مبانی نظریه محاسبه

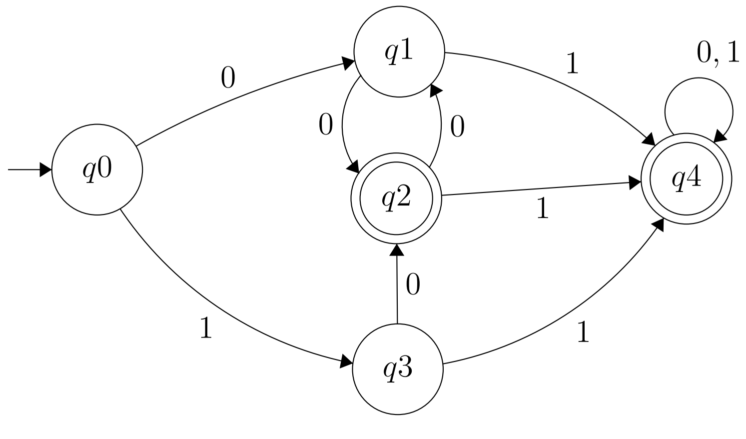
دانشکده ریاضی. دانشگاه صنعتی خواجه نصیرالدین طوسی. **ترم ۴۰۲۲**

۱ ماشین متناهی کمینه

فرض کنید ماشین متناهی $M = (Q, \Sigma, \delta, q_0, F)$ داده شده است. می‌خواهیم ماشین متناهی M' را بسازیم بطوریکه $L(M') = L(M)$ و ماشین M' کمترین تعداد وضعیت را داشته باشد (به عبارت دیگر از لحاظ تعداد وضعیت کمینه باشد.)

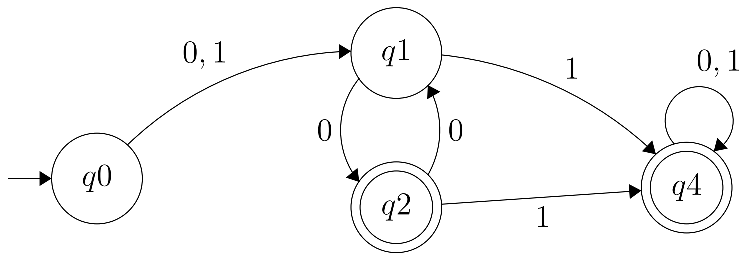
تعریف: در ماشین $M = (Q, \Sigma, \delta, q_0, F)$ زوج وضعیت (p, q) را قابل ادغام گوییم اگر برای هر رشته $w \in \Sigma^*$ داشته باشیم $\delta^*(p, w) \in F$ اگر و فقط اگر $\delta^*(q, w) \in F$. به عبارت دیگر با شروع از وضعیت p و تغذیه ماشین با رشته w به وضعیت پذیرش برویم اگر و فقط اگر با همان رشته اگر از وضعیت q شروع کنیم به یک وضعیت پذیرش برویم.

وقتی که زوج وضعیت (p, q)قابل ادغام باشند، گویی مثل هم عمل می‌کنند و از لحاظ پذیرش و عدم پذیرش رشته ها کارکرد یکسان دارند. پس می‌توان این دو وضعیت را در یک وضعیت ادغام کرد بدون اینکه زبان ماشین تغییر کند. به مثال زیر توجه کنید.



زوج وضعیت $(q1, q3)$ قابل ادغام هستند. در هر کدام از این وضعیتها باشیم، به محض اینکه حرف 1 بیاید به q_4 برویم. اگر در $q1$ یا $q3$ باشیم و به تعداد فرد 0 بیاید در وضعیت $q2$ قرار می‌گیریم که یک وضعیت پذیرش است. اگر به تعداد زوج 0 بیاید در وضعیت $q1$ قرار می‌گیریم. اگر رشته ϵ بیاید (یعنی در همانجا بمانیم) چون $q1$ و $q3$ هر دو وضعیتهای عدم پذیرش هستند پس از نظر این هم قابل ادغام هستند.

پس می‌توان دو وضعیت $q1$ و $q3$ را در هم ادغام کرد و ماشین جدیدی بدست آورد که همان زبان را می‌پذیرد ولی یک وضعیت کمتر دارد.



دقت کنید قابلیت ادغام یک رابطه هم ارزی است. اگر p و q قابل ادغام باشند، و زوج q و r هم قابل ادغام باشند، آنگاه زوج p و r هم قابل ادغام خواهند بود. لذا می‌توان وضعیتهای یک dfa را از نظر قابلیت ادغام به کلاسهای هم ارزی افراز کرد.

۱.۱ یک الگوریتم برای پیدا کردن همه زوج وضعیتهای غیر قابل ادغام

الگوریتم زیر همه زوج وضعیتهای غیر قابل ادغام در ماشین متناهی M را پیدا می‌کند.

۱. در ابتدا همه وضعیتهای غیر قابل دسترسی از وضعیت شروع را حذف می‌کنیم.

۲. همه وضعیت زوجهای (p,q) را در نظر می‌گیریم. اگر $p \in F$ در حالیکه $q \notin F$ (و یا برعکس) زوج وضعیت (p,q) را غیر قابل ادغام علامت می‌زنیم.

۳. قدم زیر را تکرار کن. کار را متوقف کن وقتی در پایان یک دور هیچ زوج جدیدی علامت زده نشود.

   (آ) برای هر زوج (p,q) و حرف الفبا $a \in \Sigma$ تغییر وضعیت $\delta(p,a) = p_a$ و $\delta(q,a) = q_a$ را محاسبه کن. اگر زوج (p_a, q_a) قبلا علامت خورده باشند (غیر قابل ادغام باشند) آنگاه زوج (p,q) را علامت بزن (به عنوان غیر قابل ادغام گزارش کن).

لم: الگوریتم بالا همه زوج وضعیتهای غیر قابل ادغام را پیدا می‌کند.

اثبات: دقت کنید دو وضعیت (p,q) اگر غیر قابل ادغام باشند، پس رشته w وجود دارد که $\sigma^*(p,w) \in F$ و $\sigma^*(q,w) \notin F$ و یا برعکس. فرض کنید w از میان همه رشته‌هایی که p و q را جدا می‌کنند، کمترین طول را داشته باشد و داشته باشیم $n = |w|$. اگر حرف اول w حرف a باشد آنگاه دو وضعیت (p', q') وجود دارند بطوریکه $\delta(p,a) = p'$ و $\delta(q,a) = q'$. اگر $w = ay$ آنگاه رشته y دو وضعیت (p', q') را از هم غیر قابل ادغام می‌کند و رشته y کوتاهترین رشته ای است که این کار را انجام می‌دهد.
با این مقدمات می‌خواهیم ادعا کنیم اگر بعد از دور n ام از قدم سوم الگوریتم، اگر دو وضعیت وجود داشته باشند که با یک رشته با طول حداکثر n غیر قابل ادغام شده‌اند، تا اینجا علامت خورده‌اند. این را می‌توان با استقرا اثبات کرد. پایه استقرا را قدم دوم الگوریتم برآورده می‌کند. اگر دو وضعیت (p,q) توسط رشته تهی غیر

قابل ادغام باشند، این به این معنی است که یکی عضو پذیرش است و دیگر عضو پذیرش نیست. این زوجها در قدم دوم علامت زده شده‌اند. با فرض استقرا در پایان دور $n-1$ همه زوج وضعیتهایی که توسط یک رشته با طول حداکثر $n-1$ تفکیک شده‌اند علامت زده شده‌اند. پس اگر زوج (p,q) توسط رشته با طول n غیر قابل ادغام شوند، در دور بعدی علامت می‌خورند.

۲.۱ الگوریتمی برای ساختن ماشینی با کمترین تعداد وضعیت

فرض کنید با استفاده از الگوریتمی که در قسمت قبل ارائه شد، همه زوج وضعیتهای غیر قابل ادغام را پیدا کنیم. همان طور که گفتیم رابطه قابلیت ادغام یک رابطه هم ارزی است. با استفاده از نتایج الگوریتم قسمت قبلی می‌توانیم کلاسهای هم ارزی این رابطه را پیدا کنیم. لذا وضعیتهای ماشین $M=(Q,\Sigma,\delta,q_0,F)$ به یک سری زیر مجموعه افراز می‌شود. فرض کنید $Q_1,\cdots,Q_k$ افراز مورد نظر باشد. دقت کنید وضعیتهایی که داخل زیرممجموعه Q_i هستند قابل ادغام هستند. ماشین $M'=(Q',\Sigma,\delta',q'_0,F')$ را بصورت زیر می‌سازیم:

۱. وضعیتهای ماشین M' همان زیرمجموعه‌های $Q_1,\cdots,Q_k$ هستند. لذا $Q'=\{Q_1,\cdots,Q_k\}$.

۲. برای هر Q_i و حرف $a\in\Sigma$ تعریف می‌کنیم

$$\delta'(Q_i,a)=Q_j$$

به شرطی که دو وضعیت $p\in Q_i$ و $q\in Q_j$ وجود داشته باشد بطوریکه $\delta(p,a)=q$.

۳. وضعیت شروع ماشین M' زیرمجموعه Q_i است بطوریکه $q_0\in Q_i$

۴. وضعیت Q_i یک وضعیت پذیرش در ماشین M' خواهد بود اگر و فقط Q_i عضوی داشته باشد که در ماشین M وضعیت پذیرش باشد.

می‌توانیم دوباره از استقرا استفاده کنیم که نشان دهیم که زبان ماشین M و M' معادل هستند.

قضیه: ماشین M' که توسط الگوریتم بالا ساخته می‌شود ماشینی است با کمترین تعداد وضعیت برای زبان $L(M')$.

اثبات: فرض کنید ماشین M' وضعیتهایش $p_0,p_1,\cdots,p_m$ باشند و تابع تغییر وضعیتش هم δ' باشد. فرض کنید ماشین M_1 با تابع تغییر وضعیت δ_1 و وضعیت شروع q_0 وجود داشته بطوریکه $L(M_1)=L(M')$ و تعداد وضعیتهای M_1 کمتر از $m+1$ باشد. چون هیچ وضعیت غیر قابل دسترسی در M' وجود ندارد پس باید رشته‌های $w_1,\cdots,w_{m+1}$ وجود داشته باشند بطوریکه برای $i=1\cdots m+1$ داریم

$$\delta'^*(p_0,w_i)=p_i$$

چون M_1 تعداد وضعیت کمتری دارد پس بنا به اصل لانه کبوتر باید دو رشته w_k و w_ℓ در این میان باشند بطوریکه

$$\delta_1^*(q_0,w_k)=\delta_1^*(q_0,w_\ell)$$

چون دو وضعیت p_k و p_ℓ غیر قابل ادغام هستند پس باید رشته x وجود داشته باشد بطوریکه $\delta'^*(p_k,x)\in F$ ولی $\delta'^*(p_\ell,x)\notin F$. و یا برعکس. اینجا F وضعیتهای پذیرش ماشین M' است. به عبارت دیگر رشته

$w_k x$ توسط M' پذیرفته می‌شود در حالی که رشته $w_\ell x$ پذیرفته نمی‌شود. از طرف دیگر داریم $\delta_1^*(q_0, w_k x)$ و $\delta_1^*(q_0, w_\ell x)$ هر دو در ماشین M_1 ختم به یک وضعیت می‌شوند. پس دو ماشین M' و M_1 در مورد رشته‌های $w_k x$ و $w_\ell x$ اختلاف نظر دارند. این یک تناقض است و لذا ماشین M_1 نمی‌تواند وجود داشته باشد.