



***Desenvolvimento de um tradutor de
máquinas de Post para máquinas de Turing
Teoria da Computação***

Jiovana Sousa Gomes(gomesjiovana@gmail.com)
Matheus Henrique Koch (matheuskoch13@gmail.com)

Engenharia de Computação
Bagé, 2019

Máquina de Post

1) Definição formal:

Uma máquina de Post é uma tupla $M = (K, \Sigma, P, s, F, D, \#)$ onde:

- K é um conjunto finito de estados;
- Σ é o alfabeto de símbolos da entrada;
- P é o alfabeto da fila;
- $s \in K$ é a instrução inicial;
- $\# \in P$ é o símbolo auxiliar que marca o final da fila;
- $F \subseteq K$ é o conjunto de estados de parada;
- D é a função de transição $K \times P \Rightarrow K \times P \times \{d, i\}$, onde d representa um desvio e i uma atribuição.

2) Equivalência entre máquinas de Post e Turing

Máquina de Post (MP) \leq Máquina de Turing (MT):

- A entrada da máquina de Post é simulada na máquina de Turing usando a estrutura de fita, onde a primeira posição da fila corresponde à posição corrente da cabeça da fita;
- O desvio realiza a leitura com remoção do símbolo mais à esquerda da entrada, o que pode ser simulado pela substituição do conteúdo da correspondente célula de fita pelo símbolo branco, resultando em símbolos brancos entre o marcador de início da fita e o primeiro símbolo não branco.;
- A atribuição de um símbolo s (à direita do conteúdo atual da fila), pode ser simulada movendo a cabeça para a direita até o primeiro símbolo branco, substituindo-o pelo símbolo s e, retornando para posição correspondente ao primeiro símbolo da fila;
- Os estados inicial e de parada podem ser interpretados da mesma forma tanto na máquina de Post quanto de Turing, pois o estado inicial é sempre a primeira função de transição e os estados de parada são aqueles que não tem funções com eles sendo a origem da transição.

3) Linguagem para descrição de máquinas de Post

Utilizando a definição da linguagem MT fornecida para realização do trabalho, para definir a linguagem de uma máquina de Post, o alfabeto dá-se por: $\Sigma = \{ (,), q, a, 0, 1, , d, i \}$. Os 2 últimos símbolos representam as operações elementares realizadas pela máquina de Post, d (desvio) e i (atribuição). As demais operações de partida e parada serão explicadas no próximo parágrafo. Os símbolos q , a , 0 e 1 são

usados para codificação dos estados e símbolos da linguagem a ser usada na máquina de Post, de forma que se o alfabeto tem n símbolos, então todos símbolos podem ser codificados com $\lceil \log_2 n \rceil$. Assim, representa-se os estados com um q seguido de um número binário e os símbolos com um a seguido de um número binário.

A transição de uma máquina de Post ocorre com base nas operações realizadas. Pode ser descrita na forma $(qn, op(i,d),x,qm)$ onde qn é o estado atual, op representa uma atribuição (i) ou desvio (d), x é uma variável a ser escrita ou removida da fila e qm é o próximo estado. Os casos especiais de partida e parada podem ser interpretados com a partida sendo sempre o estado inicial q_0 , e os estados de parada como sendo apenas estados seguintes, sem transições de saída.

4) Exemplo de representação:

A exemplo, tem-se a seguinte MP, que aceita número par de uns: $M = (\{0,1\}, D, \#)$. A representação dos elementos do alfabeto fica na forma:

Símbolo	tradução
0	a00
1	a01
#	a10

Uma máquina de post geralmente é descrita com instruções, mas na representação com estados, foram atribuídos números crescentes em binário ao iniciar pela instrução de início sendo $q000$.

O fluxo das instruções é representado então como D:

($q000,,,q001$)
 ($q001,i,a10,q010$)
 ($q010,d,a01,q011$)
 ($q010,d,a10,q100$)
 ($q010,d,a00,q010$)
 ($q011,d,a01,q010$)
 ($q011,d,a00,q011$)
 ($q011,d,a10,q101$)

Nota-se que a primeira transição possui espaços 'vazios' no local onde estaria o símbolo 'op' e a variável 'x' descritas no tópico 3. Isso ocorre devido à transição inicial não realizar operação nenhuma, e portanto não necessita de variável também. Na tradução para a máquina de Turing, isso se tornará também a transição inicial característica de ler 'branco, escrever 'branco' e avançar para direita.

5) Gramática da máquina de Post no formato BNF:

$\langle MP \rangle ::= \langle programa \rangle$
 $\langle programa \rangle ::= \langle sequência de instruções \rangle$
 $\langle sequência de instruções \rangle ::= \langle instrução \rangle \mid \langle instrução \rangle \langle sequência de instruções \rangle$
 $\langle instrução \rangle ::= (\langle estado \rangle, \langle operação \rangle, \langle variável \rangle, \langle estado \rangle)$
 $\langle estado \rangle ::= q \langle binário \rangle$
 $\langle binário \rangle ::= \langle dígito \rangle \mid \langle dígito \rangle \langle binário \rangle$
 $\langle dígito \rangle ::= 0 \mid 1$
 $\langle operação \rangle ::= i \mid d$
 $\langle variável \rangle ::= a \langle binário \rangle$

6) Tradução formal entre máquinas de Post e de Turing com base na linguagem definida na descrição do trabalho e na equivalência entre as máquinas:

A tradução a seguir é um exemplo, com estados fictícios, e símbolos sendo: a01 = 1, a00 = branco, * = 0 ou 1 e q* = estado criado ao realizar atribuição na MT.

Início MP	Início MT
(q00,,,q01)	(q00,a00,q01,a00,d)
Desvio MP	Desvio MT
(q01,d,a01,q10)	(q01,a01,q10,a00,d)
Atribuição MP	Atribuição MT
(q01,i,a01,q10)	(q01,*,q01,*,d)
-	(q01,a00,q*,a01,e)
-	(q*,*,q*,*,e)
-	(q*,a00,q10,a00,d)

Tabela 1: Exemplo de tradução formal MP -> MT

Estados de parada não podem ser identificados explicitamente nas transições, mas podem ser vistos como estados que não são 'estados iniciais' de nenhuma transição. Isso vale tanto para a máquina de Turing como a de Post.

7) Tradução Post-Turing - Resultado esperado da saída do tradutor, para a máquina de Post do tópico 4

Arquivo 'resultado.txt':

(q000,a00,q011,a00,d)
(q011,a01,q011,a01,d)
(q011,a10,q011,a10,d)
(q011,a11,q011,a11,d)
(q011,a00,q110,a01,e)
(q110,a01,q110,a01,e)
(q110,a10,q110,a10,e)
(q110,a11,q110,a11,e)
(q110,a00,q100,a00,d)
(q100,a10,q101,a00,d)
(q100,a01,q001,a00,f)
(q100,a11,q100,a00,d)
(q101,a10,q100,a00,d)
(q101,a11,q101,a00,d)
(q101,a01,q010,a00,f)

Arquivo 'tabelaTraducao', contém traduções dos estados da MP para MT:

Tabela de tradução de estados:

q000->q000
q001->q100
q010->q101
q011->q001
q100->q010
q101->q011
q110->est6

Tabela de tradução de variáveis:

a00-> ' ' -> branco
a01->a10
a10->a01
a11->a00

8) Relatório sobre o desenvolvimento do trabalho

Inicialmente, fez-se uma pesquisa na internet a fim de encontrar a melhor definição de máquina de Post e da equivalência entre as máquinas de Post e Turing. Também buscou-se exemplos de programas que implementem algo similar ao pedido no trabalho, sendo encontrados apenas simuladores de MP, porém estes foram de ajuda para entender como representar a máquina de uma forma textual.

A seguir, consta divisão das tarefas do relatório entre os integrantes do grupo:

Jiovana - Def. Formal, Linguagem e Gramática

Matheus - Implementação do tradutor em python

Achou-se melhor para a execução do trabalho seguir o exemplo da estrutura da linguagem MT apresentada, sendo a linguagem máquina de Post construída de forma similar à linguagem MT.

O grupo reuniu-se inicialmente para começar a execução e definir o que havia de ser feito, então as tarefas foram feitas em separado e depois houve o encontro da aula do dia 07/11, onde o trabalho ficou quase completamente terminado, após isso, alguns ajustes e testes foram feitos para comprovar a execução correta do tradutor.

Como pode ser percebido no tópico 7, a tradução para máquina de Turing apresentou mais estados que na máquina de Post, em razão de ser uma máquina determinística onde devem ser consideradas transições para todos símbolos quando ocorre desvio, e principalmente devido à operação de atribuição que necessita de 4 estados a mais para sua representação.

9) Utilização do tradutor

O tradutor foi escrito em Python, necessitando ser executado em algum compilador dessa linguagem. Para efetuar a tradução, um arquivo de texto contendo a máquina de post descrita na linguagem apresentada neste trabalho deve ser informado no código, no método de leitura do arquivo. O atual tem o nome de 'teste.txt'. É importante salientar que esse arquivo deve seguir um padrão, que é o seguinte:

- Na primeira linha, é informado o estado inicial da máquina de Post.
- Na linha subsequente, são informados os estados de parada, geralmente com o estado de aceitação sendo o primeiro.
- Da terceira linha em diante, estão as regras de transição da máquina.

O arquivo contendo o código que efetua a tradução tem o nome de 'tradutorPost2', ele gera como saída dois arquivos texto, um com o nome de 'resultado' contendo a lista de transições que descrevem a máquina de Turing traduzida, e um outro arquivo auxiliar que contém tabelas dos estados e símbolos com as traduções de Post para Turing, chamado 'tabelaTraducao'.

O código encontra-se disponível em:

https://github.com/Koch1/Tradutor_Maquina_Post_para_Turing.