Teoria da Computabilidade

Subseção 3

Redutibilidade

Seção 5.1 do livro
Introdução à Teoria da Computação
Michael Sipser

Redutibilidade

- Redução:
  - Esquema de conversão de um problema em algum outro, de modo que a solução deste segundo problema possa ser utilizada para resolver o primeiro.
- Se A é redutível a B, resolver A não pode ser mais difícil do que resolver B.
  - Uma solução para B fornece uma solução para A.

~~se~~ A é redutível a B
dificuldade (A) <= dificuldade (B)

- Se A é redutível a B e B é decidível, então A é decidível.
- Se A é redutível a B e A não é decidível, então B não é decidível.

Chave para provar que certos problemas são indecidíveis
(reduzindo um problema conhecidamente indecidível a ele)

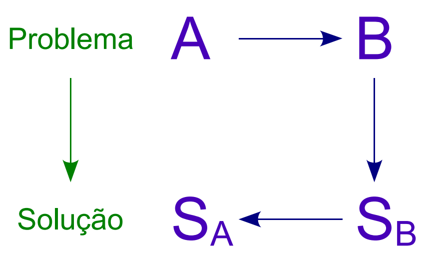
A: problema que eu JÁ SEI que é indecidível

B: problema que eu quero provar que é indecidível

Então basta mostrar que A é redutível a B

Redutibilidade

Esquema da Redução



podemos usar uma solução para B para resolver A

Como provar que um problema não é decidível?

- Mostrar que algum problema não decidível é redutível ao problema em questão.

Se A é redutível a B, então eu consigo escrever uma MT S que decida A usando uma MT R que decida B (se tal máquina R existir)
A escrita de tal máquina S (usando R) é a redução que precisa ser mostrada.

Redução de A para B para prova de indecidibilidade

A: problema indecidível
B: problema que eu quero provar que é indecidível

Prova por contradição:
- Supor que B é decidível e que a mT R decide B
- Construir uma mT S que decide A usando a mt R
- Como A é indecidível, chegamos a uma contradição, então B é indecidível

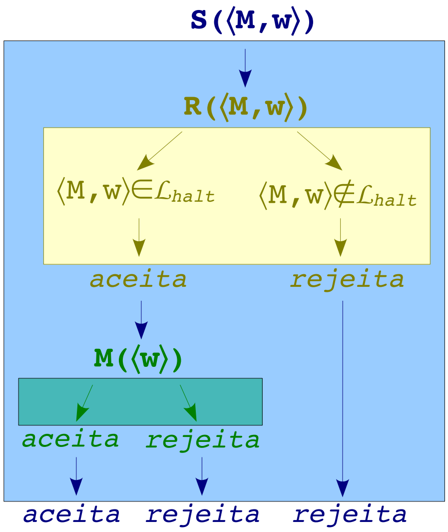
A escrita da mT S (usando R) é a redução

Problema da Parada

$\mathcal{L}_{halt} = \{\langle M, w \rangle \mid$ Máquina de Turing M pára com entrada $w\}$.

Teorema 3.18

A linguagem $\mathcal{L}_{halt}$ não é decidível.



Problema da Parada

Esquema da Prova

- Usar o problema da aceitação por MT's para provar a indecidibilidade de $\mathcal{L}_{halt}$.
- Usar a indecidibilidade de $\mathcal{L}_{MT}$ para provar a indecidibilidade de $\mathcal{L}_{halt}$.
  - $\mathcal{L}_{MT} = \{\langle M, w\rangle \mid$ Máquina de Turing M aceita a cadeia $w\}$.
- Fazer a redução de $\mathcal{L}_{MT}$ para $\mathcal{L}_{halt}$.

Problema da Parada

Esquema da Prova

- Supor que $\mathcal{L}_{halt}$ é decidível e mostrar que a linguagem $\mathcal{L}_{MT}$ é decidível (contradição).
- Supor que máquina de Turing R decide $\mathcal{L}_{halt}$.
- Máquina S usa R como subrotina para decidir $\mathcal{L}_{MT}$:
  - R recebe a entrada $\langle M, w \rangle$ e simula M.
  - Se R indica que M não pára com w, S *rejeita* pois $\langle M, w \rangle \notin \mathcal{L}_{MT}$.
  - Se R indica que M pára com w, simulação pode continuar.
- Se R existe, $\mathcal{L}_{MT}$ é decidível (contradição).

Problema da Parada

$\mathcal{L}_{halt} = \{\langle M, w\rangle \mid$ Máquina de Turing M pára com entrada $w\}$.

Teorema 3.18

A linguagem $\mathcal{L}_{halt}$ não é decidível.

Demonstração.

- Supor que máquina de Turing R decide $\mathcal{L}_{halt}$.
- Máquina de Turing S que decide $\mathcal{L}_{MT}$:
  1. S chama a máquina R com a codificação $\langle M, w\rangle$.
  2. Se R rejeita, S *rejeita*.
  3. Se R aceita, simular M com w até M parar.
  4. Se M aceitou, S *aceita*. Se M rejeitou, S *rejeita*.
- Se R decide $\mathcal{L}_{halt}$, então S decide $\mathcal{L}_{MT}$.
- Como $\mathcal{L}_{MT}$ não é decidível, $\mathcal{L}_{halt}$ também não pode ser decidível.

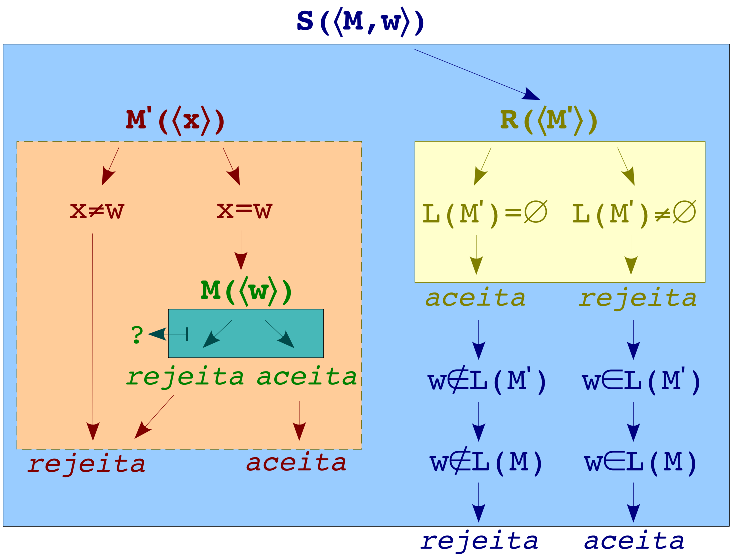
□

Problemas indecidíveis

$\mathcal{L}_{MT_\emptyset} = \{\langle M\rangle \mid M$ é uma MT e $L(M) = \emptyset\}$.

Teorema 3.19

A linguagem $\mathcal{L}_{MT_\emptyset}$ não é decidível.



Problemas indecidíveis

Esquema da Prova:

- Supor que $\mathcal{L}_{MT_\emptyset}$ é decidível e mostrar que a linguagem $\mathcal{L}_{MT}$ é decidível (contradição).
- Supor que máquina de Turing R decide $\mathcal{L}_{MT_\emptyset}$.
- Máquina S usa R como subrotina para decidir $\mathcal{L}_{MT}$:
  1. S recebe a entrada $\langle M, w\rangle$.
  2. R recebe a entrada $\langle M\rangle$.
  3. Se R aceita, $L(M) = \emptyset$ e M não aceita w.
  4. Se R rejeita, $L(M) \neq \emptyset$ e M aceita alguma cadeia.
- M aceita a cadeia w??? Resposta indefinida!

Problemas indecidíveis

Esquema da Prova:

- Supor que $\mathcal{L}_{MT_\emptyset}$ é decidível e mostrar que a linguagem $\mathcal{L}_{MT}$ é decidível (contradição).
- Supor que máquina de Turing R decide $\mathcal{L}_{MT_\emptyset}$.
- Máquina S usa R como subrotina para decidir $\mathcal{L}_{MT}$:
  1. S recebe a entrada $\langle M, w\rangle$.
  2. R recebe a entrada $\langle M'\rangle$ (M' é uma versão modificada de M que rejeita todas as cadeias, exceto w).
  3. Usar R para testar se M' reconhece $\emptyset$ ($L(M') \neq \emptyset$ se e somente se M' aceita w).

M' funciona normalmente sobre a entrada w

Problemas indecidíveis

Esquema da Prova:

- Funcionamento de M' com a cadeia x:
  1. Se $x \neq w$, M' *rejeita*.
  2. Se $x = w$, chamar M com cadeia w. Se M aceita, M' *aceita*.

Problemas indecidíveis

$\mathcal{L}_{MT_\emptyset} = \{\langle M \rangle \mid M$ é uma MT e $L(M) = \emptyset\}$.

Teorema 3.19

A linguagem $\mathcal{L}_{MT_\emptyset}$ não é decidível.

Demonstração.

- Supor que máquina de Turing R decide $\mathcal{L}_{MT_\emptyset}$.
- Máquina S usa R como subrotina para decidir $\mathcal{L}_{MT}$.
- Funcionamento de S com a entrada $\langle M, w \rangle$:
  1. Usar a descrição de M e w para construir a máquina M'.
  2. Chamar R com entrada $\langle M' \rangle$.
  3. Se R aceita, S *rejeita*. Se R rejeita, S *aceita*.
- Se R decidisse $\mathcal{L}_{MT_\emptyset}$, S decidiria $\mathcal{L}_{MT}$.
- Como $\mathcal{L}_{MT}$ não é decidível, $\mathcal{L}_{MT_\emptyset}$ também não é decidível.

□