UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

BACHARELADO EM ENGENHARIA DE COMPUTAÇÃO

LUCAS SANTOS SOUZA

AUTôMATO DETERMINíSTICO E não-DETERMINíSTICO

APUCARANA, 2021

1. INTRODUÇÃO

A teoria dos autômatos é definida como sendo um modelo matemático, desempenhando um papel importante em diversas áreas aplicadas da ciência da computação. Um autômato, ou máquina abstrata, utiliza o conceito de estado em combinação com instruções primitivas, no qual recebem especificações de como cada instrução modifica cada estado.

Um modelo, chamado autômato finito, é bastante utilizado em processamento de texto para reconhecer padrões, compiladores, projeto de hardwares e etc. Contudo possui aplicações práticas restritas, visto que a informação de saída é limitada à lógica binária aceita/rejeita. Em uma definição formal é representado por uma 5-upla (*Q, Σ, δ, q0, F*), onde:

1. *Q* é um conjunto finito conhecido como os estados, representado por círculos;
2. *Σ* é um conjunto finito de símbolos de entrada, chamado o alfabeto;
3. *δ :* é a função de transição, representado por arestas, é a responsável pela transição entre os estados a partir de um determinado símbolo do alfabeto;
4. *q0 ∈ Q* é o estado inicial;
5. *F* é um subconjunto de *Q*, denominado conjunto de estados finais.

Este modelo pode ser dividido em dois tipos:

* Autômato finito determinístico (AFD): Para um dado estado atual e o símbolo lido como entrada, o sistema assume um único estado, ou seja, o autômato pode pular para um e somente um estado.
* Autômato finito não determinístico (AFND): Para um dado estado atual e o símbolo lido como entrada, o sistema pode assumir mais de um único estado possível.

1. AUTÔMATOS PROPOSTOS
   1. AFND

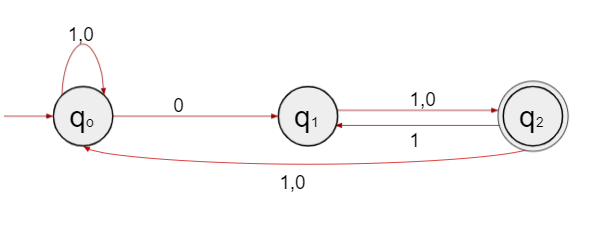


Figura : AFN PROPOSTO

Seguindo a definição formal, o AFND é uma 5-upla (*Q, Σ, δ, q0, F*), onde

1. Q = {q0, q1, q2}
2. Σ = {0, 1}
3. δ(q0,0) = q0, q1

δ(q0,1) = q0

δ(q1,0) = q2

δ(q1,1) = q2

δ(q2,0) = q0

δ(q2,1) = q0, q1

1. q0 = q0
2. F = {q2}

O AFND possui como estado inicial o q0, onde a partir dele podemos receber qualquer símbolo do nosso alfabeto, no caso do primeiro símbolo for “1” permanecemos no mesmo estado q0, no caso do símbolo for “0” podemos permanecer no mesmo estado ou seguir para o próximo, q1. Em q1 qualquer símbolo de entrada presente no alfabeto nos leva para o próximo estado, q2. No estado final, q2, também não temos nenhuma restrição de entrada, no caso da entrada “0” retornamos para o estado inicial q0, no caso da entrada “1” podemos retornar para q0 ou retornar para o estado q1.

* 1. AFD

Apesar de possuir definições diferentes, a partir de um AFND pode-se construir um AFD equivalente e para isso podemos usar a tabela de transição da seguinte forma:

|  |  |  |  |
| --- | --- | --- | --- |
| Passos | estados | 0 | 1 |
| 1 | q0 | {q0, q1} | q0 |
| 2 | {q0, q1} | {q0, q1, q2} | {q0, q2} |
| 3 | {q0, q1, q2} | {q0, q1, q2} | {q0, q1, q2} |
| 4 | {q0, q2} | {q0, q1} | {q0, q1} |

Tabela : Tabela de Transição

No primeiro passo temos o estado inicial q0 e suas transições, como o símbolo de entrada “0” pode levar a dois estados diferentes ocorre uma junção desses estados, onde esse conjunto se torna um novo estado {q0, q1}. No segundo passo a partir do novo conjunto obtido analisamos as transições de cada um de seus elementos para criar outro conjunto de estados. O mesmo ocorre para os passos 3 e 4. O estado final do AFD se dá pelo conjunto que contém o estado final do AFND, podendo ter mais de um único estado de aceitação, nesse caso, como o AFND possui o estado final q2, o AFD terá como estado final {q0, q1, q2} e {q0, q2}. A partir desta tabela é possível criar o seguinte autômato:

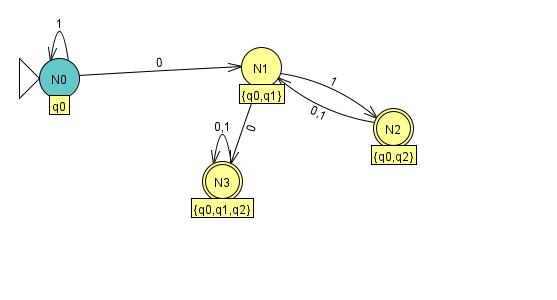


Figura : AFD

Seguindo a definição formal, o AFD é uma 5-upla (*Q, Σ, δ, q0, F*), onde

1. Q = {N0, N1, N2, N3}
2. Σ = {0, 1}
3. δ(N0,0) = N1

δ(N0,1) = N0

δ(N1,0) = N3

δ(N1,1) = N2

δ(N2,0) = N1

δ(N2,1) = N1

δ(N3,0) = N3

δ(N3,1) = N3

1. q0 = N0
2. F = {N2, N3}

O AFD possui como estado inicial N0, onde temos um loop no caso da entrada “1” que nos mantém no mesmo estado, no caso da entrada “0” passamos para o próximo estado N1. Em N1 podemos seguir para qualquer um dos estados finais, no caso da entrada “1” seguimos para o estado final N2, no caso da entrada “0”, seguimos para o estado final N3. No estado N2, temos somente um único caminho, então para qualquer entrada que esteja presente no alfabeto retornamos para o estado N1. O estado final N3 também possui somente um caminho, um loop onde independente da entrada, desde que esteja presente no alfabeto, continuamos sempre no mesmo estado.

1. CONCLUSÃO

A partir do modelo AFD podemos definir a linguagem aceita pelo autômato. Sua expressão regular pode ser descrita pela união (0 U 1). Tendo em vista que para ter a aceitação devemos ter no mínimo o conjunto de entrada {0,1} ou {0,0}, e também não podemos ter uma cadeia com a sequência “010”, ou seja, não podemos ter uma sequencia intercalando “0” e “1” tal que a quantidade de 0’s seja um numero par e a quantidade de 1’s seja ímpar. Como por exemplo “0101010”, ou “0101010”.