دانشکده‌ی علوم ریاضی

نظریه‌ی زبان‌ها و اتوماتا | ۵ دی ۱۳۹۱

جلسه‌ی ۲۷: تصمیم‌پذیری

مدرّس: دکتر شهرام خزائی | *نگارنده: امیربهشاد شهراسبی*

۱ معرّفی زبان‌های L_u و L_d

ما علاقه‌مند به حل این مسئله هستیم که "آیا ماشین تورینگ M رشته‌ی ω را می‌پذیرد؟"؛ بدین منظور زبان L_u را به شکل زیر معرفی می‌کنیم:

تعریف ۱ *زبان L_u را به شکل زیر تعریف می‌کنیم:*

$$L_u = \{M111\omega | \omega \in L(M)\}$$

با توجه به اینکه در کد ماشین تورینگ (M) زیر رشته‌ی ۱۱۱ نداریم، تعریف فوق فاقد ابهام است.
در جلسه‌ی قبل ماشین تورینگی به نام ماشین تورینگ جهانی [1] ارائه کردیم که زبان L_u را می‌پذیرد. در واقع ماشین تورینگ جهانی، ماشین تورینگی است که می‌تواند با گرفتن کد یک ماشین تورینگ و رشته‌ی ورودی، رفتار آن ماشین تورینگ نسبت به ورودی را شبیه‌سازی کند.

در جلسه‌ی گذشته دیدیم مجموعه‌ی تمام رشته‌های تحت الفبای Σ و مجموعه‌ی تمام ماشین‌های تورینگ شمارش‌پذیر هستند. بنابراین تعاریف زیر قابل ارائه هستند:

تعریف ۲ *ω_i را iاُمین رشته تحت Σ تعریف می‌کنیم.*

تعریف ۳ *M_i را ماشین تورینگی تعریف می‌کنیم که کد آن برابر ω_i است. در صورتیکه ω_i کد معتبری برای هیچ ماشین تورینگی نبود، M_i را ماشین تورینگی تعریف می‌کنیم که هیچ رشته‌ای را نمی‌پذیرد.*

اکنون زبان L_d به شکل زیر قابل تعریف است:

تعریف ۴ *L_d را مجموعه تمام رشته‌هایی مانند ω تعریف می‌کنیم که اگر $\omega = \omega_i$، آنگاه $\omega \notin L(M_i)$.*

$$L_d = \{w | w = w_i \wedge w \notin L(M_i)\}$$

[1] Universal Turing Machine

قضیه ۱ *L_d بازگشتی برشمردنی*[2] *نیست.*

برهان. برهان خلف؛ فرض کنید که ماشین تورینگی وجود داشته باشد که L_d را بپذیرد، یعنی:

$$\exists j \quad L_d = L(M_j)$$

آنگاه دو حالت داریم؛
حالت نخست:

$$\omega_j \in L_d \Rightarrow \omega_j \notin L(M_j) \Rightarrow L_d \neq L(M_j)$$

حالت دوم:

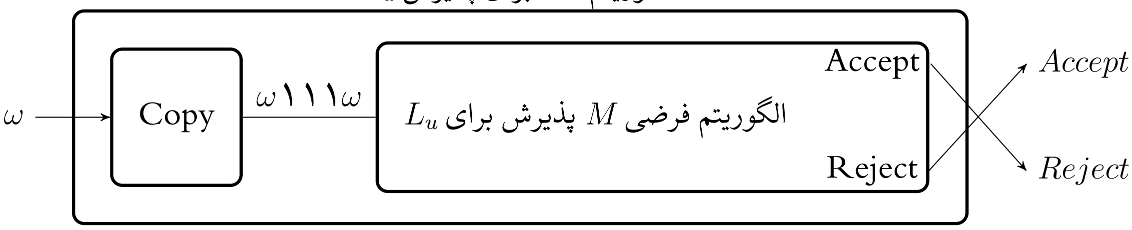
$$\omega_j \notin L_d \Rightarrow \omega_j \in L(M_j) \Rightarrow L_d \neq L(M_j)$$

در هر دو حالت به تناقض رسیدیم، لذا فرض خلف باطل بوده و L_d بازگشتی برشمردنی نیست. ■

قضیه ۲ *L_u بازگشتی برشمردنی هست، ولی بازگشتی*[3] *نیست.*

برهان. L_u بازگشتی برشمردنی هست زیرا ماشین تورینگ جهانی برای پذیرش آن وجود دارد. برای اثبات بازگشتی نبودن آن، از برهان خلف استفاده می‌کنیم:
فرض خلف: فرض کنید L_u بازگشتی باشد، پس یک الگوریتم مانند M برای پذیرش L_u وجود دارد. با استفاده از M، الگوریتم M' را به شکل زیر را برای پذیرش L_d ارائه می‌کنیم:

الگوریتم M' برای پذیرش L_d



ادعا می‌کنیم که الگوریتم M'، زبان $_d$ را می‌پذیرد. داریم:

$$\omega \in L_d \Rightarrow \omega \notin L(\omega) \Rightarrow \omega۱۱۱\omega \notin L_u$$

پس M رشته‌ی $\omega۱۱۱\omega$ را نمی‌پذیرد و لذا M' رشته‌ی ω را می‌پذیرد.
همچنین:

$$\omega \notin L_d \Rightarrow \omega \in L(\omega) \Rightarrow \omega۱۱۱\omega \in L_u$$

پس M رشته‌ی $\omega۱۱۱\omega$ را می‌پذیرد و لذا M' رشته‌ی ω را نمی‌پذیرد.
اما می‌دانیم L_d بازگشتی برشمردنی نیست، لذا فرض خلف باطل بوده و حکم ثابت می‌شود. ■

تعریف ۵ *می‌گوییم مسئله‌ی P_1 به مسئله‌ی P_2 کاهش می‌یابد، اگر با فرض وجود الگوریتمی برای حل مسئله‌ی P_2، بتوان الگوریتمی برای مسئله‌ی P_1 ارائه کرد. (حل کردن P_1 سخت‌تر از حل کردن P_2 نیست)*

قضیه ۳ *اگر مسئله‌ی P_1 به مسئله‌ی P_2 کاهش یابد و P_1 تصمیم ناپذیر باشد، P_2 نیز تصمیم‌ناپذیر است.*

[2] Recursively Enumerable
[3] Recursive

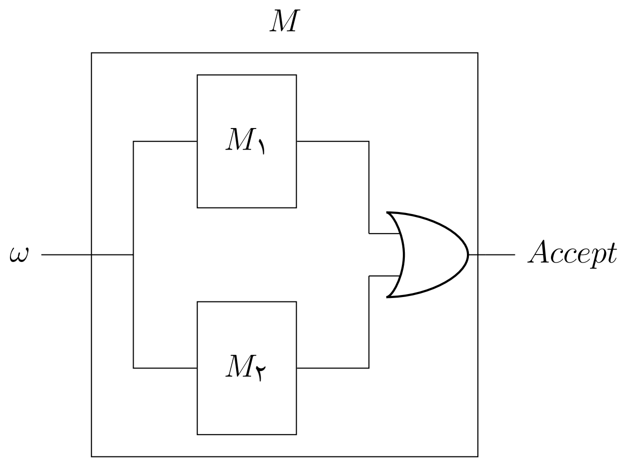
۲ ویژگی‌های بستاری

در این مبحث بسته بودن مجموعه زبان‌های بازگشتی برشمردنی و بازگشتی تحت اعمالی مانند اجتماع، الحاق، ستاره کلینی، معکوس، اشتراک، معکوس یکریختی، یکریختی، متمّم و تفاصل بررسی می‌شود.
در ادامه به عنوان نمونه چند مورد از موارد فوق را بررسی می‌کنیم:

۱.۲ اجتماع

قضیه ۴ *اگر L_1 و L_2 بازگشتی برشمردنی باشند، آنگاه $L_1 \cup L_2$ نیز بازگشتی برشمردنی است.*

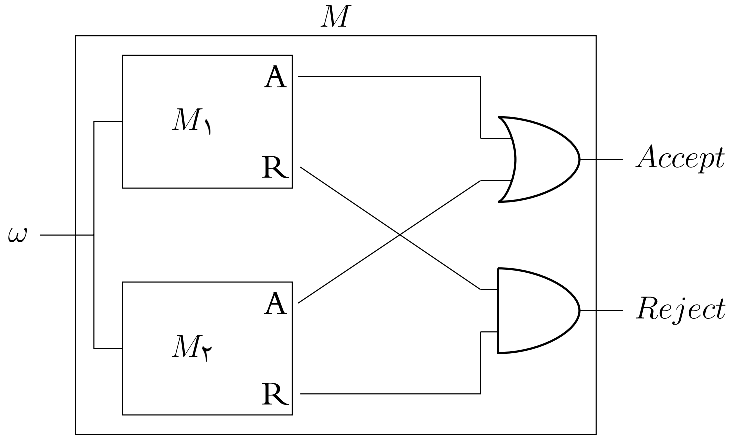
برهان. اگر L_1 و L_2 زبان‌های بازگشتی برشمردنی باشند، آنگاه ماشین‌های تورینگ M_1 و M_2 وجود دارند که L_1 و L_2 را بپذیرند. ماشین تورینگ M را طوری می‌سازیم که M_1 و M_2 را به صورت موازی شبیه‌سازی کند و به محض اینکه یکی از دو ماشین آنرا پذیرفتند، M نیز آن را پذیرفته و متوقّف شود:



■

قضیه ۵ *اگر L_1 و L_2 بازگشتی باشند، آنگاه $L_1 \cup L_2$ نیز بازگشتی است.*

برهان. اگر L_1 و L_2 بازگشتی باشند، الگوریتم‌هایی برای پذیرش L_1 و L_2 مثل M_1 و M_2 وجود دارند. الگوریتم M را بدین ترتیب ارائه می‌کنیم که الگوریتم‌های M_1 و M_2 را شبیه‌سازی کند (چون هر دو الگوریتم در زمان محدود به پایان می‌رسند، شبیه‌سازی می‌تواند به صورت سری یا موازی انجام شود). سپس در صورتی که خروجی حداقل یکی از الگوریتم‌ها $Accept$ بود، $\omega \in L_1 \cup L_2$ و خروجی $Accept$ می‌شود. در غیر اینصورت خروجی $Reject$ می‌شود.



به شکل دقیق‌تر:

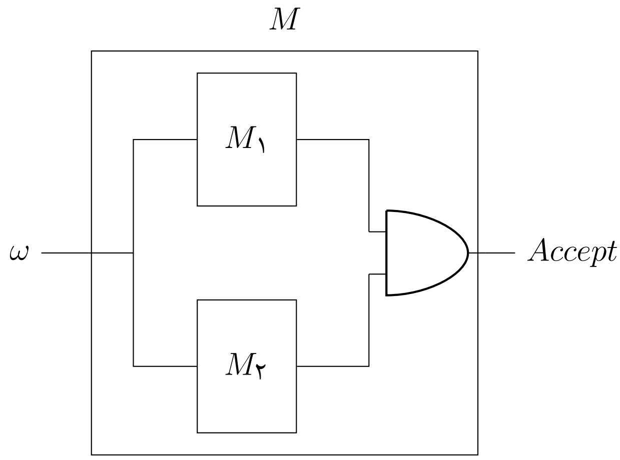
$$Accept \Rightarrow Accept_۱ \vee Accept_۲ \Rightarrow (\omega \in L(M_۱)) \vee (\omega \in L(M_۲))) \Rightarrow \omega \in (L(M_۱) \cup L(M_۲))$$
$$Reject \Rightarrow Reject_۱ \wedge Reject_۲ \Rightarrow (\omega \notin L(M_۱)) \wedge (\omega \notin L(M_۲))) \Rightarrow \omega \notin (L(M_۱) \cup L(M_۲))$$

دقت کنید چون $M_۱$ و $M_۲$ در زمان محدود به پایان می‌رسند، الگوریتم فوق نیز در زمان محدود به پایان می‌رسد.
■

۲.۲ اشتراک

قضیه ۶ *اگر $L_۱$ و $L_۲$ بازگشتی برشمردنی باشند، آنگاه $L_۱ \cap L_۲$ نیز بازگشتی برشمردنی است.*

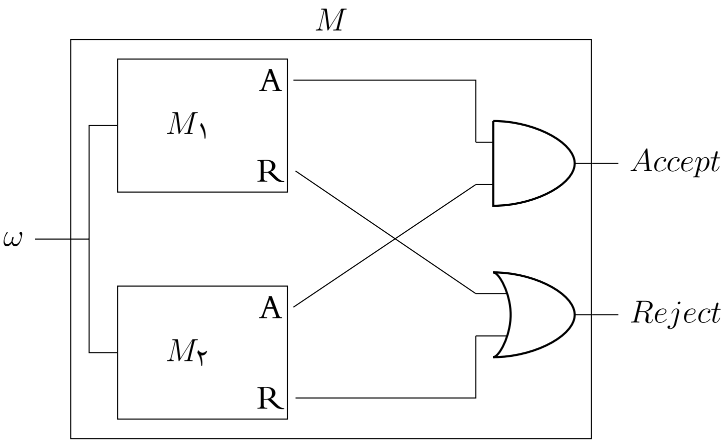
برهان. اگر $L_۱$ و $L_۲$ زبان‌های بازگشتی برشمردنی باشند، آنگاه ماشین‌های تورینگ $M_۱$ و $M_۲$ وجود دارند که $L_۱$ و $L_۲$ را بپذیرند. ماشین تورینگ M را طوری می‌سازیم که $M_۱$ و $M_۲$ را به صورت موازی شبیه‌سازی کند و به محض اینکه هر دو ماشین آنرا پذیرفتند، M نیز آن را پذیرفته و متوقّف شود:



■

قضیه ۷ *اگر $L_۱$ و $L_۲$ بازگشتی باشند، آنگاه $L_۱ \cap L_۲$ نیز بازگشتی است.*

برهان. اگر $L_۱$ و $L_۲$ بازگشتی باشند، الگوریتم‌هایی برای پذیرش $L_۱$ و $L_۲$ مثل $M_۱$ و $M_۲$ وجود دارند. الگوریتم M را بدین ترتیب ارائه می‌کنیم که الگوریتم‌های $M_۱$ و $M_۲$ را شبیه‌سازی کند (چون هر دو الگوریتم در زمان محدود به پایان می‌رسند، شبیه‌سازی می‌تواند به صورت سری یا موازی انجام شود). سپس در صورتی که خروجی هر دو الگوریتم $Accept$ بود، $\omega \in L_۱ \cap L_۲$ و خروجی $Accept$ می‌شود. در غیر اینصورت خروجی $Reject$ می‌شود.



به شکل دقیق‌تر:

$$Accept \Rightarrow Accept_۱ \wedge Accept_۲ \Rightarrow (\omega \in L(M_۱)) \wedge (\omega \in L(M_۲))) \Rightarrow \omega \in (L(M_۱) \cap L(M_۲))$$
$$Reject \Rightarrow Reject_۱ \vee Reject_۲ \Rightarrow (\omega \notin L(M_۱)) \vee (\omega \notin L(M_۲))) \Rightarrow \omega \notin (L(M_۱) \cap L(M_۲))$$

دقت کنید چون $M_۱$ و $M_۲$ در زمان محدود به پایان می‌رسند، الگوریتم فوق نیز در زمان محدود به پایان می‌رسد. ■

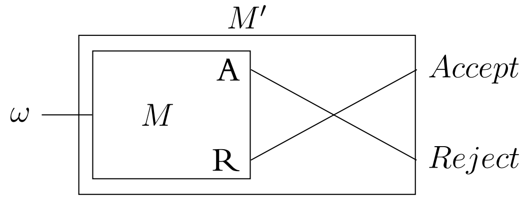
۳.۲ متمّم

قضیه ۸ *مجموعه‌ی زبان‌های بازگشتی برشمردنی نسبت به متمّم بسته نیستند.*

برهان. برای اثبات این امر کافیست یک زبان ارائه کنیم که خود بازگشتی برشمردنی باشد و متمّم آن نباشد؛ L_u این خاصیت را دارد. ■

قضیه ۹ *مجموعه‌ی زبان‌های بازگشتی نسبت به متمّم بسته هستند.*

برهان. اگر L یک زبان بازگشتی باشد، آنگاه الگوریتم M برای پذیرش L وجود دارد. الگوریتم M' را برای پذیرش $\bar{L}$ به شکل زیر ارائه می‌کنیم. برای بررسی اینکه آیا ω عضو $\bar{L}$ هست، الگوریتم M را با ورودی ω شبیه‌سازی می‌کنیم. اگر خروجی الگوریتم M، $Accept$ باشد، ω عضو L هست و در نتیجه عضو $\bar{L}$ نیست. بنابرین پاسخ M' باید $Reject$ باشد. در غیر اینصورت ω عضو L نیست و در نتیجه عضو $\bar{L}$ هست، پس پاسخ M' باید $Reject$ باشد.



■

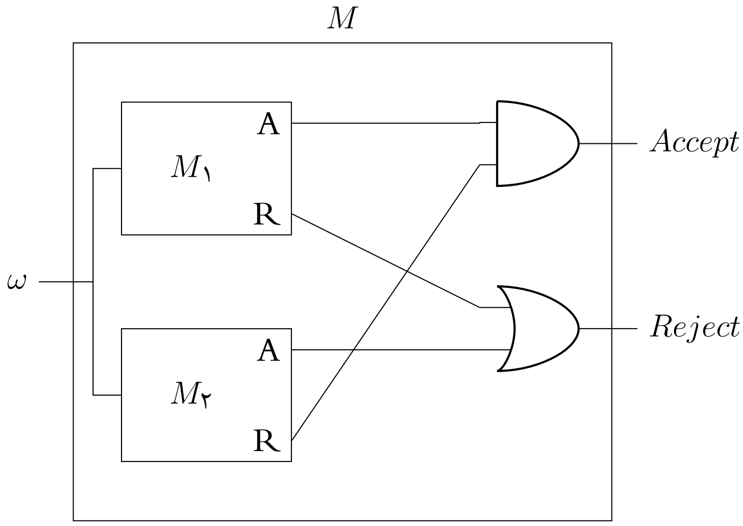
۴.۲ تفاضل

قضیه ۱۰ *مجموعه‌ی زبان‌های بازگشتی برشمردنی نسبت به تفاضل بسته نیست.*

برهان. فرض کنید $L_۱ = (۰ + ۱)^*$ و $L_۲ = L_u$، آنگاه $L_۱ - L_۲ = \bar{L_u}$؛ در اینصورت $L_۱$ و $L_۲$ بازگشتی برشمردنی هستند، حال آنکه $L_۱ - L_۲$ بازگشتی برشمردنی نیست. ■

قضیه ۱۱ *مجموعه‌ی زبان‌های بازگشتی نسبت به تفاضل بسته است.*

برهان. اگر $L_۱$ و $L_۲$ بازگشتی باشند، الگوریتم‌های $M_۱$ و $M_۲$ برای پذیرش $L_۱$ و $L_۲$ وجود دارند. الگوریتم M را برای پذیرش $L_۱ - L_۲$ بدین ترتیب ارائه می‌کنیم که الگوریتم‌های $M_۱$ و $M_۲$ را شبیه‌سازی کند (چون هر دو الگوریتم در زمان محدود به پایان می‌رسند، شبیه‌سازی می‌تواند به صورت سری یا موازی انجام شود). سپس در صورتی که خروجی $M_۱$، $Accept$ و خروجی $M_۲$، $Reject$ باشد، $\omega \in L_۱ - L_۲$ و خروجی $Accept$ می‌شود. در غیر اینصورت خروجی $Reject$ می‌شود.



■

۵.۲ الحاق

قضیه ۱۲ *مجموعه‌ی زبان‌های بازگشتی برشمردنی نسبت به الحاق بسته است.*

برهان. اگر $L_۱$ و $L_۲$ بازگشتی برشمردنی باشند، آنگاه ماشین‌های تورینگ $M_۱$ و $M_۲$ برای پذیرش آنها وجود دارد. یک ماشین تورینگ M ارائه می‌کنیم که در آن رفتار ماشین‌های تورینگ $M_۱$ و $M_۲$ نسبت به ورودی‌های x و y به ازای هر $\omega = xy$ به صورت موازی شبیه‌سازی می‌شود. اگر به ازای یکی از x و y ها، $M_۱$ و $M_۲$، x و y را بپذیرند، $x \in L_۱$ و $y \in L_۲$ است. پس $\omega = xy \in L_۱L_۲$. لذا M با پذیرفتن ω متوقف می‌شود. بدین ترتیب ماشین تورینگی برای $L_۱L_۲$ بدست می‌آید و بنابرین $L_۱L_۲$ بازگشتی برشمردنی است. ■

قضیه ۱۳ *مجموعه‌ی زبان‌های بازگشتی نسبت به الحاق بسته است.*

برهان. اگر $L_۱$ و $L_۲$ زبان‌های بازگشتی باشند، مسئله‌ی عضویت برای آنها تصمیم‌پذیر است، لذا برای بررسی عضویت یک رشته مانند ω در $L_۱L_۲$، کافیست برای تمام x و y های ممکن که $\omega = xy$، مسائل $x \in L_۱$ و $y \in L_۲$ را بررسی کنیم؛ که به علت محدود بودن تعداد حالات، در زمان متناهی به جواب می‌رسد. بنابرین مجموعه‌ی زبان‌های بازگشتی نسبت به الحاق بسته است. ■

۶.۲ ستاره‌ی کلینی

قضیه ۱۴ *مجموعه‌ی زبان‌های بازگشتی برشمردنی نسبت به عمل ستاره‌ی کلینی بسته است.*

برهان. اگر L بازگشتی برشمردنی باشند، آنگاه ماشین‌های تورینگ M برای پذیرش آنها وجود دارد. در اثبات این قضیه هم کافیست مانند الحاق، برای تمام $x_۱$، $x_۲$، ... و x_k که $\omega = x_۱x_۲...x_k$، رفتار M را نسبت به همه‌ی x_i ها به شکل موازی شبیه‌سازی کنیم. اگر به ازای یک تفکیک، M تمام x_i را بپذیرد، ω پذیرفته می‌شود. بدین ترتیب ماشین تورینگی برای L^* بدست می‌آید که بازگشتی برشمردنی بودن L^* را اثبات می‌کند. ■

قضیه ۱۵ *مجموعه‌ی زبان‌های بازگشتی نسبت به عمل ستاره‌ی کلینی بسته است.*

برهان. برای ستاره‌ی کلینی نیز در زبان‌های بازگشتی، مشابه الحاق، تمام حالات تفکیک یک رشته (مانند ω) به تعدادی زیر رشته را بررسی کرده (مثل $\omega = x_۱x_۲...x_k$) و در هر حالت، با استفاده از الگوریتم مسئله‌ی عضویت برای هر قسمت (آیا x_i عضو L هست؟)، امکان عضویت ω در L^* را بررسی می‌کنیم. مجددا به علت متناهی بودن $|\omega|$، الگوریتم در زمان متناهی و مستقل از پاسخ پایان می‌پذیرد. لذا مجموعه‌ی زبان‌های بازگشتی، تحت ستاره‌ی کلینی نیز بسته هستند. ■