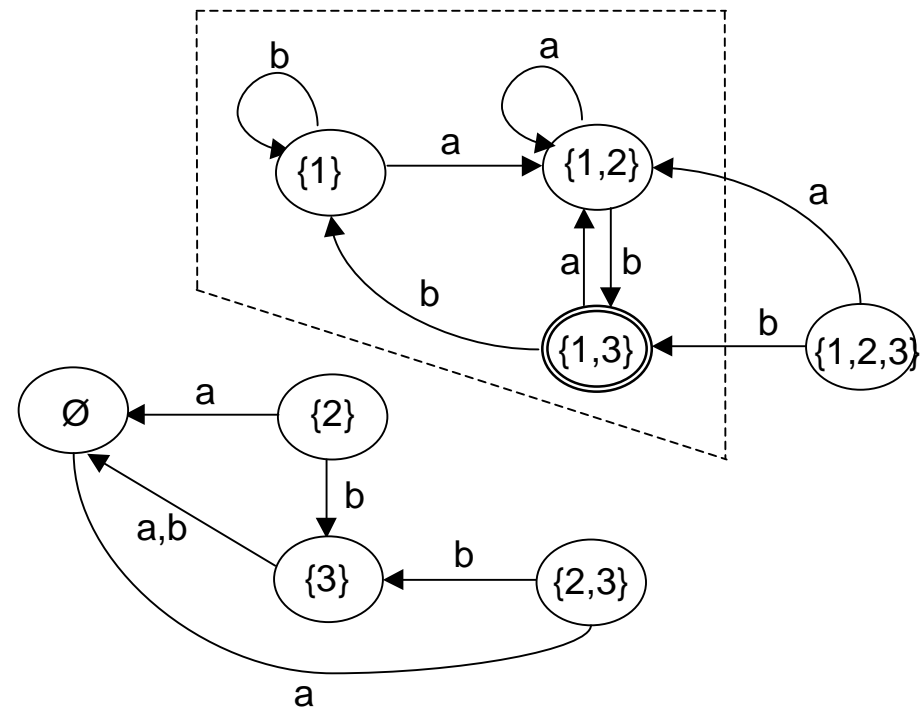
Potenzautomatenkonstruktion, Beispiel 2: Die Übergangstabelle

q	$\delta(q, a)$	$\delta(q, b)$
{1}	{1,2}	{1}
{2}	\emptyset	{3}
{3}	\emptyset	\emptyset
{1,2}	{1,2}	{1,3}
{1,3}	{1,2}	{1}
{2,3}	\emptyset	{3}
{1,2,3}	{1,2}	{1,3}
\emptyset	\emptyset	\emptyset

Potenzautomatenkonstruktion, Beispiel 2: Das Zustandsdiagramm

Nur ein Teil der
Zustände ist vom
Startzustand aus
erreichbar.

Die übrigen
Zustände sind
funktionslos.



Ein dritter Formalismus: Reguläre Ausdrücke

- Reguläre Ausdrücke sind, neben NEA und DEA, ein dritter Formalismus, um Eigenschaften von Zeichenketten bzw. Sprachen zu beschreiben.
- Definition: Die Menge der regulären Ausdrücke zu einem Alphabet Σ ist die kleinste Menge, für die gilt:
 - \emptyset ist regulärer Ausdruck
 - Jedes $a \in \Sigma$ ist regulärer Ausdruck
 - Wenn α, β reguläre Ausdrücke sind, so auch
 - $\alpha + \beta$ (Alternative)
 - $\alpha \circ \beta$ (Konkatenation)
 - α^* (Iteration)

Reguläre Ausdrücke: Die Interpretation

- Reguläre Ausdrücke beschreiben Sprachen/ Mengen von Zeichenketten über einem gegebenen Alphabet.
- Die vom regulären Ausdruck Φ über Σ beschriebene Sprache nennen wir $L(\Phi)$. $L(\Phi)$ wird für beliebige reguläre Ausdrücke in der folgenden Weise rekursiv definiert:
 - $L(\emptyset) = \emptyset$
 - $L(a) = \{a\}$ für $a \in \Sigma$
 - $L(\alpha + \beta) = L(\alpha) \cup L(\beta)$
 - $L(\alpha \circ \beta) = \{ww' \mid w \in L(\alpha) \text{ und } w' \in L(\beta)\}$
 - $L(\alpha^*) = \{w_1 \dots w_n \mid w_1, \dots, w_n \in L(\alpha), n \geq 0\}$
- Anmerkung: Es gilt $L(\emptyset^*) = \{\epsilon\}$

Reguläre Ausdrücke und endliche Automaten

- Wird eine Sprache durch einen regulären Ausdruck beschrieben, kann sie auch durch einen nicht-deterministischen endlichen Automaten dargestellt werden: Zu jedem regulären Ausdruck Φ gibt es einen NEA A mit $L(A) = L(\Phi)$.
 - Beweis: Konstruktiver Beweis durch Induktion über den Aufbau der regulären Ausdrücke.
- Da jeder NEA in einen äquivalenten DEA überführt werden kann, gilt auch: Jede reguläre Sprache wird von einem DEA akzeptiert. Die Zugehörigkeit von Worten kann also in linearer Zeit getestet werden.
- Zu jedem DEA A gibt es einen regulären Ausdruck Φ mit $L(\Phi) = L(A)$.
- NEA, DEA und reguläre Ausdrücke sind Formalismen mit äquivalenter Beschreibungsstärke. Die Sprachen, die sich durch sie beschreiben lassen, heißen reguläre Sprachen.

Einige Anwendungen von endlichen Automaten

- Morphologie:
 - Unter anderem: Flexionsmorphologie, Lemmatisierung/Stemming
- Suche in Textdokumenten
 - Z.B. Korpusuche in der Lexikografie mit regulären Ausdrücken (PERL: Sprache zur Stringsuche)
 - Suche und Informationszugriff in Archiven und Internet
- Syntax:
 - z.B. Erkennung von Wortartmustern in Phrasen/Satzteilen
 - Identifikation von Schlüsselwörtern /-phrasen in Dialogsystemen
- Dialogstruktur
 - Beschreibung von Dialogmustern mit Automaten

Syntax

- Gegenstand der Morphologie ist die Struktur des Wortes: der Aufbau von Wörtern aus Morphemen, den kleinsten funktionalen oder bedeutungstragenden Einheiten der Sprache.
- Gegenstand der Syntax ist die Struktur des Satzes: der Aufbau von Sätzen aus Wörtern.
- Morphologie beschreibt die grammatischen Eigenschaften von Wörtern, die durch Wortform oder Flexionsmorpheme kodiert werden.
- Syntax beschreibt die Interaktion der grammatischen Eigenschaften unterschiedlicher Wörter im Satz.

Eigenschaften der syntaktischen Struktur [1]

- Sätze setzen sich aus Satzteilen (Konstituenten) zusammen, die wiederum aus einfachen oder ihrerseits komplexen Satzteilen bestehen können. Sätze können deshalb beliebig lang und beliebig tief geschachtelt sein.

Konstituenten

- Er hat die Übungen gemacht.
- Der Student hat die Übungen gemacht.
- Der interessierte Student hat die Übungen gemacht.
- Der an computerlinguistischen Fragestellungen interessierte Student hat die Übungen gemacht.
- Der an computerlinguistischen Fragestellungen interessierte Student im ersten Semester hat die Übungen gemacht.
- Der an computerlinguistischen Fragestellungen interessierte Student im ersten Semester, der im Hauptfach Informatik studiert, hat die Übungen gemacht.
- Der an computerlinguistischen Fragestellungen interessierte Student im ersten Semester, der im Hauptfach, für das er sich nach langer Überlegung entschieden hat, Informatik studiert, hat die Übungen gemacht.

Syntaktische Kategorien [1]

Konstituenten-Typen werden „syntaktische Kategorien“ genannt;

Beispiele:

- Nominalphrasen (Nominalausdrücke): *er – der Student – der interessierte Student – die Übungen – computerlinguistischen Fragestellungen*
- Präpositionalphrasen (Präpositionalausdrücke): *an computerlinguistischen Fragestellungen – im ersten Semester, – nach langer Überlegung*
- Adjektivphrasen: *interessierte – an computerlinguistischen Fragestellungen interessierte*
- Satz: Haupt- und Nebensätze unterschiedlicher Art

Syntaktische Kategorien [2]

Konstituenten /syntaktische Kategorien können beliebig ineinander verschachtelt sein:

- Der Nominalausdruck „*der an computerlinguistischen Fragestellungen interessierte Student im ersten Semester, der im Hauptfach, für das er sich nach langer Überlegung entschieden hat, Informatik studiert*“ enthält
- den (Relativ-)Satz „*der im Hauptfach, für das er sich nach langer Überlegung entschieden hat, Informatik studiert*“; der enthält
- den Nominalausdruck „*Hauptfach, für das er sich nach langer Überlegung entschieden hat*“; der enthält
- den (Relativ-)Satz „*für das er sich nach langer Überlegung entschieden hat*“; der enthält
- unter anderem den Nominalausdruck „*er*“.

Kategorie und Funktion

- Die syntaktische Kategorie ergibt sich aus dem internen Aufbau einer Konstituente, insbesondere aus der Wortart ihres „lexikalischen Kopfes“: Die Konstituenten
 - *der an computerlinguistischen Fragestellungen interessierte Student*
 - *an computerlinguistischen Fragestellungen interessiert*
 - *an computerlinguistischen Fragestellungen*sind Nominal-, Adjektiv- und Präpositionalphrase, weil der jeweilige Kopf Substantiv („Nomen“), Adjektiv, bzw. Präposition ist.
- Die grammatische Funktion dagegen bezeichnet die Rolle, die eine Konstituente im ganzen Satz spielt. Ein Nominalausdruck z.B. kann, je nach Stellung im Satz unter anderem die Funktion von Subjekt, (direktem oder indirektem) Objekt, (Genitiv-) Attribut oder Prädikatsnomen besitzen.

Eigenschaften der syntaktischen Struktur

- Sätze setzen sich aus Satzteilen (Konstituenten) zusammen, die wiederum aus einfachen oder ihrerseits komplexen Satzteilen bestehen können. Sätze können deshalb beliebig lang und beliebig tief geschachtelt sein.
- Die Syntax natürlicher Sprachen erlaubt variable Wortstellung: Wörter und Konstituenten mit der gleichen Funktion können an unterschiedlichen Stellen eines Satzes stehen. Unterschiedliche Sprachen erlauben sehr unterschiedliche Freiheitsgrade.

Variable Wortstellung

Peter hat der Dozentin das Übungsblatt heute ins Büro gebracht.

Das Übungsblatt hat Peter der Dozentin heute ins Büro gebracht.

Der Dozentin hat Peter heute das Übungsblatt ins Büro gebracht.

Ins Büro hat heute Peter der Dozentin das Übungsblatt gebracht.

Heute hat Peter das Übungsblatt der Dozentin ins Büro gebracht.

Ins Büro hat das Übungsblatt der Dozentin Peter heute gebracht.

Variable Wortstellung

Peter hat der Dozentin das Übungsblatt heute ins Büro gebracht.

Das Übungsblatt hat Peter der Dozentin heute ins Büro gebracht.

Der Dozentin hat Peter heute das Übungsblatt ins Büro gebracht.

Ins Büro hat heute Peter der Dozentin das Übungsblatt gebracht.

Heute hat Peter das Übungsblatt der Dozentin ins Büro gebracht.

Ins Büro hat das Übungsblatt der Dozentin Peter heute gebracht.

** Ins Büro heute Peter das Übungsblatt hat gebracht der Dozentin.*

** Ins heute Büro der Peter Dozentin das hat Übungsblatt gebracht.*

Eigenschaften der syntaktischen Struktur [3]

- *Der [m,sg, nom]an computerlinguistischen Fragestellungen interessierte [m,sg, nom] Student [m,sg, nom] im ersten Semester, der [m,sg, nom] im Hauptfach, für das er [m,sg, nom] sich nach langer Überlegung entschieden hat [sg], Informatik studiert [sg], hat die Übungen gemacht.*

Eigenschaften der syntaktischen Struktur [3]

- Sätze setzen sich aus Satzteilen (Konstituenten) zusammen, die wiederum aus einfachen oder ihrerseits komplexen Satzteilen bestehen können. Sätze können deshalb beliebig lang und beliebig tief geschachtelt sein.
- Die Syntax natürlicher Sprachen erlaubt variable Wortstellung: Wörter und Konstituenten mit der gleichen Funktion können an unterschiedlichen Stellen eines Satzes stehen. Unterschiedliche Sprachen erlauben sehr unterschiedliche Freiheitsgrade.
- Die grammatischen Eigenschaften unterschiedlicher Wörter und Konstituenten im Satz hängen voneinander ab – zum Teil auch in Fällen, in denen die Wörter und Konstituenten im Satz weit auseinander liegen.

Kontextfreie Grammatiken: Ein Beispiel

$S \rightarrow NP VI$

$S \rightarrow NP VT NP$

$NP \rightarrow ART NN$

$VI \rightarrow \textit{schläft}$

$VI \rightarrow \textit{arbeitet}$

$VT \rightarrow \textit{studiert}$

$VT \rightarrow \textit{wählt}$

$ART \rightarrow \textit{der}$

$NP \rightarrow ART NN SREL$

$SREL \rightarrow RPRO VI$

$SREL \rightarrow RPRO NP VT$

$NN \rightarrow \textit{Student}$

$NN \rightarrow \textit{Fach}$

$RPro \rightarrow \textit{der}$

$RPro \rightarrow \textit{das}$

$ART \rightarrow \textit{das}$

- Kontextfreie Grammatiken heißen in der Linguistik auch Konstituentenstruktur-Grammatiken oder Phrasenstruktur-Grammatiken

Eine kontextfreie Grammatik für deutsche Sätze

Kompaktere Schreibweise:

- Optionale Konstituenten werden in Klammern geschrieben.
- Lexikalische Symbole werden mit Komma aneinandergehängt.

$S \rightarrow NP VI$ $NP \rightarrow ART NN (SREL)$

$SREL \rightarrow RPRO VI$ $S \rightarrow NP VT NP$

$SREL \rightarrow RPRO NP VT$

$VI \rightarrow$ *schläft, arbeitet, wartet, ...*

$VT \rightarrow$ *wählt, studiert, liest, kennt, ...*

$NN \rightarrow$ *Student, Fach, Dozentin, Professor, Buch, ...*

$RPro \rightarrow$ *der, die, das*

$ART \rightarrow$ *der, die, das*