درس مبانی نظریه محاسبه

جلسه بیست و یکم

تصمیم ناپذیری و روش تقلیل بین مسائل

Undecidability and the reduction method

مسئله A_{TM}

آیا ماشین تورینگ M رشته w را می‌پذیرد؟

$$A_{\mathrm{TM}} = \{\langle M, w\rangle \mid \text{یک ماشین تورینگ است که رشته } w \text{ را می‌پذیرد } M\}$$

در جلسه گذشته نشان دادیم زبان A_{TM} تصمیم ناپذیر است. به عبارت دیگر، با داشتن توصیف ماشین تورینگ M و رشته w، الگوریتمی وجود ندارد که مشخص کند آیا ماشین M رشته w را می‌پذیرد یا نه.

مروری بر اثبات تصمیم ناپذیر بودن A_{TM} :

فرض کردیم الگوریتم H برای مسئله A_{TM} وجود دارد (برهان خلف)

$$H(\langle M, w\rangle) = \begin{cases} \text{stops and accepts} & \text{if } M \text{ accepts } w \\ \text{stops and rejects} & \text{if } M \text{ does not accept } w. \end{cases}$$

با استفاده از H، الگوریتم D را ساختیم که ورودی اش توصیف یک ماشین تورینگ است. اگر ماشین داده شده خودش را قبول کرد، الگوریتم D جواب reject می‌دهد و اگر ماشین داده شده توصیف خودش را قبول نکند، الگوریتم D جواب accept می‌دهد.

$$D(\langle M\rangle) = \begin{cases} \mathit{accept} & \text{if } M \text{ does not accept } \langle M\rangle \\ \mathit{reject} & \text{if } M \text{ accepts } \langle M\rangle. \end{cases}$$

الگوریتم D روی توصیف خودش $\langle D\rangle$ باید بصورت زیر عمل کند.

$$D(\langle D\rangle) = \begin{cases} accept & \text{if } D \text{ does not accept } \langle D\rangle \\ reject & \text{if } D \text{ accepts } \langle D\rangle. \end{cases}$$

اما این یک تناقض است. اگر D توصیف خودش را رد کند باید آن را قبول کند! (به همین ترتیب اگر D توصیف خودش را قبول کند، باید آن را رد کند!)

پس فرض ما مبنی بر وجود الگوریتم H برای زبان A_{TM} باید نادرست باشد. در نتیجه A_{TM} تصمیم ناپذیر است. $\square$

اثبات قبل از دیدگاهی دیگر

فرض کنید ماشینهای تورینگ را به ترتیب بنویسیم $M_1,\ M_2,\ M_3,\ \ldots,$

بر اساس این ترتیب می‌توانیم جدول زیر را تصور کنیم. هر سطر و ستون مرتبط با یک ماشین تورینگ است. در درایه مربوط به سطر i و ستون j نتیجه محاسبات ماشین تورینگ M_i روی توصیف ماشین تورینگ M_j نوشته شده است (نتایج فرضی است). عبارت ∞ به معنی عدم توقف ماشین است.

	$\langle M_1\rangle$	$\langle M_2\rangle$	$\langle M_3\rangle$	$\langle M_4\rangle$	$\cdots$
M_1	*accept*	∞	*accept*	∞	
M_2	*accept*	*accept*	*accept*	*accept*	
M_3	∞	*reject*	∞	*reject*	$\cdots$
M_4	*accept*	*accept*	∞	*reject*	
$\vdots$		$\vdots$			

با استفاده از ماشین تصمیم گیرنده (الگوریتم) H، می‌توانیم حالات عدم توقف را تشخیص دهیم و جدول قبلی را بصورت زیر بروز کنیم.

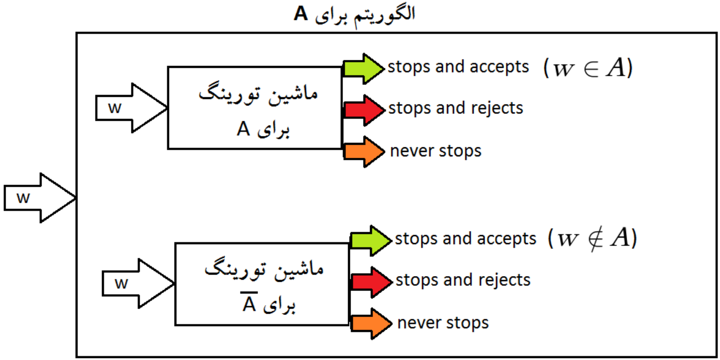
	$\langle M_1\rangle$	$\langle M_2\rangle$	$\langle M_3\rangle$	$\langle M_4\rangle$	$\cdots$	$\langle D\rangle$	$\cdots$
M_1	$\underline{accept}$	$reject$	$accept$	$reject$		$accept$	
M_2	$accept$	$\underline{accept}$	$accept$	$accept$	$\cdots$	$accept$	$\cdots$
M_3	$reject$	$reject$	$\underline{reject}$	$reject$		$reject$	
M_4	$accept$	$accept$	$reject$	$\underline{reject}$		$accept$	
$\vdots$			$\vdots$		$\ddots$		
D	$reject$	$reject$	$accept$	$accept$		$\underline{?}$	
$\vdots$			$\vdots$				$\ddots$

ماشین تورینگ D که قبلا در موردش صحبت کردیم، باید یکی از ماشینهای تورینگ در جدوال بالا باشد (یکی از سطرهای جدول). سوال این است که D روی توصیف خودش چه جوابی می‌دهد؟ هر جوابی که بدهد باعث تناقض خواهد شد. درنتیجه فرض مبنی بر وجود الگوریتم H نادرست است. $\square$

این یک نمونه از روش قطری diagonalization method برای اثبات است.

لم: اگر زبان A و متمم آن $\overline{A}$ هر دو قابل تشخیص با تورینگ باشند آنگاه A تصمیم پذیر است. به عبارت دیگر

$$(A \in \text{ Turing-recognizable}) \text{ and } (\overline{A} \in \text{ Turing-recognizable}) \rightarrow A \in \text{decidable}$$



نتیجه لم قبل: متمم زبان A_{TM} قابل تشخیص با تورینگ نیست. به عبارت دیگر،

$$\overline{A_{\mathrm{TM}}} \notin \text{Turing-recognizable}$$

اثبات: اگر $\overline{A_{\mathrm{TM}}}$ قابل تشخیص با تورینگ باشد، از آنجا که A_{TM} قابل تشخیص با تورینگ است، زبان A تصمیم پذیر خواهد شد. این متناقض با یافته‌های ماست. $\square$

مسائل تصمیم ناپذیر دیگر

با علم به اینکه $A_{\rm TM}$ تصمیم ناپذیر است، می‌توانیم نشان دهیم زبانهای دیگری هم هستند که تصمیم ناپذیرند. این کار را با استفاده از روش تقلیل reduction method انجام می‌دهیم.

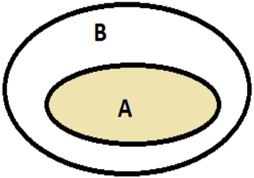
فرض کنید می‌خواهیم نشان دهیم زبان B تصمیم ناپذیر است. چگونه این کار را انجام دهیم؟

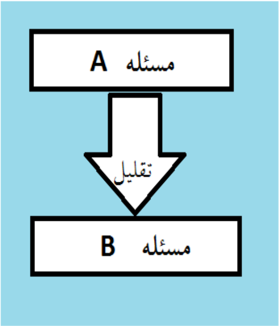
یک راه این است که مانند اثبات تصمیم ناپذیری $A_{\rm TM}$ مستقیما از روش قطری استفاده کنیم.

یک راه دیگر استفاده از روش تقلیل است. اگر بتوانیم نشان دهیم با فرض تصمیم پذیر بودن B، مسئله $A_{\rm TM}$ تصمیم پذیر خواهد شد (تناقض)، آنگاه ثابت کرده‌ایم B نیز تصمیم پذیر نیست. یعنی از یک الگوریتم برای B یک الگوریتم برای $A_{\rm TM}$ استخراج می‌کنیم.

Reduction Method

تعریف: مسئله A را تقلیل‌ـپذیر به مسئله B گویند اگر با یک الگوریتم برای B بتوان مسئله A را حل کرد. اینطور نشان داده‌ایم که مسئله A از مسئله B سخت‌تر نیست. به عبارت دیگر، مسئله B حداقل به سختی مسئله A است.





اگر با یک الگوریتم برای B

بتوان مسئله A را حل کرد

تقلیل در علم، فلسفه، فرهنگ و سیاست

- تقلیل جهانشناسی به فیزیک (علوم طبیعی)
- تقلیل فلسفه به منطق و زبانشناسی
- تقلیل هوش به توانایی حل مسئله
- تقلیل روانشناسی به روانپزشکی و بیوشیمی
- تقلیل مشکلات کلان کشور به تحریمهای اقتصادی!
- تقلیل دین به شریعت و مناسک مذهبی
- ...

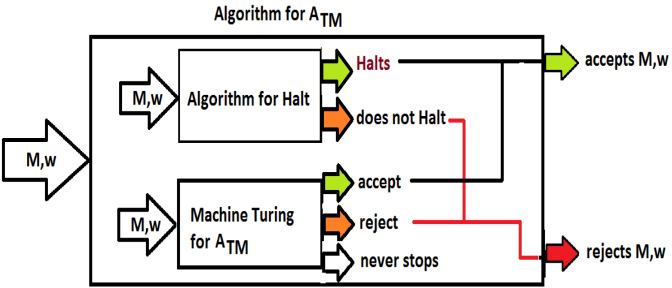
مسئله توقف ماشین تورینگ

$Halt =$

$\{\langle M, w\rangle \mid$ M یک ماشین تورینگ است که روی رشته w متوقف می‌شود $\}$

قضیه: مسئله $Halt$ تصمیم ناپذیر است.

اثبات: از روش تقلیل استفاده می‌کنیم. نشان می‌دهیم اگر الگوریتمی برای $Halt$ وجود داشته باشد، آنگاه مسئله A_{TM} را می‌توان حل کرد. (مسئله A_{TM} را به مسئله $Halt$ تقلیل می‌دهیم.)



فرض کنید الگوریتم T مسئله $Halt$ را حل می‌کند. برای حل A_{TM} دو کار را همزمان انجام می‌دهیم. دقت کنید، در مسئله A_{TM} می‌خواهیم بدانیم ماشین M رشته w را می‌پذیرد یا نه.

- اجرای M روی رشته w شبیه‌سازی می‌کنیم. اگر M متوقف شد، می‌دانیم که رشته w را پذیرفته است یا نه.
- همزمان از الگوریتم T هم استفاده می‌کنیم و چک می‌کنیم آیا ماشین M روی رشته w متوقف می‌شود یا نه. اگر جواب این بود که متوقف نمی‌شود می‌دانیم که ماشین M رشته w را نمی‌پذیرد. پس اجرای بند بالا (شبیه‌سازی) را قطع می‌کنیم.

$$E_{\mathsf{TM}} = \{\langle M\rangle |\ M \text{ is a } \mathsf{TM} \text{ and } L(M) = \emptyset\}.$$

قضیه: E_{TM} تصمیم ناپذیر است.

اثبات: با روش تقلیل.