Marco Kunze (makunze@cs.tu-berlin.de) **WS 2001/2002**
Sebastian Nowozin (nowozin@cs.tu-berlin.de) **21. 1. 2002**

Lösungen zur 3. Projektaufgabe TheGI1

Definition: Turing-Aufzähler

Ein "Turing-Aufzähler" einer Sprache L ist eine deterministische Turingmaschine $M = (Q, \Sigma, \Gamma, \Box, k, \Delta, q_A, E)$, die folgende Eigenschaften besitzt:

- $k \geq 2$

  Sie besitzt genau ein Ausgabeband und mindestens ein Arbeitsband. Sie benötigt kein Eingabeband.

- $\{\$\} \in \Gamma_M$, $\{\$\} \notin \Sigma$

  Ein zusätzliches, in Σ nicht enthaltendes Terminal "\$" wird auf dem Ausgabeband verwendet um Wortgrenzen zu markieren.

- $Q \times \Gamma^k \times Q \times \Gamma^k \times \{L\} \times \{L, N, R\}^{k-1} \cap \Delta = \emptyset$

  Das heißt, für das Ausgabeband gibt es nur die Kopfbewegungen Rechts und Nicht-Bewegung.

- $Q \times \Gamma^k \times Q \times (\Gamma \setminus (\Sigma \cup \{\$\})) \times \Gamma^{k-1} \times \{R\} \times \{L, N, R\}^{k-1} \cap \Delta = \emptyset$

  Das heißt, für jede Rechtsbewegung auf dem Ausgabeband muss ein Symbol aus $\Sigma \cup \{\$\}$ geschrieben werden.

- Das erste Zeichen auf dem Ausgabeband ist immer ein "\$".

Der Turing-Aufzähler berechnet auf dem Ausgabeband eine Menge von Wörtern $w_0, w_1, \ldots$, die durch das Markierungszeichen "\$" getrennt werden. Sei die so berechnete Menge der Wörter die Menge U. Für U gilt: $U \subseteq \Sigma_L^\star$. Ein Aufzähler M für eine Sprache L generiert $L(M_{auf}) = U$.
Der Aufzähler kann terminieren, wenn L endlich ist. Wenn L unendlich ist, so terminiert er nie. Wir definieren, dass wenn der Aufzähler terminieren sollte, er dies nicht akzeptierend macht.

1. L Turing-aufzählbar $\Rightarrow$ L akzeptierbar

Wenn L Turing aufzählbar ist, so existiert eine DTM M_{auf}, die L aufzählt. Wir konstruieren eine Maschine M, die L akzeptiert.
Die gegebene Maschine M_{auf} schreibt akzeptierte Wörter der Sprache auf ihr Ausgabeband. Diese sind durch Trennzeichen abgegrenzt. Das Trennzeichen bezeichnen wir mit "\$". Da nach Definition des Turing-Aufzählers dieser auf dem Ausgabeband nur Nicht- und Rechtsbewegungen vollführen kann, muss die Maschine nach jedem Wort dass es auf das Ausgabeband schreibt ein Trennzeichen "\$" schreiben und eine Rechtbewegung vollführen. Es gibt also in M_{auf} Übergänge der in Abbildung 1 gezeigten Form.

$$q_A \xrightarrow{\Box, \$, R} q_B$$

Abbildung 1: Form der "\$" schreibenden Übergänge in M_{auf}

Alle in M_{auf} enthaltenden Übergänge dieser Form ersetzen wir, indem wir den Teil zwischen den entsprechenden q_A und q_B Zuständen mit der in Abbildung 2 beschriebenen Maschine M_A ersetzen.
Bei den neuen Übergängen beziehen sich die oberen Ersetzungsregeln auf das Ausgabeband von M_{auf}, die unteren auf das zu akzeptierende Wort. M ist die so erzeugte Maschine.

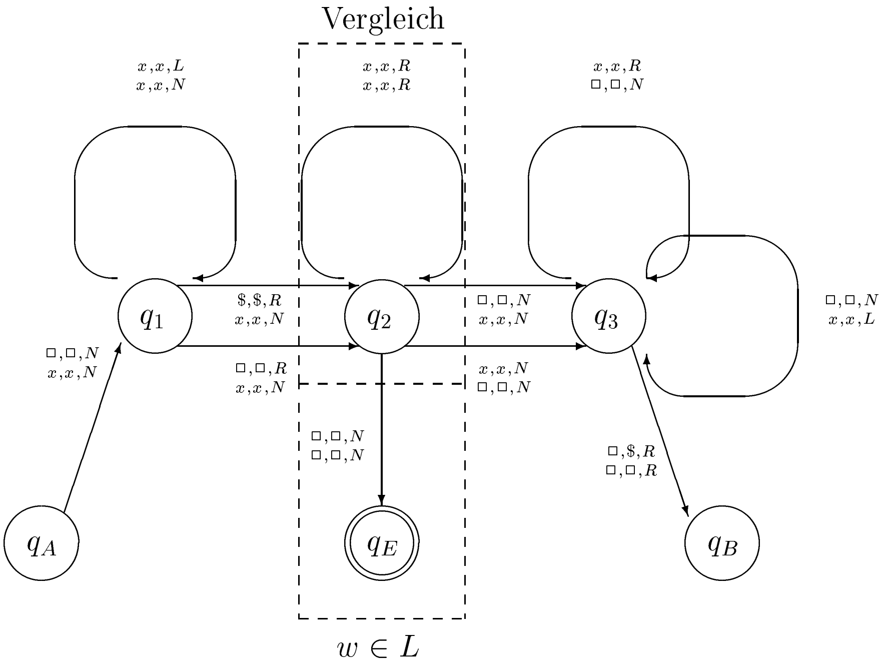


Abbildung 2: Teilmaschine M_A: Ersatzübergänge zwischen q_A und q_B

Dabei wird das auf Akzeptanz zu überprüfende Eingabewort w auf ein neues Eingabeband geschrieben, das ausschliesslich von M_A benutzt wird.

<u>Beweis</u>: Aufgrund der Konstruktion von M ist klar, dass M_A immer dann durchlaufen wird, wenn ein neues Wort auf dem Ausgabeband von M_{auf} berechnet wurde. M_A vergleicht dann jedes neue Wort mit dem Eingabewort. Es gibt zwei mögliche Fälle:

(a) $w \in L$

Alle Zeichen stimmen beim Vergleich in q_2 überein und M_A terminiert akzeptierend, da nach q_E überführt wird.

(b) $w \notin L$

Der Vergleich in q_2 schlägt fehl, die Zeichen des Eingabewortes stimmen nicht mit denen des erzeugten Wortes überein. M_A wird über q_3 und q_B verlassen, dann wird M_{auf} erneut durchlaufen.

Also terminiert die gesamte Maschine M akzeptierend genau dann, wenn ein erzeugtes Wort mit dem Eingabewort übereinstimmt, also in der Sprache enthalten ist. Da jedes Wort der Sprache irgendwann auf dem Band erzeugt wird, ist M damit ein Akzeptor für L, was zu beweisen war.

2. L Turing-aufzählbar $\Rightarrow \exists$ Aufzähler ohne Wiederholung

Wenn L Turing aufzählbar ist, so existiert eine DTM M_{auf}, die L aufzählt. Wir konstruieren einen Turing-Aufzähler M_{uniq}, der L ohne Wiederholungen aufzählt.

- Idee: Ähnlich wie im ersten Aufgabenteil modifizieren wir jeden Übergang der ein "\$" erzeugt so, dass wir überprüfen ob das jeweils erzeugte Wort schon einmal erzeugt wurde. Dazu benutzen wir zwei zusätzliche Bänder, von denen das eine das neue Ausgabeband wird, und das andere zur vorrübergehenden Speicherung des neuen Wortes benutzt wird. Die Teilmaschine M_A, die in den Übergang zwischen q_A und q_B eingesetzt wird ist in Abbildung 3 zu sehen. Sind zwei Bandoperationen angegeben, so ist die obere für das Ausgabeband der ursprünglichen Maschine M_{auf} und die untere für das temporäre Wortband. Sind drei Bandoperationen angegeben, so ist die zusätzliche, unterste für das neue Aufzähl-Ausgabeband der Maschine M_{uniq}.

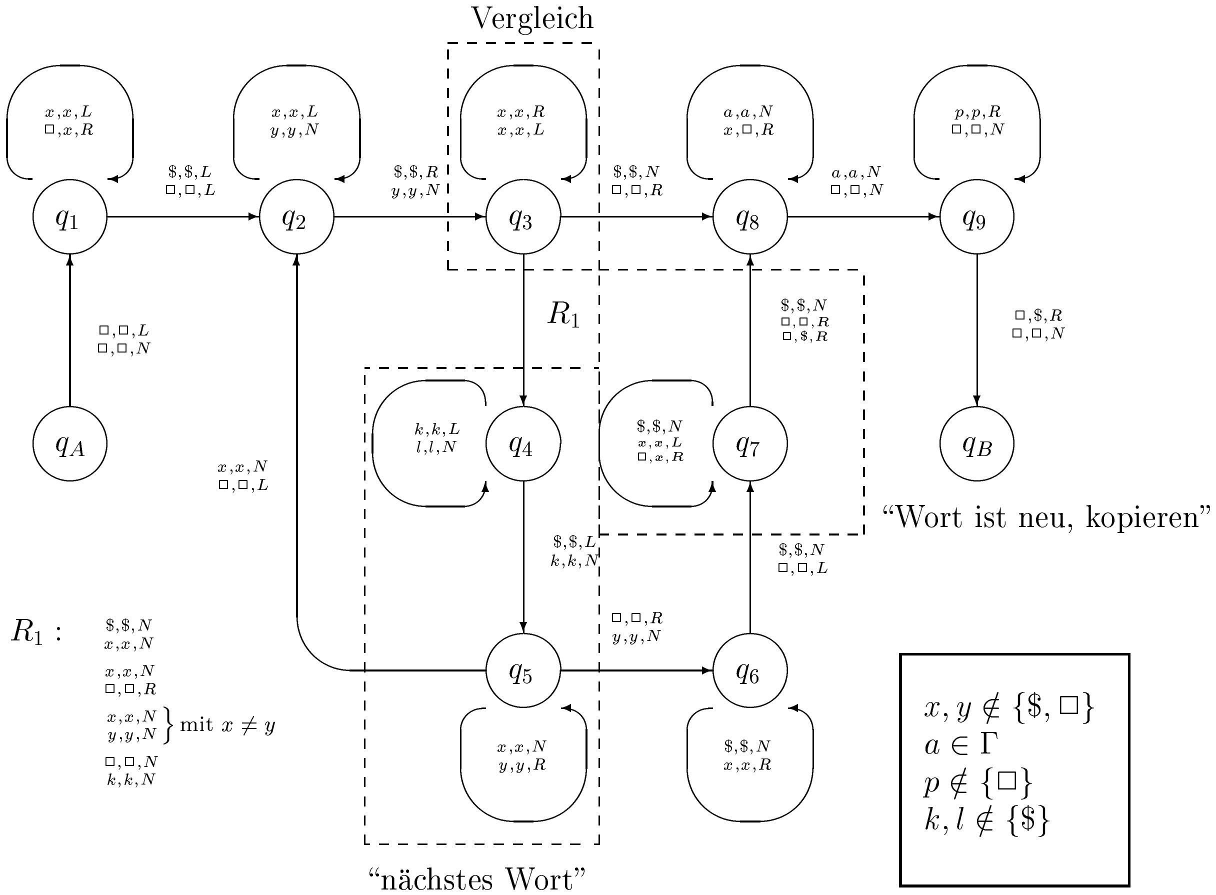


Abbildung 3: M_{uniq} Teilmaschine M_A: Ersatzübergänge zwischen q_A und q_B

- Beweis: Da wir das Ausgabeband des ursprünglichen Aufzählers M_{auf} nur lesen, aber nicht verändern, zählt M_{uniq} immer noch alle Wörter aus L auf diesem Band auf. Das heißt jedes ursprünglich erzeugte Wort wird gespeichert.

  M_A wird immer dann durchlaufen, wenn ein neues Wort auf dem Ausgabeband erzeugt wurde. Aufgrund der Konstruktion von M_A vergleicht sie das neue Wort in q_3 mit allen bereits gespeicherten Wörtern. Wir unterscheiden zwei Fälle:

  (a) w wurde bereits geschrieben
  Aufgrund der Konstruktion von M_A wird w beim Vergleich mit den bereits geschriebenen Wörtern erkannt ($q_3 \to q_8$). w wird nicht auf das neue Aufzählband kopiert.

(Anschaulich in Abbildung 3: es wird kein Zustand mit drei Bandoperationen durchlaufen)

(b) w ist ein neues Wort

w wird mit allen gespeicherten Wörtern auf dem Aufzählband verglichen. Da kein gespeichertes Wort mit w übereinstimmt wird für jedes Wort ($q_5 \rightarrow q_2$) durchlaufen. Wurden alle gespeicherten Worte durchlaufen, so wird in q_5 ein "$\square$" gelesen. Damit wird ($q_5 \rightarrow q_6$) überführt und das Wort auf das neue Ausgabeband geschrieben (q_7).

Damit wirken die mit M_A neu eingeführten Übergänge wie ein Filter, der jedes Wort höchstens einmal hindurchlässt. Da alle Wörter des ursprünglichen Aufzählers M_{auf} diesen Filter durchlaufen, jedoch jedes Wort nur einmal hindurchgelassen wird ist klar, dass M_{uniq} genau einen Aufzähler für $L(M_{auf})$ ohne Wiederholungen darstellt.

3. L geordnet Turing-aufzählbar $\Leftrightarrow$ L entscheidbar

Wir beweisen beide Implikationsrichtungen getrennt.

- Richtung: L geordnet Turing-aufzählbar $\Rightarrow$ L entscheidbar

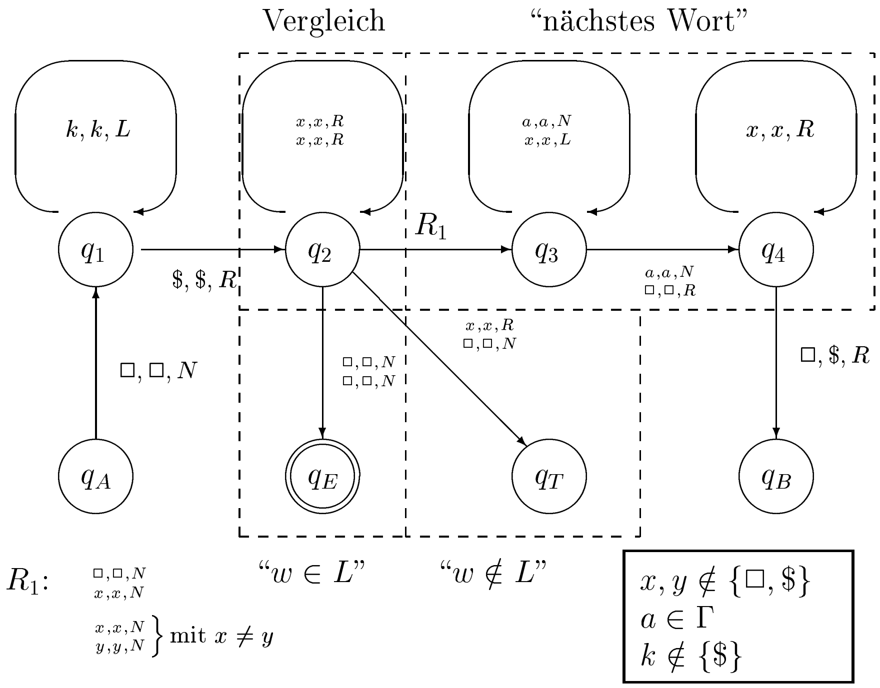


Abbildung 4: M_{ent} Teilmaschine M_A: Ersatzübergänge zwischen q_A und q_B

Die Idee ist ähnlich wie in der ersten und zweiten Aufgabe: Wir gehen von einem geordneten Turing-Aufzähler M_{auf} aus und konstruieren daraus einen Entscheider M_{ent}.

Wenn L geordnet Turing-aufzählbar ist, so existiert ein Turing-Aufzähler M_{auf}, der L geordnet aufzählt. Da sich der Aufzähler auf dem Aufzählband schreibend nur nach rechts bewegen kann existiert mindestens ein Übergang der in Abbildung 1 gezeigten Form. Diese Übergang wird immer dann durchschritten, wenn ein neues Wort auf dem Aufzählband erzeugt wurde. Wir ersetzen jeden Übergang dieser Form durch die in Abbildung 4 gezeigte Teilmaschine M_A. Weiter fügen wir ein neues Band hinzu, auf dem das Eingabewort w steht. Übergänge in M_A, die nur eine Bandoperation haben beziehen sich auf das Aufzählband, bei Übergängen mit zwei Operationen bezieht sich die zusätzliche, untere Anweisung auf das Eingabeband. Die so neu entstandene Maschine nennen wir M_{ent}.

Behauptung: M_{ent} entscheidet "$w \in L$".

Beweis: Es gibt zwei mögliche Fälle

(a) $w \in L$

Sei $w \in L$. $(q_A \to \ldots \to q_B)$ wird immer dann durchlaufen, wenn ein neues Wort auf dem Aufzählband erzeugt wurde. Im Zustand q_2 wird das neue Wort mit dem Eingabewort w verglichen. Sind sie identisch, so wird mit $(q_2 \to q_E)$ überführt und M_A terminiert akzeptierend. Da $w \in L$, zählt M_{auf} unweigerlich irgendwann w auf, und der Vergleich in q_2 ist positiv. Also terminiert M_A akzeptierend, und damit terminiert auch M_{ent} akzeptierend.

(b) $w \notin L$

Da M_{auf} geordnet ist, ist jedes Wort auf dem Aufzählband mindestens so lang wie sein Vorgänger. Das heißt, wenn die Länge des gerade erzeugten Wortes grösser $|w|$ ist, kann w auf dem Aufzählband nicht mehr erzeugt werden. Da $(q_A \to \ldots \to q_B)$ für jedes erzeugte Wort durchlaufen wird, werden alle erzeugten Worte im Zustand q_2 mit w verglichen.

Wir unterscheiden zwei Fälle, die in M_A mit dem Eingabewort w auftreten.

- w kürzer als das erzeugte Wort
  Ist $|w|$ dabei kleiner als die Länge des erzeugten Wortes wird in Zustand q_T überführt und M_A terminiert nicht akzeptierend. Damit terminiert auch M_{ent} nicht akzeptierend.
- w länger oder gleich lang wie das erzeugte Wort
  Da $w \notin L$, zählt M_{auf} nie das Wort w auf. Für den Fall das $|w|$ grösser oder gleich der Länge des erzeugten Wortes ist, schlägt damit der Vergleich in q_2 an irgendeiner Stelle fehl und es wird mit $(q_2 \to q_3)$ überführt. M_A wird über q_B verlassen.

Da die Länge der erzeugten Worte nicht kleiner werden kann und kein Wort doppelt erzeugt wird, steigt unweigerlich die Länge der erzeugten Wörter. Solange $|w| \geq |\text{neues Wort}|$ ist, schlagen alle Vergleiche fehl. Unweigerlich ist irgendwann $|w| < |\text{neues Wort}|$, dann terminiert die Maschine nicht akzeptierend.

Da genau einer der beiden Fälle unweigerlich erreicht werden muss ist M_{ent} ein Entscheider für L.

- Richtung: L entscheidbar $\Rightarrow$ L geordnet Turing-aufzählbar

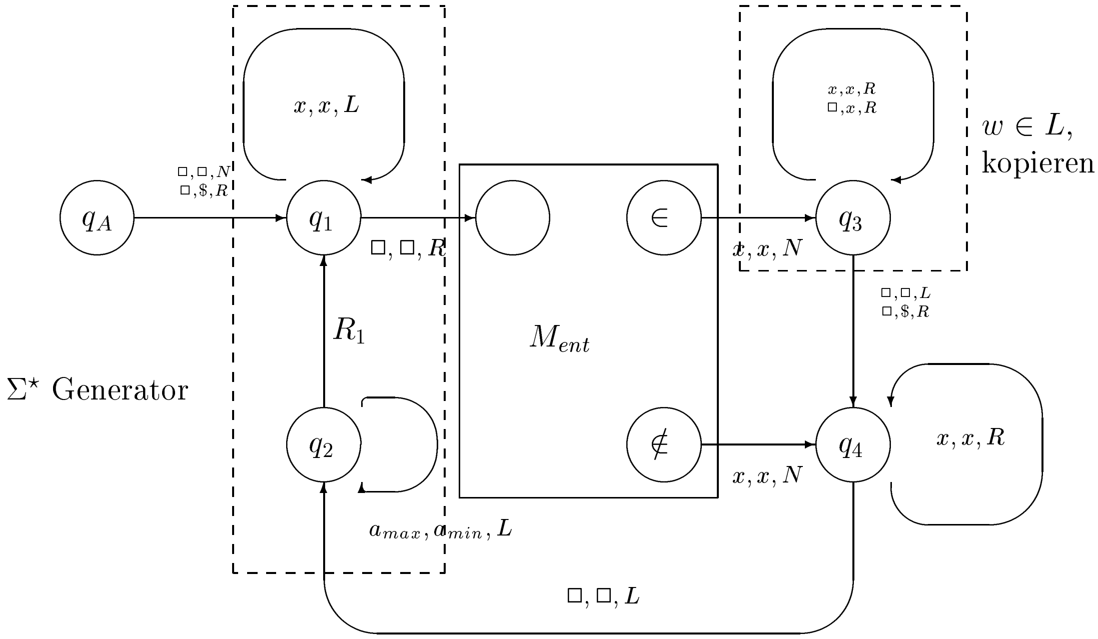
Wenn L entscheidbar ist, gibt es einen Entscheider M_{ent}, der bei gegebenem w entscheidet ob $w \in L$. Wir konstruieren mit Hilfe von M_{ent} einen geordneten Turing-Aufzähler M_{auf}, der L geordnet aufzählt.

Idee: Wir generieren geordnet alle $w \in \Sigma^\star$ und testen ob M_{ent} das Wort akzeptiert oder nicht. Wenn es akzeptiert wird, so wird es auf ein Aufzählband geschrieben, ansonsten verworfen. Danach wird das nächste mögliche Wort in $\Sigma^\star$ generiert.

Die in Abbildung 5 gezeigte Konstruktion besteht aus drei Teilen: dem Wort-generierendem Teil in q_1 und q_2, dem Entscheider M_{ent} und dem Kopierteil in q_3. q_4 und q_A sind nur Hilfszustände. Die oberen Bandanweisungen beziehen sich auf das Band, dass $\Sigma^\star$ generiert. Wenn zusätzlich eine zweite Bandanweisung gegeben ist, so bezieht sie sich auf das Aufzählband von M_{auf}.

Um geordnet aufzählen zu können, ist es nötig dem Alphabet Σ eine Reihenfolge zuzuordnen. Wir definieren diese mit $\Sigma = \{a_{min}, a_{min+1}, \ldots, a_{max-1}, a_{max}\}$.

Wir verlangen vom Entscheider M_{ent} dass er $w \in L$ entscheidet wenn er als Eingabe auf dem Eingabeband ein mit "$\square$" terminiertes Wort w gegeben kriegt, und der Lesekopf auf dem ersten Zeichen w_0 steht. Genau so hat er das Band auch zu verlassen.



R_1: $\{(\Box, a_{min}, N), (a_{min}, a_{min+1}, N), \ldots, (a_{max-1}, a_{max}, N)\}$ $\boxed{x \notin \{\Box\}}$

Abbildung 5: geordneter Turing-Aufzähler M_{auf}

<u>Behauptung</u>: M_{auf} zählt L auf.

<u>Beweis</u>: Alle Wörter aus $\Sigma^\star$ werden geordnet generiert. Wir nennen das aktuell generierte Wort w. Aufgrund der Konstruktion von M_{auf} wird w sofort Eingabe des Entscheiders. Also ist die Eingabe des Entscheiders auch geordnet. Der Entscheider liefert immer eine Antwort auf die Frage "$w \in L$?". Es gibt zwei mögliche Fälle

- $w \in L$

  Wenn $w \in L$ dann wird M_{ent} nach q_3 verlassen. In q_3 wird w auf das Aufzählband kopiert. Danach wird durch den Übergang $(q_3 \to q_4)$ ein "\$" geschrieben. Das Wort w wird also auf das Aufzählband geschrieben, wenn es in L ist. Dies gilt für jedes Wort das in L ist. Da die Wörter geordnet getestet werden, sind die Wörter auf dem Ausgabeband auch geordnet.

- $w \notin L$

  M_{ent} wird zu q_4 verlassen, wenn $w \notin L$. Damit wird w nicht auf das Aufzählband geschrieben.

Nachdem das Wort entweder auf das Aufzählband geschrieben wurde ($w \in L$) oder verworfen wurde ($w \notin L$), wird das nächste Wort aus $\Sigma^\star$ generiert und erneut getestet.

Da $L \subseteq \Sigma^\star$ werden alle Wörter getestet und M_{auf} zählt L geordnet auf. M_{auf} terminiert nie.

4. Sonderfall: L endlich, oder $L = \emptyset$

Zum besseren Verständnis sind als Beispiele die Ausgaben einiger möglicher Aufzähler für endliche Sprachen aufgeführt.

Sprache	Bandinhalt
$\emptyset$	$\ldots, \Box, \$, \Box, \ldots$
$\{\lambda\}$	$\ldots, \Box, \$, \$, \Box, \ldots$
$\{a, b, c\}$	$\ldots, \Box, \$, a, \$, a, \$, c, \$, b, \$, \Box, \ldots$

Für die einzelnen Aufgaben gelten bei endlichen Sprachen folgende Einschränkungen:

- Definition

  Wenn L endlich ist, kann M terminieren, muss dies aber nicht tun. Wenn die Maschine nicht terminiert, kann sie entweder Wörter auf dem Ausgabeband wiederholen oder in eine Schleife laufen, ohne jemals wieder Wörter zu erzeugen.

  Für $L = \emptyset$ steht nur ein "\$" auf dem Aufzählband. Die Maschine kann terminieren. Auf dem Aufzählband bewegt sie sich außer für das erste "\$" nicht, da Rechtsbewegungen nur schreibend erlaubt sind.

- 1. L aufzählbar $\Rightarrow$ L akzeptierbar

  Für endliche L bleibt die Konstruktion korrekt und M bleibt Akzeptor für L. Der Beweis ist analog. Ausnahme: Da M_{auf} terminieren kann, dies aber per Definition nicht akzeptierend tut, kann M nur innerhalb von M_A akzeptierend terminieren. Wenn aber M_{auf} terminiert, so ist M_A bereits für jedes $w \in L$ mindestens einmal durchlaufen worden. Damit bleibt auch in diesem Fall M Akzeptor für L.

- 2. L aufzählbar $\Rightarrow \exists\ M_{auf}$ ohne Wiederholungen

  Sowohl Beweis als auch Konstruktion bleiben korrekt. Für endliche L terminiert M_{uniq} genau dann, wenn M_{auf} terminiert. Für $L = \emptyset$ ist "\$" auf dem Aufzählband.

- 3a. L geordnet aufzählbar $\Rightarrow$ L entscheidbar

  Beweis und Konstruktion werden in Spezialfällen falsch. Denn für den Fall, das alle erzeugten Wörter kürzer sind als die Eingabe w muß der Aufzähler nicht terminieren, sondern kann ewig arbeiten, ohne weitere Wörter aufzuzählen. Dann ist M_{ent} kein Entscheider mehr, sondern nur noch Akzeptor für L. In allen Fällen, in denen M_{auf} terminiert, oder mindestens ein erzeugtes Wort aus $L(M_{auf})$ länger ist als die Eingabe w sind Beweis und Konstruktion jedoch korrekt.

- 3b. L entscheidbar $\Rightarrow$ L geordnet aufzählbar

  Beweis und Konstruktion sind korrekt. M_{auf} terminiert nie, was unsere Definition jedoch erlaubt.