FGI2 Uebung zum 21. Oktober

Arne Beer, MN 6489196

21. Oktober 2013

1.3 (a) Die Sprache L(A) laesst sich nicht durch einen regulaeren Ausdruck beschreiben. Der Ausdruck der der Sprache am naechsten kommen wuerde waere: $a^{x\{0,n-1\}}cb^{x\{0,n-1\}}$, wobei hier jedoch nicht sichergestellt ist, dass a und b in gleicher Anzahl wiederholt werden.

(b) $L(A_n) = \{a^x cb^x | x < n\}$

(c) Anhand der Struktur des uns vorgelegten Automaten kann man erkennen, dass der Automat $2n$ Zustaende besitzt, wobei die geraden Zustaende durch die Relation $\delta(q_a, a) = q_{a+2}$ verbunden sind und die ungeraden Zustaende durch die Relation $\delta(q_a, b) = q_{a+2}$. Zudem ist jeder gerade Zustand durch die Relation $\delta(q_a, c) = q_{a+1}$ mit einem ungeraden Zustand verbunden. Der Automat laesst sich also durch folgende Gramatik beschreiben:

$$S \longrightarrow aSb|c$$

(d) Die vorherige Grammatik $S \longrightarrow aSb|c$ laesst sich nicht als rechts oder linkslineare Grammatik beschreiben und ist dementsprechend keine regulaere Grammatik, was wiederrum eine Voraussetzung fuer eine regulaere Sprache ist. Jede Umformung der Grammatik wuerde zu einer Form der folgenden Art fuehren.

$$S \longrightarrow aX|c$$

$$X \longrightarrow Sb$$

Da diese Grammatik weder links noch rechtslinear ist, ist die Grammatik und somit die Sprache nicht regulaer.

(e)

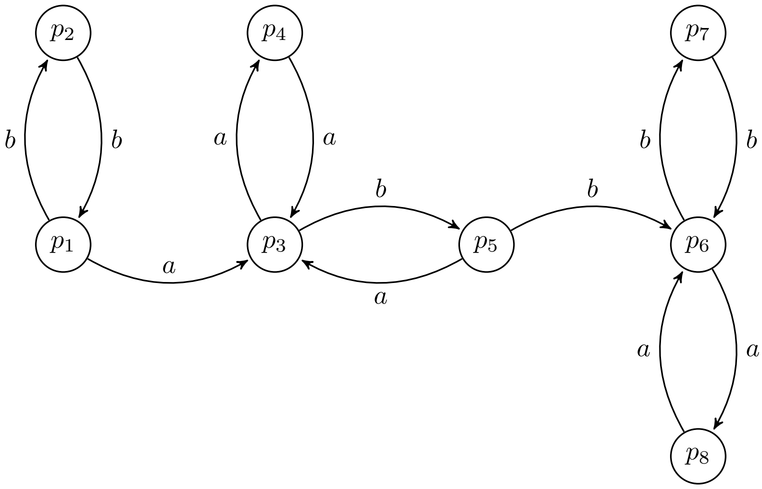
1.4 (a) Man definiert fuer jede Kante $q_1 \xrightarrow{a} q_2$ mit $a \in \Sigma$ einen Zwischenzustand $q_1 \xrightarrow{a} q_z, q_z \xrightarrow{a} q_2$. Somit ist abgesichert, dass der Automat jedes gedoppelte Wort der Sprache L(A) aktzeptiert. Der Automat ist nun zwar nicht mehr vollstaendig, aber es war lediglich vorrausgesetzt, dass der Automat deterministisch ist.

(b) Es muss abgesichert werden, von jedem Zustand mit dem Eingabewort abb ein Endzustand, in diesem Falle p_3 erreicht werden kann.
Fuer den Zustand p_0: $p_0 \xrightarrow{a} p_1$, $p_1 \xrightarrow{b} p_2$, $p_2 \xrightarrow{b} p_3$
Fuer den Zustand p_1 $p_1 \xrightarrow{a} p_1$, $p_1 \xrightarrow{b} p_2$, $p_2 \xrightarrow{b} p_3$
Fuer den Zustand p_2 $p_2 \xrightarrow{a} p_1$, $p_1 \xrightarrow{b} p_2$, $p_2 \xrightarrow{b} p_3$
Fuer den Zustand p_2 $p_3 \xrightarrow{a} p_3$, $p_3 \xrightarrow{b} p_3$, $p_3 \xrightarrow{b} p_3$

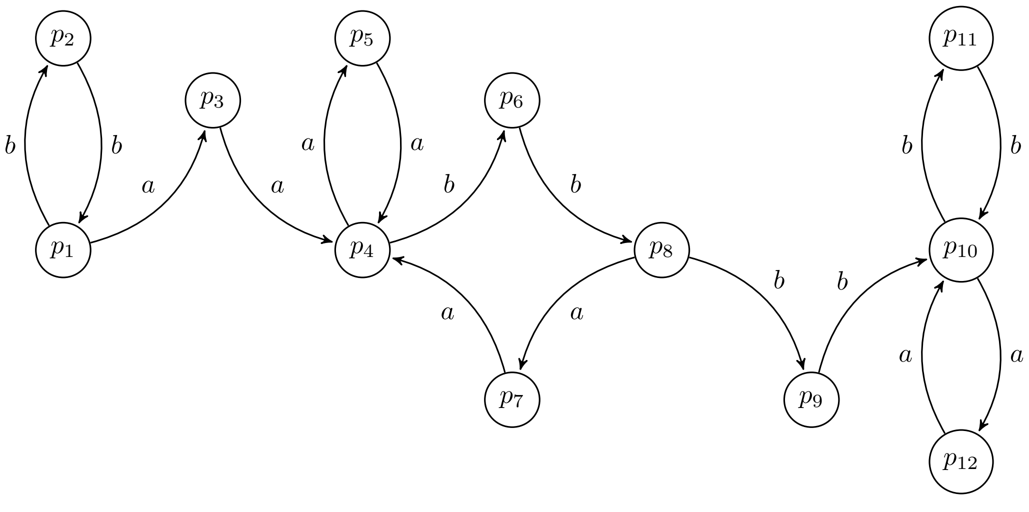
Es ist also bewiesen, dass von jedem Zustand mit dem Eingabewort abb ein Endzustand erreicht werden kann.

(c) Die vom Automaten beschrieben Sprache laesst sich folgend als regulaerer Ausdruck beschreiben. $L(A) = b^*a^+b(ab)^*ba^*b^*$

(d) Zunaechst wird jede Schleife innerhalb des Automaten, durch einen Zwischenzustand ersetzt.



Nun werden alle Uebergaenge von einem Zustand in einen anderen Zustand mit einem Zwischenzustand versehen.



Dies ist der fertig konstruierte Automat, der die gedoppelten Woerter von L(A) aktzeptiert.

(e) Die vom Automaten aktzeptierte Sprache laesst sich folgend beschreiben:

$$L(A) = (bb)^*(aa)^+bb(aabb)^*bb(aa)^*(bb)^*$$

Wie man sieht, ist der regulaere Ausdruck equivalent zu dem des urspruenglichen Automaten, bis auf das jeder Buchstabe des Alphabets verdoppelt wurde. Somit ist abgesichert, dass er exakt die gedoppelten Worte von $L(A)$ aktzeptiert