درس مبانی نظریه محاسبه

ماشینهای متناهی غیر قطعی

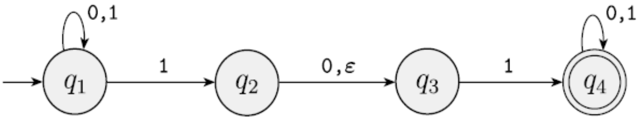
Non-deterministic Finite State Machines

مدل ماشین متناهی غیر قطعی

ماشین متناهی غیر قطعی nfa مانند ماشین قطعی dfa است با این تفاوت اساسی که تغییر وضعیت در این مدل می‌تواند بصورت غیر قطعی انجام شود.

این بدین معنی است که ماشین برای وضعیت بعدی ممکن است چند گزینه داشته باشد. در این مدل انگار ماشین از خود اختیار دارد و می‌تواند یکی از میان چند گزینه را انتخاب کند و مسیر خاصی را از میان چند مسیر ممکن طی کند. ماشین غیرقطعی حتی می‌تواند بدون مصرف رشته ورودی از یک وضعیت به وضعیت دیگر برود.

در شکل زیر دیاگرام تغییر وضعیت یک nfa نشان داده شده است.



در ماشین بالا می‌بینید که ماشین در وضعیت q_1 برای حرف 1 دو گزینه دارد: ماندن در وضعیت q_1 و یا رفتن به وضعیت q_2. همچنین ماشین در وضعیت q_2 می‌تواند بدون مصرف رشته ورودی به وضعیت q_3 برود.

تفاوتهای ماشین متناهی قطعی و غیر قطعی

- ماشین غیر قطعی در یک وضعیت با یک حرف ورودی می‌تواند چند گزینه برای وضعیت بعدی داشته باشد.

- ماشین غیرقطعی می‌تواند با مصرف ϵ (بدون مصرف رشته ورودی) از یک وضعیت به وضعیت دیگر برود بشرطی که فلش مورد نظر در دیاگرام قرارد داده شده باشد.

- ماشین غیرقطعی در یک وضعیت می‌تواند هیچ فلش خروجی نداشته باشد یا اینکه برای بعضی از حروف فلش خروجی نداشته باشد. (برای مثال در ماشین صفحه قبل، ماشین در وضعیت q_3 فلش خروجی برای 0 ندارد. این بدین معنی است که ماشین اگر وضعیت q_3 باشد و حرف ورودی 0 باشد پردازش رشته متوقف می‌شود.)

تعریف رسمی یک ماشین متناهی غیرقطعی

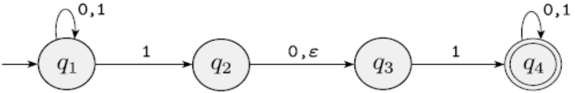
A ***nondeterministic finite automaton*** is a 5-tuple $(Q, \Sigma, \delta, q_0, F)$, where

1. Q is a finite set of states,
2. Σ is a finite alphabet,
3. $\delta \colon Q \times \Sigma_\varepsilon \longrightarrow \mathcal{P}(Q)$ is the transition function,
4. $q_0 \in Q$ is the start state, and
5. $F \subseteq Q$ is the set of accept states.

Σ_ϵ به معنی یک الفبای متناهی است که شامل رشته تهی ϵ نیز می‌باشد.

$P(Q)$ به معنی همه زیرمجموعه‌های مجموعه Q است.

یک مثال



The formal description of N_1 is $(Q, \Sigma, \delta, q_1, F)$, where

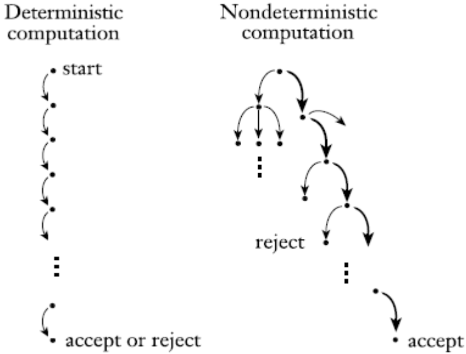
1. $Q = \{q_1, q_2, q_3, q_4\}$,
2. $\Sigma = \{0,1\}$,
3. δ is given as

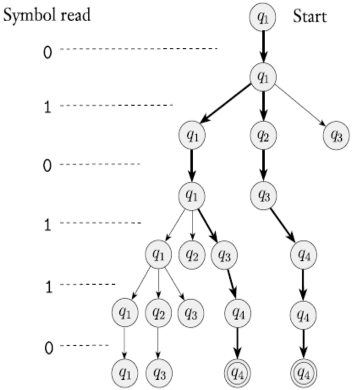
	0	1	ε
q_1	$\{q_1\}$	$\{q_1, q_2\}$	$\emptyset$
q_2	$\{q_3\}$	$\emptyset$	$\{q_3\}$
q_3	$\emptyset$	$\{q_4\}$	$\emptyset$
q_4	$\{q_4\}$	$\{q_4\}$	$\emptyset$,

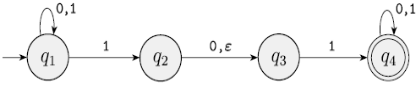
4. q_1 is the start state, and
5. $F = \{q_4\}$.

چه وقت ماشین غیر قطعی M رشته w را می‌پذیرد؟

تعریف : اگر ماشین غیرقطعی M برای رشته ورودی w از وضعیت اولیه به یک وضعیت پذیرش یک مسیر تبدیل وضعیت وجود داشته باشد، آنگاه گوییم ماشین رشته w را می‌پذیرد در غیر این صورت ماشین رشته را رد می‌کند و جزو زبانش نیست.

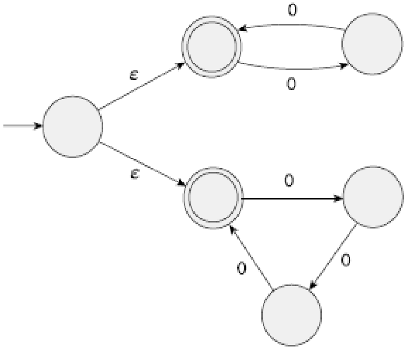



Symbol read
Start
0
1
0
1
1
0
q_1
q_2
q_3
q_4


0,1
1
0,ε
1
0,1
q_1
q_2
q_3
q_4

یک مثال

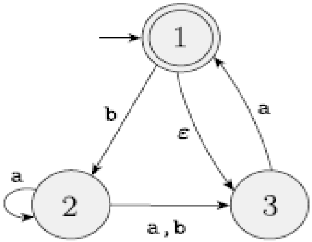
زبان ماشین غیرقطعی زیر چیست؟ حدس شما چیست؟



رشته‌هایی از الفبای $\{0\}$ که طولشان ضریبی از 2 یا 3 است.

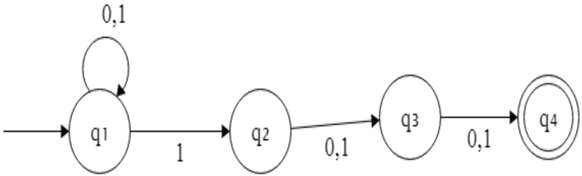
یک مثال

چند رشته مثال بزنید که ماشین زیر آنها را بپذیرد. چند رشته مثال بزنید که ماشین زیر آنها را رد کند.



یک nfa برای زبان زیر طراحی کنید.

همه رشته‌هایی از الفبای $\{0, 1\}$ که حرف سوم از آخر 1 باشد.



چرا عدم قطعیت؟

- ماشینهای متناهی غیرقطعی توانایی خوبی در مدلسازی پروسه‌هایی که در آن انتخاب وجود دارد، دارند. برای مثال در مدلسازی بازی ها و موقعیتهایی که بازیگرانی وجود دارند که از خود اختیار دارند.

- عدم قطعیت را می‌توان به منزله وجود مسیرهای مختلف و موازی به یک جواب تعبیر کرد. هر مسیر به یک وضعیت پذیرش به معنی یک شاهد و گواهی برای قبول رشته مورد نظر است (مانند اینکه برای یک ادعا چند اثبات مختلف وجود داشته باشد.)

- در بعضی جاها یک ماشین متناهی را می‌توان به نسبت بهتر و سریعتر با یک مدل غیر قطعی بیان کرد.

- عدم قطعیت در محاسبات مفهومی گسترده است و در مدلهای محاسباتی به تعابیر مختلف ظاهر می‌شود.

آیا ماشینهای متناهی غیرقطعی قوی تر هستند؟

چون ماشینهای غیرقطعی تعمیمی از ماشینهای قطعی هستند پس هرچه توسط یک dfa پذیرفته شود توسط یک nfa نیز پذیرفته می‌شود.

اما آیا زبانی وجود دارد که توسط یک nfa پذیرفته شود ولی توسط یک dfa پذیرفته نشود؟

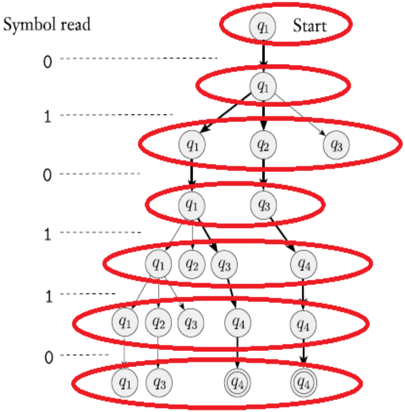
قضیه: اگر زبان L توسط یک nfa پذیرفته شود، آنگاه یک dfa وجود دارد که L را می‌پذیرد.

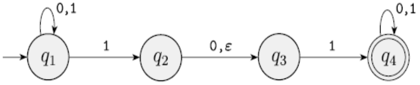
نتیجه: مدل ماشین متناهی غیرقطعی از لحاظ قدرت پذیرش از مدل ماشین متناهی قطعی قوی‌تر نیست.

نتیجه: زبان L منظم است اگر و فقط اگر توسط یک nfa (ماشین متناهی غیر قطعی) پذیرفته شود.

تبدیل nfa به یک dfa

ایده بطور خلاصه: در یک ماشین nfa ، برای هر رشته ورودی در هر زمان زیرمجموعه‌ای از وضعیتهای وجود دارد که ماشین می‌تواند در آن باشد. به مثال زیر توجه کنید. موقع پردازش رشته ورودی، انگار ماشین نامعین بین زیرمجموعه از وضعیتها حرکت می‌کند.





تبدیل nfa به یک dfa

اگر یک nfa به تعداد k وضعیت داشته باشد، پس به تعداد 2^k زیرمجموعه از وضعیتها می‌توانیم داشته باشیم. در هر زمان ماشین می‌تواند در یکی از این زیرمجموعه‌ها باشد.

فرض کنید ماشین متناهی غیرقطعی N را داشته باشیم.

$$N = (Q, \Sigma, \delta, q_0, F)$$

با داشتن N ماشین متناهی قطعی M را می‌سازیم بطوریکه

$$M = (Q', \Sigma, \delta', q'_0, F')$$

$$L(N) = L(M)$$

فرض: فعلا فرض کنید فلش ϵ نداریم. بعدا این فرض را برمی‌داریم.

- متناظر با هر زیرمجموعه از Q یک وضعیت در Q' داریم.

$$Q' = P(Q)$$

- برای $R \in Q'$ و $a \in \Sigma$ داریم:

$\delta'(R, a)$ شامل همه وضعیتهایی است که از اعضای داخل R می‌توان با حرف a به آنها رفت.

- $q_0' = \{q_0\}$

- $F' = \{R \subseteq Q' \mid F \cap R \neq \emptyset\}$

وقتی که فلش ϵ داریم. تعریف می‌کنیم برای هر $R \in Q'$ داریم:

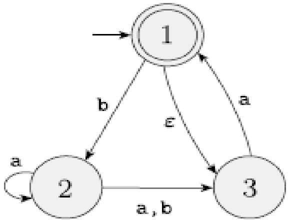
$E(R)$ شامل همه وضعیتهایی است که از اعضای داخل R می‌توان به تعداد صفر و یا بیشتر فلش ϵ به آنها رفت.

$\delta'(R, a)$ شامل همه وضعیتهایی است که از اعضای داخل $E(\delta(R, a))$ می‌توان با حرف a به آنها رفت.

این کار را باید برای وضعیت شروع هم انجام دهیم. یعنی $q'_0 = \{q_0\}$ به $E(\{q_0\})$ تغییر پیدا می‌کند.

یک مثال

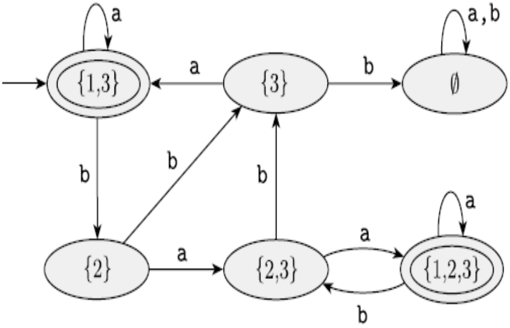
$$\{\emptyset, \{1\}, \{2\}, \{3\}, \{1,2\}, \{1,3\}, \{2,3\}, \{1,2,3\}\}.$$



دقت کنید: چون فلش ϵ داریم نخست برای هر زیرمجموعه R از وضعیتها، مجموعه $E(R)$ را محاسبه کنید. سپس طبق دستورالعمل تبدیل را انجام دهید.

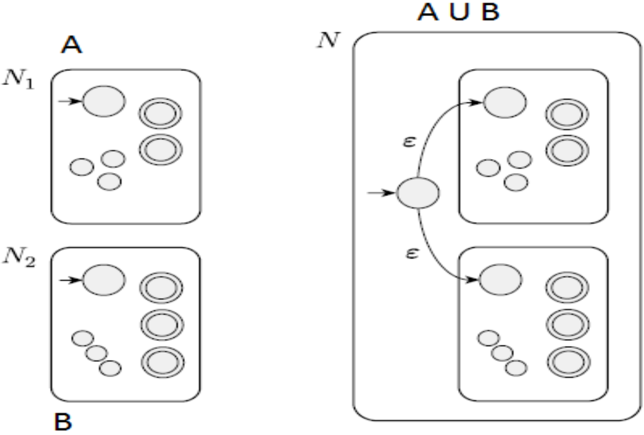
جواب نهایی بعد از حذف وضعیتهای اضافه

$$E(\{1\}) = \{1, 3\}$$

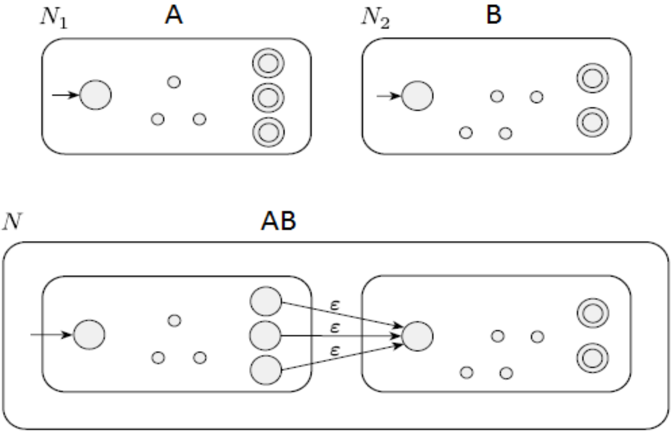


بسته بودن زبانهای منظم تحت عملگر اجتماع

یادآوری: زبان L منظم است اگر و فقط اگر توسط یک nfa (ماشین متناهی غیر قطعی) پذیرفته شود.



بسته بودن زبانهای منظم تحت عملگر اتصال



بسته بودن زبانهای منظم تحت عملگر ستاره