

MÁQUINA DE TURING E AUTÔMATO DE PILHA

Delryson da Silva Saraiva
Katarina Ires de Castro Pinheiro
Universidade Federal do Maranhão – UFMA – São Luís – MA
{delryson.ss, katarina.ires}@discente.ufma.br

Resumo

Este artigo apresenta uma comparação entre a Máquina de Turing e o Autômato de Pilha, dois modelos fundamentais da Teoria da Computação. São discutidas suas características, diferenças e aplicações, bem como suas limitações no reconhecimento de linguagens formais. O Autômato de Pilha se destaca no reconhecimento de linguagens livres de contexto e na análise sintática, oferecendo um modelo eficiente, porém limitado em termos de capacidade computacional. Por outro lado, a Máquina de Turing apresenta uma expressividade muito maior, permitindo a simulação de qualquer algoritmo computável e estabelecendo os limites da computabilidade teórica.

Palavras-chave: máquinas de Turing, autômato de pilha, linguagens formais.

1. Introdução

A Teoria da Computação busca compreender os limites e capacidades dos modelos formais de computação (Hopcroft et al., 2006). Entre esses modelos, destacam-se a Máquina de Turing e o Autômato de Pilha, essenciais para o estudo da computabilidade e da análise de linguagens formais (Sipser, 2012).

A Máquina de Turing, proposta por Alan Turing em 1936, é um dos pilares da computação moderna e define o conceito de algoritmo computacionalmente viável (Turing, 1936). Este artigo apresenta uma comparação entre esses dois modelos, destacando suas características, diferenças e aplicações.

2. Definição Formal

2.1 Autômato de Pilha

Um Autômato de Pilha (AP) é formalmente definido como uma 7-tupla:

$$M=(Q,\Sigma,\Gamma,\delta,q_0,Z,F), \quad (1)$$

onde:

- Q é um conjunto finito de estados;
- Σ é o alfabeto de entrada;
- Γ é o alfabeto da pilha;
- $\delta: Q \times (\Sigma \cup \{\epsilon\}) \times \Gamma \rightarrow P(Q \times \Gamma^*)$ é a função de transição;
- $q_0 \in Q$ é o estado inicial;
- $Z \in \Gamma$ é o símbolo inicial da pilha;
- $F \subseteq Q$ é o conjunto de estados finais.

A computação de um AP ocorre ao ler símbolos da entrada e modificar a pilha de acordo com a função de transição δ . Um AP pode aceitar uma palavra pelo estado final ou pelo esvaziamento da pilha.

2.2 Máquina de Turing

Uma Máquina de Turing (MT) é formalmente definida como uma 7-tupla:

$$M=(Q, \Sigma, \Gamma, \delta, q_0, q_{\text{accept}}, q_{\text{reject}}) \quad (2)$$

na qual:

- Q : Conjunto finito de estados.
- Σ : Alfabeto de entrada. ($\Sigma=\{a,b,c\}$ para este exemplo).
- Γ : Alfabeto da fita ($\Gamma=\{a,b,c,X,Y,\#\}$, onde X e Y são marcadores).
- δ : Função de transição $\delta:Q \times \Gamma \rightarrow Q \times \Gamma \times \{L,R,S\}$, que define mudanças de estado, substituições e movimentos.
- q_0 : Estado inicial.
- q_{accept} : Estado de aceitação.
- q_{reject} : Estado de rejeição.

A computação de uma MT ocorre ao ler símbolos da fita, escrever novos símbolos e mover a cabeça de leitura/escrita para a esquerda (L), direita (R) ou permanecer no mesmo lugar (S), conforme definido por δ . A aceitação de uma palavra ocorre quando a MT entra no estado q_{accept} .

3. Comparação entre os Modelos Computacionais

3.1 Autômato de Pilha

O Autômato de Pilha (AP) é um modelo computacional que estende o Autômato Finito ao incluir uma estrutura de dados chamada pilha. Essa estrutura permite que o AP tenha uma memória limitada baseada no princípio LIFO (Last In, First Out). Os APs são capazes de reconhecer Linguagens Livres de Contexto (LLC), como as usadas na análise sintática de linguagens de programação. Suas principais características são:

- Possui um conjunto finito de estados.
- Usa uma pilha para armazenar símbolos temporariamente.
- As transições dependem do estado atual, da entrada lida e do topo da pilha.
- Pode ser determinístico (DAP) ou não determinístico (NAP), sendo o NAP mais poderoso para reconhecimento de LLCs.

Apesar de sua utilidade, os APs não conseguem reconhecer linguagens mais complexas, como as Sensíveis ao Contexto, pois sua pilha não é suficiente para armazenar informações mais detalhadas sobre a estrutura da linguagem.

3.2 Máquina de Turing

A Máquina de Turing (MT), proposta por Alan Turing, é um modelo computacional mais poderoso que o AP, sendo capaz de simular qualquer algoritmo computacionalmente viável. Diferente do AP, que usa uma pilha, a MT possui uma fita infinita que pode ser lida e escrita repetidamente. Isso permite que ela reconheça Linguagens Recursivamente Enumeráveis e resolva problemas que exigem memória ilimitada.

Suas principais características são:

- Possui uma fita infinita dividida em células.
- Utiliza uma cabeça de leitura/escrita que se move pela fita.
- As transições são definidas por uma função de transição que altera o estado, escreve na fita e move a cabeça de leitura/escrita.
- É um modelo fundamental para a definição de computabilidade.

A MT é mais poderosa que o AP, pois pode simular qualquer autômato de pilha e resolver problemas que exigem mais memória e controle sobre os dados armazenados.

2.5 Tabela resumo entre Máquina de Turing e Autômato de Pilha

Característica	Autômato de Pilha (AP)	Máquina de Turing (MT)
Memória	Pilha (LIFO)	Fita infinita
Poder computacional	Linguagens Livres de Contexto	Linguagens Recursivamente Enumeráveis
Possibilidade de retrocesso	Não pode acessar elementos antigos diretamente	Pode mover a cabeça de leitura para frente e para trás
Determinismo	Pode ser determinístico ou não	Geralmente determinística, mas pode ser não determinística
Aplicações	Compiladores, análise sintática	Modelagem da computação, decidibilidade de problemas

Tabela 01 - Resumo comparativo entre MT e AP

4. Conclusão

A Máquina de Turing e o Autômato de Pilha representam dois marcos fundamentais na teoria da computação, cada um com sua importância e aplicações específicas. A distinção entre esses dois modelos reflete diretamente na forma como são utilizados em diversas áreas da computação. Enquanto o Autômato de Pilha é essencial no desenvolvimento de compiladores e interpretação de linguagens formais, a Máquina de Turing fornece a base para o estudo da decidibilidade e complexidade computacional. Ao examinar suas características e compará-las, é possível compreender melhor as restrições impostas por cada modelo e a relação entre computação prática e teórica.

Ao considerar a aplicabilidade desses modelos, percebe-se que a expressividade computacional da Máquina de Turing justifica sua adoção em problemas mais sofisticados, enquanto o Autômato de Pilha continua relevante para cenários onde a análise sintática eficiente é primordial. Dessa forma, a compreensão desses conceitos é essencial para profissionais e pesquisadores da computação, auxiliando na construção de sistemas computacionais mais robustos e fundamentados nos princípios formais da computação.

5. Referências

- HOPCROFT, J. E., ULLMAN, J. D. *Introduction to Automata Theory, Languages, and Computation*. Pearson, 2007.
- SIPSER, M. *Introduction to the Theory of Computation*. Cengage Learning, 2012.
- TURING, A. M. (1936). On Computable Numbers, with an Application to the Entscheidungsproblem. Proceedings of the London Mathematical Society.