ACH2043
INTRODUÇÃO À TEORIA DA COMPUTAÇÃO

Aula 19

Final do Cap 4.2 – O problema da parada
Cap 5 - Redutilibidade
Cap 5.1 – Problemas indecidíveis (parte 1)

Profa. Ariane Machado Lima
ariane.machado@usp.br

Na aula passada...

$$A_{\mathsf{MT}} = \{\langle M, w\rangle \mid M \text{ é uma MT e } M \text{ aceita } w\}.$$

TEOREMA 4.11

A_{MT} é indecidível.

Na aula de hoje

- Uma relação entre a decidibilidade de linguagens e seus complementos
- Como provar que outros problemas são indecidíveis...
- ... usando a técnica de redutibilidade

Voltando às linguagens Turing-NÃO-Reconhecíveis

- Vimos que existem linguagens que NÃO são Turing-reconhecíveis
- Perguntas:
  - Quais são alguns exemplos delas?
  - O que isso tem a ver com linguagens Turing-decidíveis?

Decidibilidade e reconhecibilidade

- Uma linguagem é Turing-decidível se ela E seu complemento forem ambas Turing-reconhecíveis
- Uma linguagem é **co-Turing-reconhecível** se ela for o complemento de uma linguagem Turing-reconhecível (def. do livro)
- Uma linguagem é **co-Turing-reconhecível** se seu complemento for Turing-reconhecível (acho mais preciso)

Decidibilidade e reconhecibilidade

TEOREMA 4.22 ....

Uma linguagem é decidível sse ela é Turing-reconhecível e co-Turing-reconhecível.

Em outras palavras, uma linguagem é decidível exatamente quando ela e seu complemento são ambas Turing-reconhecíveis.

Prova

TEOREMA 4.22

Uma linguagem é decidível sse ela é Turing-reconhecível e co-Turing-reconhecível.

- (=>) Se uma linguagem A é Turing-decidível, então ela é também Turing-reconhecível (pelas suas próprias definições).

Além disso, como existe uma MT_A que, para qualquer cadeia w, decide se w pertence ou não a A, pode-se construir uma MT_{AC} que reconhece o complemento de A, usando MT_A ... (fazendo o quê?) .

Prova

TEOREMA 4.22 ....

Uma linguagem é decidível sse ela é Turing-reconhecível e co-Turing-reconhecível.

- (=>) Se uma linguagem A é Turing-decidível, então ela é também Turing-reconhecível (pelas suas próprias definições).

Além disso, como existe uma MT_A que, para qualquer cadeia w, decide se w pertence ou não a A, pode-se construir uma MT_{AC} que reconhece o complemento de A, usando MT_A ... (fazendo o quê?) aceitando tudo o que MT_A rejeita e rejeitando o que MT_A aceita.

Prova

TEOREMA 4.22 ....

Uma linguagem é decidível sse ela é Turing-reconhecível e co-Turing-reconhecível.

(<=) Se uma linguagem A é Turing-reconhecível, consigo aceitar todas as cadeias de A (usando uma máquina M_1). O problema são as cadeias do complemento de A... Mas se o complemento também é reconhecível (por uma MT_2), posso rejeitar tudo o que a MT_2 do complemento aceita...

Prova

M = "Sobre a entrada w:

1. Rode ambas, M_1 e M_2, sobre a entrada w em paralelo.
2. Se M_1 aceita, *aceite*; se M_2 aceita, *rejeite*."

- Onde rodar em paralelo significa rodar cada MT em uma fita diferente, rodando um passo de cada uma de cada vez e alternadamente, até que uma delas aceite

Retomando...

$$A_{\text{MT}} = \{\langle M, w\rangle \mid M \text{ é uma MT e } M \text{ aceita } w\}.$$

TEOREMA 4.11

A_{MT} é indecidível.

- Uma linguagem é Turing-decidível se ela E seu complemento forem ambas Turing-reconhecíveis

- O complemento de A_{MT} é o quê?

COROLÁRIO 4.23

$\overline{A_{\mathsf{MT}}}$ não é Turing-reconhecível.

COROLÁRIO 4.23

$\overline{A_{MT}}$ não é Turing-reconhecível.

PROVA Sabemos que A_{MT} é Turing-reconheível. Se $\overline{A_{MT}}$ também fosse Turing-reconhecível, A_{MT} seria decidível. O Teorema 4.11 nos diz que A_{MT} não é decidível, portanto $\overline{A_{MT}}$ não pode ser Turing-reconhecível.

Na aula de hoje

- Uma relação entre a decidibilidade de linguagens e seus complementos
    - Uma linguagem é Turing-decidível se ela E seu complemento forem ambas Turing-reconhecíveis
- Como provar que outros problemas são indecidíveis...
- ... usando a técnica de redutibilidade

Redutibilidade

- Redução: conversão de um problema A em outro problema B de forma que a solução de B seja usada para solucionar A
- Ex:
  - Se você tem amigos morando em Paris, viajar para Paris pode ser reduzido a
  - Comprar uma passagem aérea de São Paulo a Paris, que pode ser reduzido a
  - Ganhar dinheiro para a passagem, que pode ser reduzido a
  - Encontrar um emprego

Redutibilidade

- Exemplos matemáticos:
  - Medir a área de um retângulo pode ser reduzido a medir a altura e a largura do retângulo
  - Resolver um problema de equações lineares pode ser reduzido ao problema de inverter uma matriz

Redutibilidade

- Utilidade:
  - Se A é redutível a B
    - A não pode ser mais (fácil/difícil?) do que B

Redutibilidade

Não pode ser mais fácil porque não faria sentido reduzir um problema mais fácil para um mais difícil.
Não pode ser mais difícil porque daí a solução de B não garantiria a solução de A.

- Utilidade:
  - Se A é redutível a B
    - A não pode ser mais fácil nem mais difícil do que B
    - Se B for decidível, A também será
    - Se A for indecidível, B também será

Redutibilidade

Não pode ser mais fácil porque não faria sentido reduzir um problema mais fácil para um mais difícil.
Não pode ser mais difícil porque daí a solução de B não garantiria a solução de A.

- Utilidade:
  - Se A é redutível a B
    - A não pode ser mais fácil nem mais difícil do que B
    - Se B for decidível, A também será
    - Se A for indecidível, B também será

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

Ex: Problema da Parada

- $PARA_{MT}$ = {<M,w> : M é uma MT e M pára sobre a entrada w}
- Que problema indecidível pode ser reduzido a esse?

Ex: Problema da Parada

- $PARA_{MT}$ = {<M,w> : M é uma MT e M pára sobre a entrada w}
- A_{MT} = {<M,w> : M é uma MT e M aceita w}
- A_{MT} (que é indecidível) pode ser reduzido a $PARA_{MT}$, pois se eu tiver a solução de $PARA_{MT}$ terei a solução de A_{MT}. Ou seja, se existir uma MT que decida $PARA_{MT}$, então essa máquina poderia ser usada para decidir A_{MT}.
- Mas A_{MT} é indecidível! Logo, $PARA_{MT}$ é indecidível

Ex: Problema da Parada

- Prova (tem que mostrar a redução!): assuma, por contradição, que uma MT R decida $PARA_{MT}$. Então construímos S que usa R para decidir A_{MT} :

Ex: Problema da Parada

- Prova (tem que mostrar a redução!): assuma, por contradição, que uma MT R decida $PARA_{MT}$. Então construímos S que usa R para decidir A_{MT} :

S = “Sobre a entrada <M, w>, uma codificação de uma MT M e uma cadeia w:

1. Rode a MT R sobre a entrada <M, w>.
2. Se R rejeita,   .
3. Se R aceita,

Ex: Problema da Parada

- Prova (tem que mostrar a redução!): assuma, por contradição, que uma MT R decida $PARA_{MT}$. Então construímos S que usa R para decidir A_{MT} :

S = “Sobre a entrada <M, w>, uma codificação de uma MT M e uma cadeia w:

1. Rode a MT R sobre a entrada <M, w>.
2. Se R rejeita, rejeite.
3. Se R aceita,

Ex: Problema da Parada

- Prova (tem que mostrar a redução!): assuma, por contradição, que uma MT R decida $PARA_{MT}$. Então construímos S que usa R para decidir A_{MT} :

S = “Sobre a entrada <M, w>, uma codificação de uma MT M e uma cadeia w:

1. Rode a MT R sobre a entrada <M, w>.

2. Se R rejeita, rejeite.

3. Se R aceita, simule M sobre w até ela pare.

2. Se M aceitou, aceite; se M rejeitou, rejeite.”

Ex: Problema da Parada

- Prova (tem que mostrar a redução!): assuma, por contradição, que uma MT R decida $PARA_{MT}$. Então construímos S que usa R para decidir A_{MT} .

- Logo A_{MT} pode ser reduzido a $PARA_{MT}$
- Como A_{MT} é indecidível, $PARA_{MT}$ é indecidível