درس مبانی نظریه محاسبه

جلسه ششم

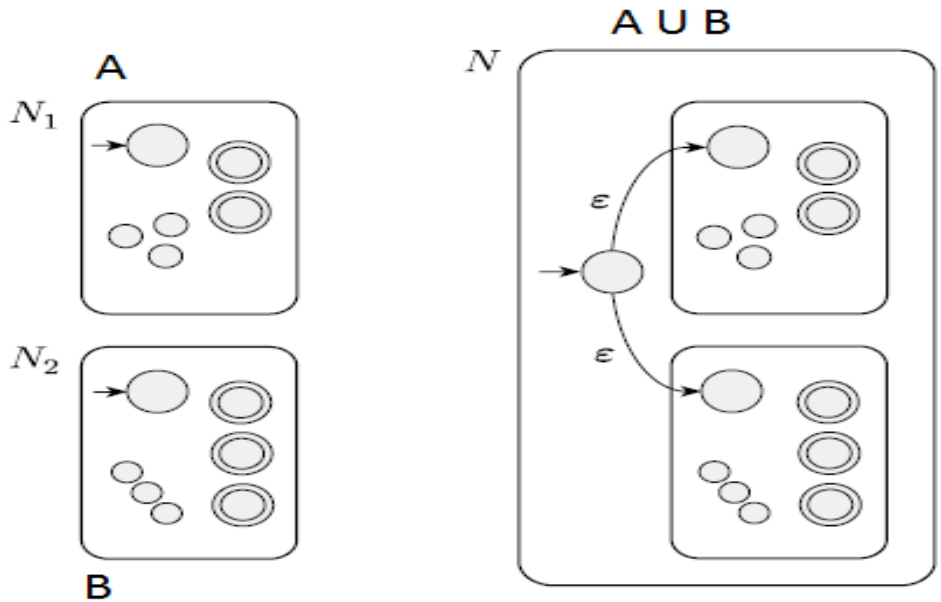
عبارات منظم و ماشین متناهی

Regular Expression and Finite State Machines

بسته بودن زبانهای منظم تحت عملگر اجتماع

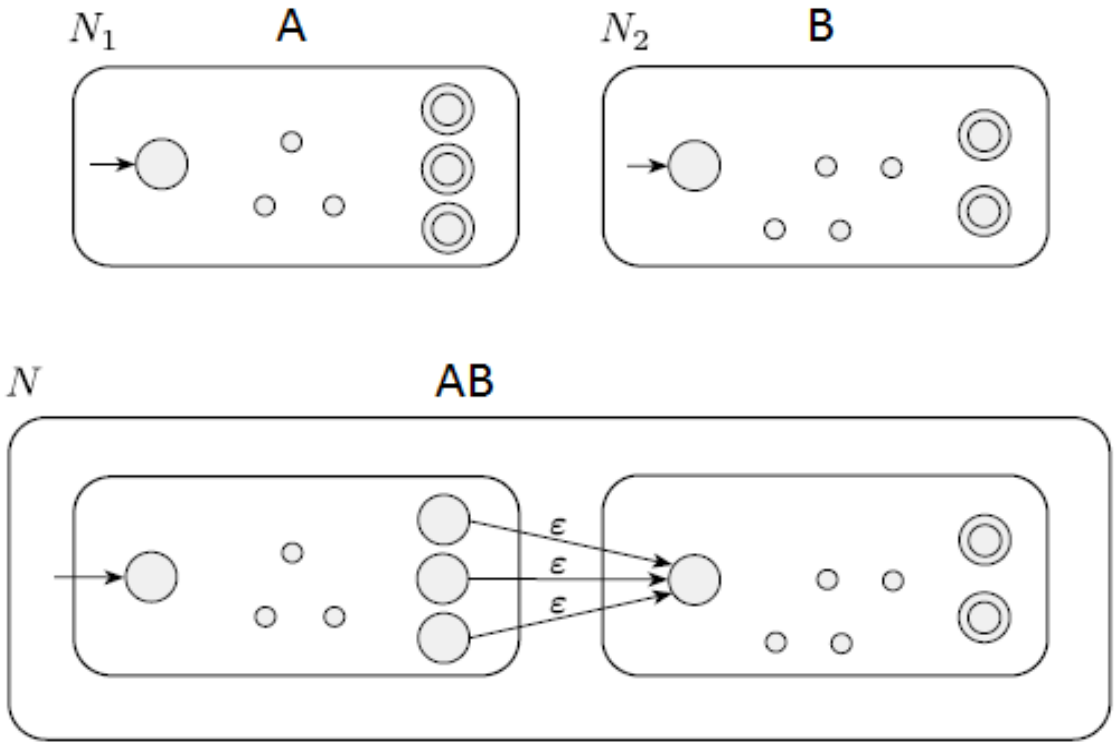
یادآوری: زبان L منظم است اگر و فقط اگر توسط یک nfa (ماشین متناهی غیر قطعی) پذیرفته شود.

فرض کنید زبان A توسط ماشین متناهی N_1 پذیرفته می‌شود و زبان B توسط ماشین متناهی N_2. برای ساخت ماشین متناهی N برای زبان $A \cup B$ یک وضعیت اولیه جدید به ترکیب دو ماشین N_1 و N_2 اضافه می‌کنیم و از آن یک فلش ϵ به وضعیتهای اولیه N_1 و N_2 می‌گذاریم.



بسته بودن زبانهای منظم تحت عملگر اتصال

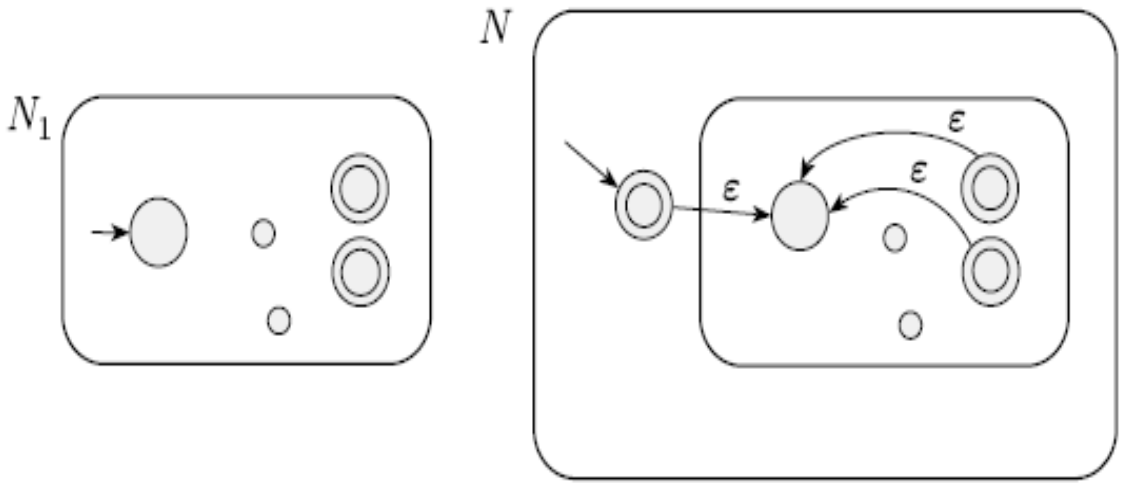
از همه وضعیتهای پذیرش N_1 یک فلش ϵ به وضعیت اولیه N_2 قرار می‌دهیم.
وضعیتهای پذیرش N_1 را هم از حالت پذیرش خارج می‌کنیم.



بسته بودن زبانهای منظم تحت عملگر ستاره

میخواهیم با استفاده از ماشین N_1 برای زبان منظم A یک ماشین متناهی برای زبان A^* بسازیم.

از وضعیتهای پذیرش N_1 یک فلش ϵ به سمت وضعیت شروع N_1 قرار می‌دهیم. می‌دانیم که زبان A^* لزوما شامل رشته تهی است. چون ممکن است زبان A شامل رشته تهی نباشد، برای رفع این مشکل، یک وضعیت اولیه به ماشین N_1 اضافه می‌کنیم و وضعیت اولیه جدید را وضعیت پذیرش می‌کنیم. از وضعیت اولیه جدید هم یک فلش ϵ به وضعیت اولیه قبلی قرار می‌دهیم.



عبارات منظم

عبارات منظم یک شیوه دیگر بیان زبانهای منظم هستند. برای مثال زبان منظم رشته‌های با طول زوج را می‌توان بصورت $(\Sigma\Sigma)^*$ بیان کرد. این یک نمونه از عبارات منظم است.

تعریف: گوییم R یک عبارت منظم است اگر بصورت یکی از حالات زیر باشد:

- تک حرف a از الفبای Σ.
- ϵ
- $\emptyset$
- $(R_1 + R_2)$ وقتی که R_1 و R_2 خود عبارات منظم باشند. دقت کنید کتاب مرجع از علامت $\cup$ بجای $+$ استفاده کرده است. در اینجا علامت $+$ به معنی یا و اجتماع است.
- $(R_1 R_2)$ وقتی که R_1 و R_2 خود عبارات منظم باشند.
- R^* وقتی که R یک عبارت منظم باشد.

عبارات منظم

تفسیر: $(R_1 + R_2)$ زبانی را بیان می‌کند که حاصل اجتماع دو زبان است که توسط عبارات منظم R_1 و R_2 بیان شده است. به عبارت دیگر اگر $L(R_1)$ زبان مربوط به عبارت R_1 و $L(R_2)$ زبان مربوط به عبارت R_2 باشد آنگاه

$$L(R_1 + R_2) = L(R_1) \cup L(R_2)$$

تفسیر: (R_1R_2) زبانی را بیان می‌کند که حاصل از اتصال زبانهای عبارات منظم R_1 و R_2 است. به عبارت دیگر،

$$L(R_1R_2) = L(R_1)L(R_2)$$

تفسیر: R^* زبانی را بیان می‌کند که حاصل از اعمال عملگر ستاره روی زبان مربوط به عبارت منظم R است. به عبارت دیگر

$$L(R^*) = L(R)^*$$

چند مثال

در مثالهای زیر الفبا را $\Sigma = \{a, b\}$ فرض گرفته‌ایم.

- b^*ab^*
- $\Sigma^*a\Sigma^*$
- $\Sigma^*bba\Sigma^*$
- a^+
- $a^*(ba^+)^*$
- $(\Sigma\Sigma\Sigma)^*$
- $ab + ba$
- $a\Sigma^*a + b\Sigma^*b + a + b$
- $(b + \epsilon)a^*$
- $a^*\emptyset$
- $(\emptyset)^*$

- b^*ab^* همه رشته‌هایی که دقیقا یک a دارند.
- $\Sigma^*a\Sigma^*$ همه رشته‌هایی که حداقل یک a دارند.
- $\Sigma^*bba\Sigma^*$ همه رشته‌هایی که شامل زیررشته bba هستند.
- a^+ رشته‌های غیر تهی که فقط شامل a باشند.
- $a^*(ba^+)^*$ هرجا b آمد حداقل یک a بلافاصله بعدش بیاید.
- $(\Sigma\Sigma\Sigma)^*$ رشته‌هایی که طولشان مضربی از 3 است.
- $ab + ba$ $\{ab, ba\}$
- $a\Sigma^*a + b\Sigma^*b + a + b$

رشته‌هایی که اول و آخر رشته مثل هم باشد (شامل رشته تهی نیست.)

- $ba^* + a^* = (b + \epsilon)a^*$
- $a^*\emptyset$ $\emptyset$
- $(\emptyset)^*$ $\{\epsilon\}$

دقت کنید $A^* = \{\epsilon, A, AA, AAA, \ldots\}$

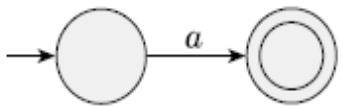
عبارات منظم و ماشینهای متناهی

قضیه: اگر R یک عبارت منظم باشد، آنگاه $L(R)$ را یک زبان منظم است.

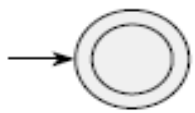
اثبات: نشان می‌دهیم یک ماشین متناهی nfa برای $L(R)$ می‌توان ساخت. حالات مختلف را بررسی می‌کنیم.

طبق تعریف، عبارات منظم R می‌تواند به یکی از حالات زیر باشد.

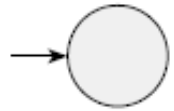
- تک حرف a از الفبای Σ.



- ϵ



- $\emptyset$



- $R_1 + R_2$

مشابه اثبات بسته بودن زبانهای منظم تحت اجتماع

- $R_1 R_2$

مشابه اثبات بسته بودن زبانهای منظم تحت اتصال

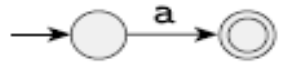
- R^*

مشابه اثبات بسته بودن زبانهای منظم تحت عملگر ستاره

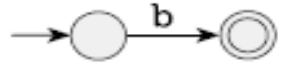
یک مثال

$(ab \cup a)^*$

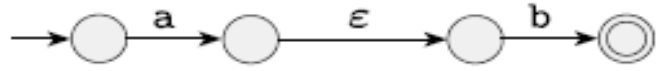
a



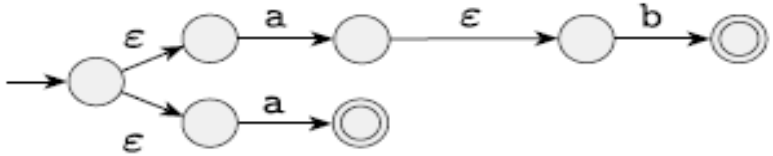
b



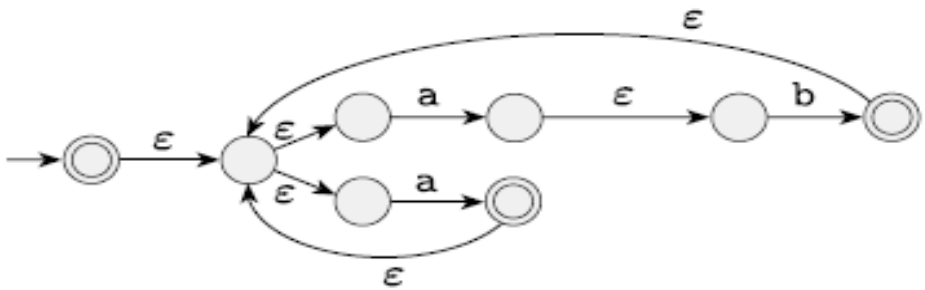
ab



$ab \cup a$



$(ab \cup a)^*$



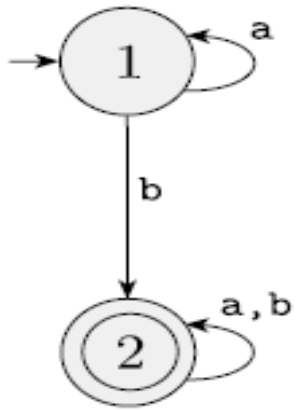
عبارات منظم و ماشینهای متناهی

قضیه: اگر M یک ماشین متناهی باشد، آنگاه عبارت منظم R وجود دارد که معادل $L(M)$ است.

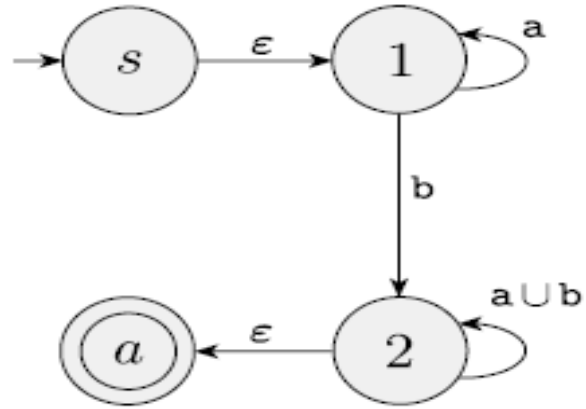
اثبات کامل قضیه در کتاب موجود است.

خلاصه پروسه تبدیل ماشین متناهی به عبارت منظم: می‌توانیم فرض کنیم ماشین تنها یک وضعیت پذیرش دارد (چرا؟) ماشین ابتدا تبدیل به یک gnfa می‌شود که نوعی از nfa است که ساختار ویژه‌ای دارد. در هر مرحله یک وضعیت (غیر از وضعیت پذیرش و وضعیت شروع) حذف می‌شود و بجای آن برچسپ روی فلشها تغییر می‌کند. در انتها دو وضعیت شروع و وضعیت پذیرش با یک فلش بین آنها باقی می‌مانند. برچسپ روی فلش عبارت منظم حاصل را نشان می‌دهد.

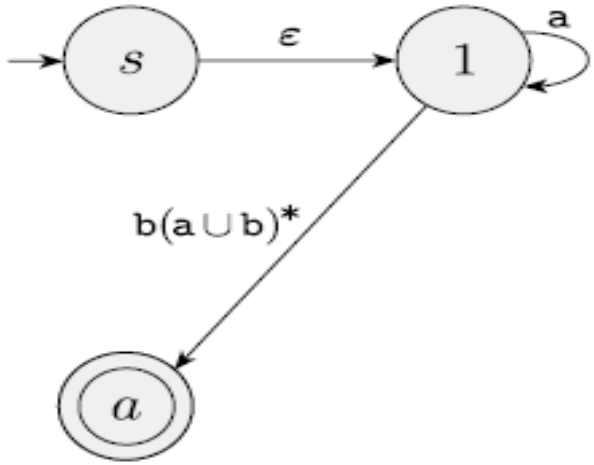
یک مثال از تبدیل ماشین متناهی به عبارت منظم



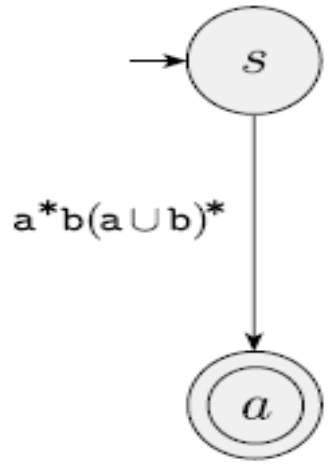
(a)



(b)



(c)



(d)