

MATA51 - Teoria da Computação - T02

Projeto Final - Turing Airlines



Fernando Antonio Marques Schettini - 221216929,
Heverton Silva dos Reis - 221118967

Novembro, 2023

Sumário

1	Introdução	3
2	Solução para MT1	3
2.1	Definição formal	4
2.2	Módulos	4
2.3	Complexidade	9
2.4	Exemplos	9
2.4.1	Exemplo de execução 1:	9
2.4.2	Exemplo de execução 2:	12
3	Solução para MT2	15
3.1	Definição Formal	15
3.2	Módulos	17
3.3	Complexidade	19
3.4	Exemplos	20
3.4.1	Exemplo de execução 1:	20
3.4.2	Exemplo de execução 2:	21
3.4.3	Exemplo de execução 3:	22
4	Considerações Finais	24

1 Introdução

As máquinas de Turing são uma classe de máquinas proposta por volta de 1930 pelo matemático inglês Alan Turing, as quais o modelo serve de base aos sistemas computacionais atuais [2].

O presente estudo detalha a utilização dessas máquinas para resolver um desafio específico: o cálculo de horários de partida e chegada de voos, levando em conta os fusos horários internacionais. O objetivo é determinar o horário de chegada em uma cidade destino, considerando as longitudes das cidades de origem e destino, além da duração do voo.

A complexidade deste problema é abordada através do desenvolvimento e análise de duas soluções distintas baseadas em máquinas de Turing, aqui denominadas MT1 e MT2. Para cada solução, é explorado detalhadamente sua definição formal, módulos criados, análise de complexidade e exemplos práticos de aplicação. Adicionalmente, discute-se o cálculo do número máximo de passos que cada máquina pode executar, utilizando notação assintótica para ilustração.

Um aspecto crucial deste estudo são os fusos horários, que são determinados pela diferença longitudinal em relação ao meridiano de Greenwich. O planeta é dividido em 24 zonas, cada uma com 15° de longitude. Seguimos uma regra simples para os cálculos: adicionar uma hora para cada 15° a leste de Greenwich e subtrair uma hora para cada 15° a oeste.

2 Solução para MT1

A solução para o problema principal, por meio da construção de uma máquina de Turing de fita única e determinística - a partir de então referida apenas como MT1 - se deu a partir da abstração, em termos da linguagem da fita, da seguinte expressão matemática:

$$H_B = H_A + (F_B - F_A)$$

Na equação, H_B é o *horário local na cidade de destino* quando o voo partiu, H_A é o *horário local na cidade de origem* quando o voo partiu, F_B é o *fuso horário da cidade de destino* e F_A é o *fuso horário da cidade de origem*. Esta expressão retorna o horário local na cidade de destino do voo, no momento em que o avião decola da cidade de origem. Para obter a saída final, ou seja, o horário local na cidade de destino quando o avião pousa, deve-se somar a duração do voo ao resultado da expressão anterior, resultando em:

$$S = H_B + D_{voo}$$

Na equação, D_{voo} é a *duração do voo*.

2.1 Definição formal

A MT1 projetada segue a definição formal, formada pela seguinte 7-upla, que é complementada pela tabela 1:

$$MT1 = (Q, \Sigma, \Gamma, s, b, F, \delta)$$

Tabela 1: Descrição dos Símbolos, significados de cada elemento da MT1.

Símbolo	Significado	Valor
Q	Conjunto finito de estados de MT1	Os estados estão descritos nas figuras adiante.
Σ	O alfabeto da entrada de MT1	$\Sigma = \{+, -, 0, 1\}$
Γ	É o alfabeto da fita i , onde $b \in \Gamma^i$ e $\Sigma \subseteq \Gamma^i$	$\Gamma^i = \{+, -, 0, 1\}, \forall i \in \mathbf{N}$
s	Estado inicial de MT1	$M0.0$
b	Símbolo branco tal que $b \in \Gamma$	$b = \square$
F	Conjunto dos estados finais de MT1 tal que $F \subseteq Q$	$\{F\}$
δ	Função de transição $\delta : Q \times \Gamma \rightarrow Q \times \Gamma \times \{E, D, S\}$, que descreve o comportamento de máquina de Turing em seus estados e símbolos lidos.	As funções de transições estão descritas nas figuras adiante.

A linguagem da fita é composta por um alfabeto com os seguintes caracteres básicos $\{+, -, 0, 1\}$. Os valores numéricos são representados em base unária através do caractere '1'. Cada entrada é separada por um caractere '0'. Por fim, o hemisfério de cada fuso (Leste ou Oeste de Greenwich) é identificado por meio dos caracteres '+' e '-'. O caractere '+' foi escolhido para referenciar fusos no hemisfério Leste e '-' foi escolhido para referenciar fusos no hemisfério a Oeste de Greenwich.

As entradas são organizadas da seguinte maneira: a primeira sequência de caracteres antes do '0' representa o F_B , ainda em graus; logo após o primeiro separador, tem-se a segunda sequência de caracteres, que representa o F_A , também em graus; após o segundo separador segue a duração do voo em horas (D_{voo}). Por fim, após o terceiro separador, tem-se o horário em que o voo saiu na cidade de origem, ou seja o H_A , seguido por um último separador. Assim, um exemplo de entrada para a **MT1** se apresenta:

$$-11111111111111110 - 1111111111111111011101111110$$

A saída final da fita será um valor numérico representado em base unária por uma sequência de caracteres '1' que denotam o horário em que o voo pousa na cidade de destino.

2.2 Módulos

O funcionamento da MT1 pode ser resumido na seguinte sequência de passos:

1. A fita é carregada com a entrada, no modelo previamente estabelecido.
2. O cabeçote é posicionado no início da fita.
3. A máquina traduz os fusos de graus para horas, da seguinte maneira:

- 3.1. Copia o caractere identificador de hemisfério (CIH) para o fim da fita.
- 3.2. Divide o fuso em graus por 15.
- 3.3. Insere o resultado no fim da fita, após o caractere identificador.
- 3.4. Insere um caractere separador no final da fita.
- 3.5. Repete os passos anteriores para o outro fuso.
4. A máquina subtrai o fuso da cidade de destino pelo da cidade de origem.
5. A máquina soma o resultado do passo anterior ao horário de partida na cidade de origem.
6. A máquina soma o resultado do passo anterior à duração do voo.
7. A máquina verifica se o resultado do passo anterior excede 24 horas.
 - 7.1. Se sim, subtrai o valor por 24.
8. O cabeçote é posicionado de volta ao início da fita.

Para que fosse possível executar com precisão cada passo descrito anteriormente, a MT1 foi dividida em outras máquinas menores chamadas de módulos, aqui referenciados como **M0**, **M1**, **M2**, **M3** e **M4**.

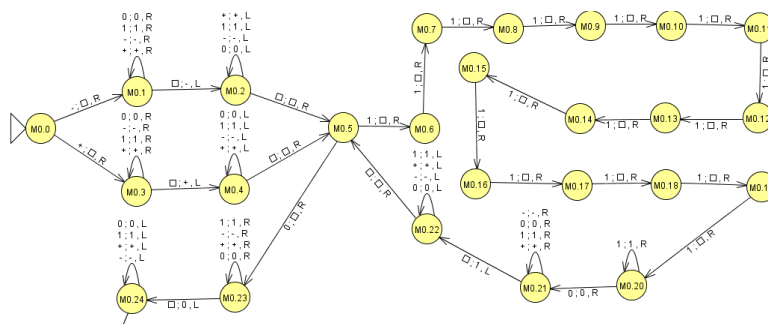


Figura 1: Diagrama de estados para a M0.

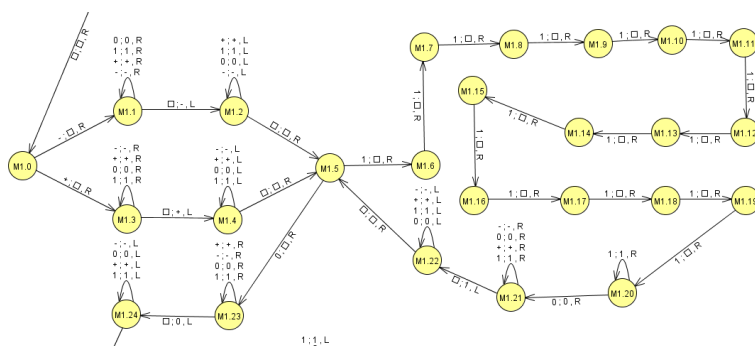


Figura 2: Diagrama de estados para a M1.

Os módulos **M0** e **M1** são os módulos iniciais, representados pela **Figura 1** e pela **Figura 2** respectivamente. Eles foram criados para atender ao passo 3 descrito nos passos de execução. Nesse sentido, com o cabeçote no início da fita, a máquina identifica o primeiro caractere como um caractere identificador de hemisfério (CIH) e o copia para o final da fita, retornando o cabeçote para o início da fita ao final deste ciclo. Em seguida, a máquina realiza a divisão do valor unário do fuso em graus por quinze (15) através de um processo de contagem, ou seja, a máquina apaga quinze (15) caracteres '1' do valor do fuso, percorre a fita até o final e insere um caractere '1' logo após o CIH. Então a máquina retorna o cabeçote ao início da fita, e repete o procedimento descrito até que seja lido o caractere separador '0'.

Ao encontrar o caractere separador '0', a máquina será conduzida ao módulo **M1**, que funciona de maneira idêntica ao módulo **M0**. Todavia, enquanto **M0** gera o valor convertido em horas do fuso na cidade de destino, F_B , **M1** gera o valor convertido em horas do fuso na cidade de origem, F_A .

Um ponto a salientar é que a máquina sempre retorna o cabeçote para o começo da fita ao final do processamento de cada máquina.

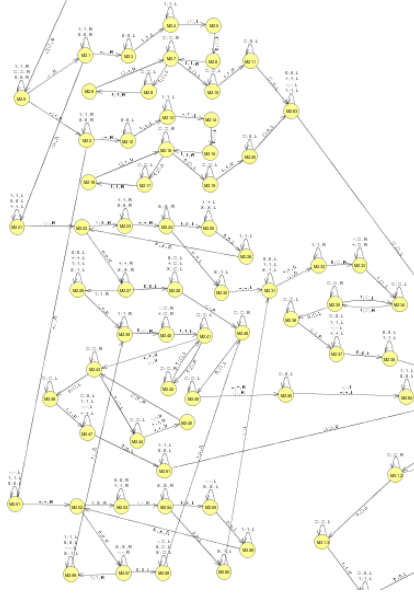


Figura 3: Diagrama de estados para a M2.

O módulo **M2** exemplificado na **Figura 3**, tem como função realizar o passo 4, onde a máquina subtrai o fuso da cidade de destino pelo da cidade de origem, ou seja, realiza a operação $(F_B - F_A)$. Um ponto de atenção desta operação, é que o caractere identificador de hemisfério CIH, (os símbolos '+' e '-') altera a maneira como este cálculo é realizado e abre margem para algumas condicionais.

A primeira operação realizada por **M2** é mudar o sinal de F_A . Sendo assim, o cabeçote, a partir do início da fita caminha para a direita até encontrar o segundo CIH (o primeiro sempre é ignorado). Ao encontrar o segundo CIH a máquina altera o caractere de acordo com a regra: se o caractere for um '+', ele será alterado para '-', caso o CIH seja um - será alterado para '+' conforme a equação que calcula o resultado descrito anteriormente.

A partir daí as seguintes condicionais são atingidas: se o primeiro CIH for um '+' (o primeiro CIH é não é modificado, e é considerado para as demais operações, esta é a primeira bifurcação a partir da leitura do primeiro CIH). A máquina avança o cabeçote até o segundo CIH e verifica se ele também é um '+', caso seja, a máquina simplesmente soma os valores unários após este caractere com os valores unários anteriores. No processo, os caracteres *separador* e *identificador* são apagados e o módulo copia os valores nos espaços vazios. A situação se repete se o primeiro CIH for um '-' e o segundo também for um '-', já que é o comportamento esperado para o caso de sinais iguais.

Quando os CIH são diferentes, é preciso utilizar uma abordagem diferente. Deve-se verificar qual dos dois valores unários é o maior, já que o CIH que o precede será preservado. Para tal, **M2** faz uso de um submódulo comparador que verifica as três possibilidades decorrentes: quando $F_B > F_A$, quando $F_B < F_A$ e quando $F_B = F_A$ [3]. Em todas as três possibilidades será realizada uma operação de subtração que consiste em, apagar os caracteres *separador* e *identificador* entre os valores unários e, em seguida, apagar um caractere 1 do maior valor, para cada caractere 1 do menor valor. O caractere CIH associado ao maior valor será copiado para a posição do primeiro CIH. Salienta-se que, no caso em que os valores são iguais, a máquina deverá zerar os valores unários. Os módulos subsequentes estão preparados para lidar com este caso.

Finalizadas as operações, a máquina retorna o cabeçote para o início da fita, como já comentado e encaminha o fluxo para o próximo módulo.

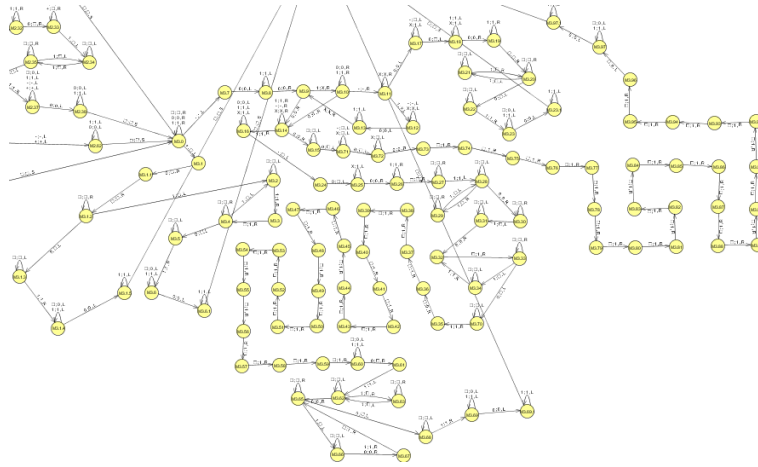
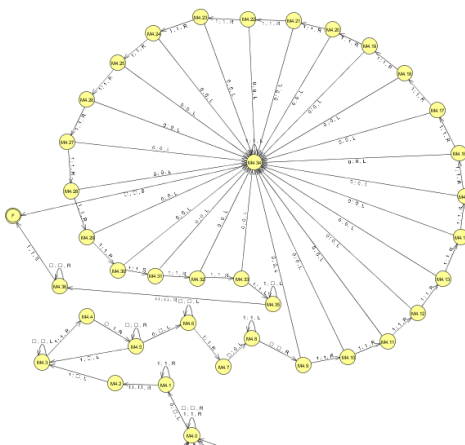


Figura 4: Diagrama de estados para a M3.

O funcionamento do módulo **M3**, exemplificado pela **Figura 4** é similar ao **M2** e foca em resolver o passo 5 do pseudocódigo desta seção. Os cálculos aqui dependem diretamente do resultado deixado pelo módulo anterior. Nesse sentido, a máquina irá somar ou subtrair o resultado deixado pela **M2** de acordo com o sinal decorrente do número unário da fita. Sendo assim, a máquina desloca o cabeçote para a direita até encontrar um CIH, caso o CIH seja um '+', a máquina simplesmente somará os valores unários calculados pela **M2** aos valores unários que representam o horário local na origem (H_A), convenientemente posicionados próximos desde a carga inicial da fita, utilizando os mesmos métodos já anteriormente descritos.

Independentemente do caminho tomado, ao final das operações, a máquina irá mover o cabeçote de volta para o início da fita e irá encaminhar o fluxo para o próximo módulo. O resultado das operações deste módulo pode ser interpretado como sendo o H_B da fórmula de base.



O resultado dessa operação pode exceder o valor de vinte e quatro unidades unárias, o que representa uma saída inválida. Para contornar este caso, a máquina sempre verifica se o resultado da soma excedeu o limite ou não. Caso não, a máquina simplesmente irá retornar o valor final e sinalizar o fim das operações retornando o cabeçote para o início da fita. Caso tenha excedido o limite, a máquina irá apagar os vinte e quatro primeiros caracteres '1' da fita. Os caracteres restantes serão considerados a resposta final e a máquina irá sinalizar o fim das operações retornando o cabeçote para o início da fita.

2.3 Complexidade

A complexidade de MT1 é medida ao analisar o pior caso a ser executado pela máquina. Nesta situação, entende-se que o pior caso como aquele em que o voo parte do fuso mais extremo a Leste ($+180^\circ$) até o mais extremo a Oeste (-180°), ou seja, realiza uma volta ao mundo. Além disso, o voo parte a meia noite e dura 24h.

Dado tais condições iniciais, a palavra de entrada totalizará 414 caracteres, ou seja, $n = 414$, e terá a seguinte composição:

$$+(1)^{180}0 - (1)^{180}0(1)^{24}0(1)^{24}$$

Como há diversas etapas, nos quais o cabeçote da fita percorre várias vezes em um ciclo, tem-se algumas equações quadráticas em função de n . Com isso, conclui-se que no pior caso o polinômio de segundo grau limita superiormente a **MT1**, logo o algoritmo da **MT1** é definido como $O(n^2)$.

Ao analisar separadamente cada iteração do cabeçote da **MT1**, obtemos a seguinte equação em termos de n :

$$T_{MT1}(n) = 2n^2 - 990,5n + 82690$$

Para $n = 414$, obtemos a equação que descreve a quantidade de de operações da MT1 no pior caso.

:

$$T_{MT1} = 15415$$

2.4 Exemplos

Para a validação da MT1, foram conduzidos testes que testavam os "*casos de borda*" da máquina, alguns destes testes estão disponíveis na **Tabela 2**, com a relação entre entrada da máquina e saída esperada:

2.4.1 Exemplo de execução 1:

Parâmetros de entrada:

$$(F_B, F_A, D_{voo}, H_A) = (15^\circ \text{ Leste}, 75^\circ \text{ Oeste}, 4h, 15h)$$

1. Estado da fita:

$$\{+(1)^{15}0 - (1)^{75}0(1)^40(1)^{15}0\}$$

Ação: Módulo **M0**, com o cabeçote no início da fita, copia o caractere identificador de fuso para o final da fita e volta para o início.

Tabela 2: Entrada x Saída da MT1

Parâmetros				Entrada Correspondente	Saída
H	F_p	F_d	V	Fita	Fita
15h	75° L	15° O	4h	$+(1)^{75}0 - (1)^{15}0(1)^40(1)^{15}0$	10
2h	150° L	150° O	1h	$+(1)^{150}0 - (1)^{150}0(1)^10(1)^20$	$(1)^{23}0$
0h	150° O	150° L	1h	$-(1)^{150}0 + (1)^{150}010(1)^{24}0$	$(1)^50$
0h	150° O	15° L	50h	$-(1)^{150}0 + (1)^{150}0(1)^{50}0(1)^{24}0$	$(1)^60$
23h	0°	0°	4h	$-(1)^{150}0 + (1)^{150}0(1)^{50}0(1)^{24}0$	$(1)^30$
10h	30° L	30° O	4h	$+(1)^{30}0 - (1)^{30}0(1)^40(1)^{10}0$	$(1)^{22}0$
10h	30° L	90° L	4h	$+(1)^{30}0 + (1)^{90}0(1)^40(1)^{10}0$	$(1)^{10}0$
10h	30° O	75° O	4h	$-(1)^{30}0 - (1)^{75}0(1)^40(1)^{10}0$	$(1)^{17}0$
10h	30° O	30° L	4h	$-(1)^{30}0 + (1)^{30}0(1)^40(1)^{10}0$	$(1)^{10}0$
2h	150° O	150° L	1h	$-(1)^{150}0 + (1)^{150}0(1)^10(1)^20$	$(1)^80$

2. Estado da fita:

$$\{\square(1)^{15}0 - (1)^{75}0(1)^40(1)^{15}0+\}$$

Ação: Módulo **M0**, com o cabeçote no início da fita, divide os valores unários de F_B por 15 e armazena o resultado no final da fita, voltando para o início ao fim da operação.

3. Estado da fita:

$$\{\square\square0 - (1)^{75}0(1)^40(1)^{15}0 + 1\}$$

Ação: Módulo **M0**, com o cabeçote no início da fita, ao ler o caractere '0', copia para o final da fita, voltando para o início ao fim da operação.

4. Estado da fita:

$$\{\square\square\square - (1)^{75}0(1)^40(1)^{15}0 + 10\}$$

Ação: Módulo **M1**, com o cabeçote no início da fita, copia o caractere identificador de fuso para o final da fita e volta para o início.

5. Estado da fita:

$$\{\square(1)^{75}0(1)^40(1)^{15}0 + 10-\}$$

Ação: Módulo **M1**, com o cabeçote no início da fita, divide os valores unários de F_A por 15 e armazena o resultado no final da fita, voltando para o início ao fim da operação.

6. Estado da fita:

$$\{\square\square0(1)^40(1)^{15}0 + 10 - 11111\}$$

Ação: Módulo **M1**, com o cabeçote no início da fita, ao ler o caractere '0', o copia para o final da fita, voltando para o início ao fim da operação.

7. Estado da fita:

$$\{\square\square\square(1)^40(1)^{15}0 + 10 - 111110\}$$

Ação: Módulo **M2**, com o cabeçote no início da fita, percorre toda a palavra até atingir o segundo caractere identificador de fuso e o inverte.

8. Estado da fita:

$$\{\square(1)^40(1)^{15}0 + 10 + 111110\}$$

Ação: Módulo **M2** avança com o cabeçote para a esquerda até encontrar o primeiro caractere identificador de fuso e perceber que os sinais são iguais, então, apaga os caracteres separadores entre os valores unários para iniciar a soma.

9. Estado da fita:

$$\{\square(1)^40(1)^{15}0 + 1\square\square111110\}$$

Ação: Módulo **M2** soma os valores ao deslocar os caracteres mais a direita para os espaços em branco mais a esquerda. Ao final desloca o cabeçote de volta para o início da fita.

10. Estado da fita:

$$\{\square(1)^40(1)^{15}0 + 1111110\square\}$$

Ação: Módulo **M3**, com o cabeçote no início da fita, desloca para a direita até encontrar o caractere identificador de fuso, ao verificar que é um + apaga os caracteres separadores entre os valores unários para iniciar a soma.

11. Estado da fita:

$$\{\square(1)^40(1)^{15}\square\square1111110\square\}$$

Ação: Módulo **M3** realiza a soma com os valores unários imediatamente antecedentes, ao final, desloca o cabeçote de volta para o início.

12. Estado da fita:

$$\{\square(1)^40(1)^{21}0\square\}$$

Ação: Módulo **M3**, com o cabeçote no início da fita, avança para a direita até encontrar o primeiro caractere separador, apagando-o em seguida para preparar a soma.

13. Estado da fita:

$$\{\square(1)^4\square(1)^{21}0\square\}$$

Ação: Módulo **M3** realiza a soma dos valores restantes e retorna o cabeçote para o início da fita.

14. Estado da fita:

$$\{\square(1)^{25}0\square\}$$

Ação: **M4**, com o cabeçote no início da fita, verifica se o resultado da operação anterior excedeu 24 caracteres. Em caso positivo, apaga os caracteres excedentes. Ao final retorna o cabeçote para o início da fita.

14. Estado da fita:

$$\{\square10\square\}$$

Ação: Halt.

2.4.2 Exemplo de execução 2:

Parâmetros de entrada:

$$(F_B, F_A, D_{voo}, H_A) = (45^\circ Oeste, 135^\circ Leste, 24h, 5h)$$

1. Estado da fita:

$$\{-(1)^{45}0 + (1)^{135}0(1)^{24}0(1)^50\}$$

Ação: Módulo **M0**, com o cabeçote no início da fita, copia o caractere identificador de fuso para o final da fita e volta para o início.

2. Estado da fita:

$$\{\square(1)^{45}0 + (1)^{135}0(1)^{24}0(1)^50-\}$$

Ação: Módulo **M0**, com o cabeçote no início da fita, divide os valores unários de F_B por 15 e armazena o resultado no final da fita, voltando para o início ao fim da operação.

3. Estado da fita:

$$\{\square\square0 + (1)^{135}0(1)^{24}0(1)^50 - 111\}$$

Ação: Módulo **M0**, com o cabeçote no início da fita, ao ler o caractere '0', copia para o final da fita, voltando para o início ao fim da operação.

4. Estado da fita:

$$\{\square\square\square + (1)^{135}0(1)^{24}0(1)^50 - 1110\}$$

Ação: Módulo **M1**, com o cabeçote no início da fita, copia o caractere identificador de fuso para o final da fita e volta para o início.

5. Estado da fita:

$$\{\square(1)^{135}0(1)^{24}0(1)^50 - 1110+\}$$

Ação: Módulo **M1**, com o cabeçote no início da fita, divide os valores unários de F_A por 15 e armazena o resultado no final da fita, voltando para o início ao fim da operação.

6. Estado da fita:

$$\{\square\square0(1)^{24}0(1)^50 - 1110 + 111111111\}$$

Ação: Módulo **M1**, com o cabeçote no início da fita, ao ler o caractere '0', o copia para o final da fita, voltando para o início ao fim da operação.

7. Estado da fita:

$$\{\square\square\square(1)^{24}0(1)^50 - 1110 + 1111111110\}$$

Ação: Módulo **M2**, com o cabeçote no início da fita, percorre toda a palavra até atingir o segundo caractere identificador de fuso e o inverte.

8. Estado da fita:

$$\{\square(1)^{24}0(1)^50 - 1110 - 1111111110\}$$

Ação: Módulo **M2** avança com o cabeçote para a esquerda até encontrar o primeiro caractere identificador de fuso e perceber que os sinais são iguais, então, apaga os caracteres separadores entre os valores unários para iniciar a soma.

9. Estado da fita:

$$\{\square(1)^{24}0(1)^50 - 111\square\square1111111110\}$$

Ação: Módulo **M2** soma os valores ao deslocar os caracteres mais a direita para os espaços em branco mais a esquerda. Ao final desloca o cabeçote de volta para o início da fita.

10. Estado da fita:

$$\{\square(1)^{24}0(1)^50 - 1111111111110\square\}$$

Ação: Módulo **M3**, com o cabeçote no início da fita, desloca para a direita até encontrar o caractere identificador de fuso, ao verificar que é um $-$ desloca o cabeçote para a esquerda até encontrar um '0'. Então desloca o cabeçote para a direita, e, ao ler um '1', apaga o caractere e escreve um 'X' em seu lugar, deslocando o cabeçote para a direita até encontrar o CIH.

11. Estado da fita:

$$\{\square(1)^{24}0X11110 - 111111111110\square\}$$

Ação: Ao encontrar o primeiro caractere '1' após o CIH, o Módulo **M3** o substitui por um 'X' e retorna o cabeçote para a esquerda até encontrar um 'X'. Então desloca o cabeçote para a direita, e, ao ler um '1', apaga o caractere e escreve um 'X' em seu lugar, deslocando o cabeçote para a direita até encontrar o CIH. A máquina irá transitar entre o passo anterior e este até que não haja mais '1' à esquerda do penúltimo separador 0.

12. Estado da fita:

$$\{\square(1)^{24}0XXXXX0 - XXXXX11111110\square\}$$

Ação: percebendo que o lado direito do penúltimo separador é maior que o lado esquerdo, o Módulo **M3** substitui todos os caracteres 'X' por '1' e prepara o sistema para a subtração, ao remover o separador e o CIH entre os valores.

13. Estado da fita:

$$\{\square(1)^{24}011111\square\square111111111110\square\}$$

Ação: o Módulo **M3** realiza a subtração do valor mais a direita pelo valor mais a esquerda, por remover um caractere '1' do valor à direita, pra cada valor '1' do valor à esquerda

14. Estado da fita:

$$\{\square(1)^{24}0\square\square\square\square\square\square\square\square\square\square\square\square11111110\square\}$$

Ação: o Módulo **M3** desloca os caracteres '1' mais a direita para junto do primeiro separador

15. Estado da fita:

$$\{\square(1)^{24}0111111110\square\}$$

Ação: o Módulo **M3** escreve vinte e quatro caracteres '1' após o último separador

16. Estado da fita:

$$\{\square(1)^{24}0111111110(1)^{24}0\}$$

Ação: **M3** apaga o segundo separador e prepara para mais uma subtração

17. Estado da fita:

$$\{\square(1)^{24}01111111\square(1)^{24}0\}$$

Ação: **M3** subtrai o valor mais à esquerda do valor mais à direita

18. Estado da fita:

$$\{\square(1)^{24}0\square\square\square\square\square\square\square\square\square\square\square\square\square\square111111111111111110\}$$

Ação: **M3** desloca os caracteres '1' para à esquerda, ao final, desloca o cabeçote para o início.

19. Estado da fita:

$$\{\square(1)^{24}01111111111111111110\}$$

Ação: em **M4**, com o cabeçote no início da fita, desloca até o primeiro separador, e o apaga, preparando para um soma.

20. Estado da fita:

$$\{\square(1)^{24}\square(1)^{17}0\}$$

Ação: o módulo **M4**, realiza a soma, deslocando todos os caracteres '1' do valor mais a direita para a esquerda. Ao final, desloca o cabeçote para o início da fita.

21. Estado da fita:

$$\{\square(1)^{41}0\}$$

Ação: **M4**, com o cabeçote no início da fita, verifica se o resultado da operação anterior excedeu 24 caracteres 1. Em caso positivo, apaga os caracteres excedentes. Ao final retorna o cabeçote para o início da fita.

22. Estado da fita:

$$\{\square(1)^{17}0\square\}$$

Ação: Halt.

3 Solução para MT2

Existem várias definições alternativas de máquinas de Turing, incluindo versões com múltiplas fitas ou com não-determinismo. Elas são chamadas de variantes do modelo de máquina de Turing. O modelo original e suas variantes razoáveis têm todos o mesmo poder, ou seja, reconhecem a mesma classe de linguagens [1].

A segunda máquina de Turing desenvolvida mantém o princípio básico de operação da máquina original de Turing com modificações: a MT2 incorpora uma segunda fita, funciona sem o símbolo ' $<$ ' que limita as fitas para a esquerda e aceita por parada. A inclusão de múltiplas fitas em uma máquina de Turing oferece vantagens na resolução de problemas.

Uma delas é a função de memória adicional, permitindo armazenar dados ou executar funções em paralelo com o código principal. Isso pode resultar em uma economia de operações, pois evita a necessidade de percorrer continuamente uma única fita extensa. Em cenários onde a fita primária se torna excessivamente longa, alternar para uma segunda fita pode facilitar o processamento.

À medida que a quantidade de fitas aumenta, podemos identificar semelhanças entre a máquina de Turing e os computadores modernos, antecipando assim o desenvolvimento das memórias utilizadas nas arquiteturas contemporâneas. A inclusão de múltiplas fitas na máquina de Turing amplia a versatilidade das MTs, aproximando-a ainda mais do funcionamento dos computadores atuais.

As máquinas de Turing de múltiplas fitas parecem ser mais flexíveis do que as máquinas de Turing comuns, mas podemos mostrar que elas são equivalentes em poder. Toda máquina de Turing de múltiplas fitas tem uma máquina de Turing de fita única equivalente [1].

MT2 faz cálculos usando a base numérica unária. A equação de base usada na MT2 para descrever o horário da chegada do avião é:

$$H_d = H_p + V + \frac{F_p}{15} - \frac{F_d}{15}$$

onde H_d é o *horário local na cidade de destino*, H_p é o *horário local na cidade de partida*, V é a *duração do voo*, F_p é o *fuso horário do local de partida* e F_d é o *fuso horário do local de destino*.

3.1 Definição Formal

A MT2 projetada segue a definição formal, formada pela seguinte 9-upla, que é complementada pela tabela 3:

$$MT2 = (Q, \Sigma^1, \Sigma^2, \Gamma^1, \Gamma^2, s, b, F, \delta)$$

Tabela 3: Descrição dos Símbolos, significados de cada elemento da MT2.

Símbolo	Significado	Valor
Q	Conjunto finito de estados de MT2	Os estados estão descritos na figura 6.
Σ^i	O alfabeto da entrada de MT2 para cada fita	$\Sigma^i = \{1, D, E\}$
Γ^i	É o alfabeto da fita i , onde $b \in \Gamma^i$ e $\Sigma^i \subseteq \Gamma^i$	$\Gamma^i = \{1, O, L\}, \forall i \in \mathbb{N}$
s	Estado inicial de MT2	$s = \text{start}$
b	Símbolo branco tal que $b \in \Gamma^i$	$b = \square$
F	Conjunto dos estados finais de MT2 tal que $F \subseteq Q$	$F = \{\text{mod}24\}$
δ	Função de transição $\delta : Q \times \Gamma \rightarrow Q \times \Gamma \times \{E, D, S\}$, que descreve o comportamento de máquina de Turing em seus estados e símbolos lidos.	As funções de transições estão descritas na figura 6.

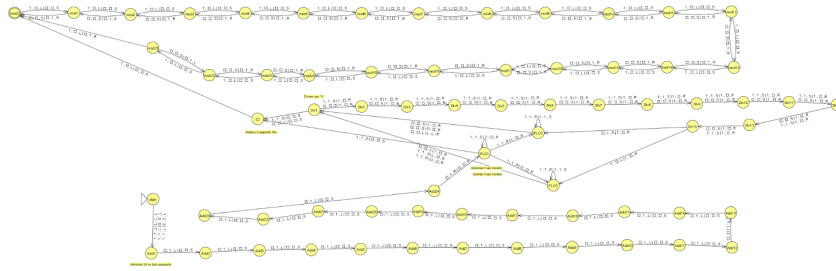


Figura 6: Espaço de estados Q e funções de transição de MT2.

As possíveis entradas para MT2 são limitadas pela linguagem que MT2 consegue reconhecer e são formadas por uma tupla w_1, w_2 inseridas nas fitas 1 e 2. w_1 e w_2 são uma concatenação de outras 2 palavras, elas representam as informações de entrada: V , a duração do voo em horas, H_p a hora de partida (na cidade de origem), L_o é a longitude da cidade de origem e L_d é longitude da cidade de destino, formalmente definida como:

$$L(MT2) = \{(VH_p, L_dL_o) | V \in L_v, H_p \in L_{H_p}, L_o \in L_F, L_d \in L_F\}$$

$$L_v = \{w | w \in \{1\}^+\}$$

$$L_{H_p} = \{w | w \in \{1\}^i, \forall 0 \leq i \leq 23\}$$

$$L_F = \{dw | d \in \{O, L, Z\}, w \in \{1\}^i, \forall 0 \leq i \leq 150, i \bmod 15 = 0\}$$

A saída de MT2 é o horário de chegada do voo na base unária. É escrito na segunda fita e representada pela linguagem:

$$L_{exit} = \{w | w \in \{1\}^i, \forall 0 \leq i \leq 23\}$$

3.2 Módulos

O funcionamento de MT2 pode ser descrito pelo seguinte pseudocódigo, logo depois de serem inseridas as palavras de entrada:

1. Escreva na primeira fita, na esquerda da palavra recebida, 24 caracteres '1'.
2. Volte o ponteiro para o fim da primeira palavra.
3. Tomando como base o caractere 'L', 'O' ou 'Z' da palavra na segunda fita, adicione ou subtraia o resultado da divisão do número unário por 15 da palavra contida na primeira fita. Se 'L' (leste, GMT positivo), então some, se 'O' (oeste, GMT negativo) subtraia, se 'Z' (zero) passe uma casa.
4. Divida o número unário por 15 e com base na decisão do passo 3, modifique a palavra na fita 1. O módulo de divisão, no processo, apaga caracteres na fita 2.
5. Repita o passo 3 uma vez, agora com uma tomada de decisão invertida: Se 'L' (leste, GMT positivo), então subtraia. Caso 'O' (oeste, GMT negativo) some e se 'Z' (zero) passe uma casa.
6. Divida o número unário por 15 e com base na decisão do passo 5, modifique a palavra na fita. O módulo de divisão, no processo, apaga caracteres na fita 2.
7. Calcule o resto da divisão de 24 do número restante na primeira fita.
8. Escreva o resultado na fita 2.

Salienta-se que, ao inserir duração do voo e horário de partida juntos na base unária na primeira fita, como consequência fornece o número somado em unário dos dois.

Destaca-se também que para calcular o horário final, é usado o resto da divisão por 24, que é útil em casos onde o horário calculado supera 24 horas. O passo 1 é uma estratégia usada para os casos em que as subtrações realizadas resultam em resultados negativos, onde resultados como -1 e -2 equivalem respectivamente a 23 horas e 22 horas, e são convertidos usando a operação de resto no final.

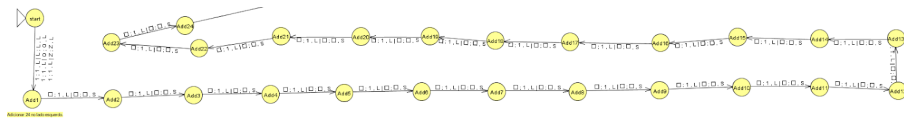


Figura 7: Módulo *Add24* da MT2.

A **Figura 7** descreve o conjunto de estados, $\{Add1, \dots, Add24\}$ que formam o módulo *Add24* e é executado em todos os casos. A segunda fita permanece parada, enquanto na primeira fita é escrito 24 caracteres '1' com um deslocamento para a esquerda.

3.3 Complexidade

Para analisar a complexidade de MT2, é preciso analisar com cuidado o algoritmo executado por ela no pior caso. O pior caso está relacionado essencialmente à duração do voo, já que dos parâmetros inseridos em MT2 é o único que não tem limite superior, por isso quanto maior a duração do voo, mais operações serão realizadas. Para melhor detalhamento, podemos descrever o número de MT2 pela seguinte equação:

$$\begin{aligned} T_{MT2}(w_1, w_2) = & T_{Add24}(w_1, w_2) \\ & + |w_1| + 24 \\ & + 2.(T_{IFLO}(w_3, w_4) + T_{Div15}(w_4, w_5)) \\ & + T_{mod24}(w_6, w_7) \end{aligned}$$

Na primeira execução do módulo *IFLO*, a palavra na fita um deve ser percorrida novamente até o seu final para posicionamento do ponteiro corretamente, essa operação cresce de acordo com a duração do voo, o horário de partida e um custo fixo de 24 inseridos pelo módulo *Add24*. Essa é a razão da soma de $|w_1| + 24$ na equação.

Serão executadas 2 vezes a operação de decisão e a divisão do grau do meridiano. No pior caso, o caráter de orientação será primeiro ‘L’ e depois ‘O’, desencadeando duas operações de soma, adicionando mais 20 no número da fita 1, porque afetará a complexidade do cálculo de resto em *mod24*.

Existe um custo fixo de 25 transições executadas no total no estado inicial e módulo *Add24* no início da máquina.

$$T_{Add24}(w_1, w_2) = 25$$

Existe um custo fixo de 2 responsáveis pela decisão de somar ou subtrair em *IFLO*.

$$T_{IFLO}(w_1, w_2) = 2$$

Em geral, *Div15* tem a complexidade baseada em $|L_d|$ e $|L_o|$ proveniente de w_2 . Mas no pior caso, os graus serão máximos, respectivamente 150 cada, resultando num custo fixo de 150.

$$T_{Div15}(w_1, w_2) = 150$$

Por fim, já que o cálculo do resto do número gerado na fita 1 não tem custo fixo no pior caso, já que a duração do voo não tem limite superior. A operação de também exige o custo fixo de 2 para transição do módulo *Div15*.

Entretanto, deve-se contabilizar as operações sobre os 20 caracteres adicionados previamente em *Div15* e os 24 caracteres adicionados em *add24* e o custo de ações para escrever o resto na fita 2, que no pior caso é 23. Por isso *mod24* tem a execução de tarefas descrita por:

$$T_{mod24}(w_6, w_7) = |w_1| + 20 + 24 + 23 + 2$$

Destaca-se que no pior caso, todas as operações tem custo fixo, com exceção do ajuste do ponteiro e cálculo de resto, que tem custo linear de acordo com o tamanho de w_1 :

$$T_{MT2}(w_1, w_2) = 25 + |w_1| + 24 + |w_1| + 2.(2 + 150) + 20 + 24 + 23 + 2$$

$$T_{MT2}(w_1, w_2) = 2.|w_1| + 442$$

Com o número de operações definidas, é trivial concluir que no pior caso um polinômio de primeiro grau consegue limitar superiormente MT2 no pior caso, consequentemente, definindo o algoritmo de MT2 como $O(n)$:

$$2 \cdot |w_1| + C \geq 2 \cdot |w_1| + 442$$

$$C \geq 442$$

A fim de atingir o menor número de operações para se obter o resultado, existem modificações a serem feitas para melhorar a eficiência da máquina, como por exemplo expressar os números de outras bases diminuindo a complexidade das operações. No entanto, a simplicidade foi o objetivo principal por trás da MT2.

3.4 Exemplos

Novamente, para a fase de testes da MT2, o grupo conduziu um conjunto de testes que testavam os “casos de borda” da máquina, alguns destes testes estão disponíveis na tabela 4, com a relação entre entrada da máquina e saída esperada:

Tabela 4: Entrada x Saída da MT2

Parâmetros				Entrada Correspondente		Saída	
H	F_p	F_d	V	Fita 1	Fita 2	Fita 1	Fita 2
15h	15° L	75° O	4h	$(1)^{15}(1)^4$	$L(1)^{15}O(1)^{75}$	□	$(1)^1$
2h	150° L	150° O	1h	$(1)^2(1)^1$	$L(1)^{150}O(1)^{150}$	□	$(1)^{23}$
0h	150° O	150° L	1h	$(1)^1$	$O(1)^{150}L(1)^{150}$	□	$(1)^5$
0h	150° O	15° L	50h	$(1)^{50}$	$O(1)^{150}L(1)^{15}$	□	$(1)^6$
23h	0°	0°	4h	$(1)^{23}(1)^4$	ZZ	□	$(1)^3$
10h	30° L	30° O	4h	$(1)^{10}(1)^4$	$L(1)^{30}O(1)^{30}$	□	$(1)^{22}$
10h	30° L	90° L	4h	$(1)^{10}(1)^4$	$L(1)^{30}L(1)^{90}$	□	$(1)^{10}$
10h	30° O	75° O	4h	$(1)^{10}(1)^4$	$O(1)^{30}O(1)^{75}$	□	$(1)^{17}$
10h	30° O	30° L	4h	$(1)^{10}(1)^4$	$O(1)^{30}L(1)^{30}$	□	$(1)^{10}$
2h	150° O	150° L	1h	$(1)^{15}(1)^1$	$O(1)^{150}L(1)^{150}$	□	$(1)^8$

3.4.1 Exemplo de execução 1:

Parâmetros de entrada:

$$(H_p, F_p, F_D, V) = (0h, 150^\circ Oeste, 150^\circ Leste, 1h)$$

1. Estado das fitas:

$$Fita1 : \{1\}$$

$$Fita2 : \{O(1)^{150}L(1)^{150}\}$$

Ação: Módulo *Add24*, adiciona 24 caracteres na primeira fita.

2. Estado das fitas:

$$Fita1 : \{111111111111111111111111111111\}$$

$$Fita2 : \{O(1)^{150}L(1)^{150}\}$$

Ação: Módulo *IFLO* lê ‘O’ e chama *Div15* com a transição para subtração.

3. Estado das fitas:

$$Fita1 : \{11111111111111111111111111111111\}$$

$$Fita2 : \{(1)^{150}L(1)^{150}\}$$

Ação: Módulo *Div15* é executado 10 vezes, apagando 10 caracteres na fita 1.

4. Estado das fitas:

$$Fita1 : \{1111111111111111\}$$

$$Fita2 : \{L(1)^{150}\}$$

Ação: Módulo *IFLO* lê ‘L’ e chama *Div15* com a transição para subtração.

5. Estado das fitas:

$$Fita1 : \{1111111111111111\}$$

$$Fita2 : \{(1)^{150}\}$$

Ação: Módulo *Div15*, executado 10 vezes, apagando 10 caracteres na fita 1.

6. Estado das fitas:

$$Fita1 : \{1111111111111111\}$$

$$Fita2 : \{\}$$

Ação: Módulo *mod24*, calculando o resto da divisão por 24 número da primeira fita.

7. Estado das fitas:

$$Fita1 : \{\}$$

$$Fita2 : \{11111\}$$

Ação: Halt.

3.4.2 Exemplo de execução 2:

Parâmetros de Entrada:

$$(H_p, F_d, F_p, V) = (23h, 0^\circ, 0^\circ, 4h)$$

1. Estado das fitas:

$$Fita1 : \{11111111111111111111111111111111\}$$

$$Fita2 : \{ZZ\}$$

2. Estado das fitas:

$$Fita2 : \{ZZ\}$$

3. Estado das fitas:

$$Fita2 : \{Z\}$$

4. Estado das fitas:

$$Fita2 : \{\}$$

5. Estado das fitas:

$$Fita2 : \{111\}$$

3.4.3 Exemplo de execução 3:

$$(H_p, F_d, F_p, V) = (15h, 15^\circ Oeste, 75^\circ Leste, 5h)$$

1. Estado das fitas:

$$Fita2 : \{L(1)^{75}O(1)^{15}\}$$

2. Estado das fitas:

$$Fita2 : \{L(1)^{75}O(1)^{15}\}$$

3. Estado das fitas:

$$Fita2 : \{(1)^{75}O(1)^{15}\}$$

4. Estado das fitas:

$$Fita2 : \{O(1)^{15}\}$$

5. Estado das fitas:

$$Fita2 : \{(1)^{15}\}$$

6. Estado das fitas:

$$Fita2 : \{\}$$

7. Estado das fitas:

$$Fita2 : \{1\}$$

23

4 Considerações Finais

A implementação de dois modelos distintos de máquinas de Turing, MT1 e MT2, para resolução de um problema simples proporcionou uma análise de suas capacidades e limitações. Destaca-se o entendimento da mecânica computacional, a versatilidade e a eficiência desses modelos em simular processos.

Ao abordar o cálculo simples de horários de voos internacionais, este experimento aplicou as MTs num cenário prático e uma abordagem simples. Desta forma, é possível explorar conceitos essenciais da teoria da computação, como complexidade e decidibilidade, por exemplo.

Mesmo sendo produtos da década de 1930, as máquinas de Turing permanecem fundamentais no estudo da decidibilidade de problemas e na compreensão dos princípios que norteiam o funcionamento dos computadores modernos. A durabilidade deste modelo, que até o momento não foi superado em termos de poder computacional, ressalta a sua importância histórica e contínua na evolução da computação.

Referências

- [1] SIPSER, Michael. *Michael. Introduction to the Theory of Computation*. 2ª Edição. Boston: Thomson Course Technology, 2006.
- [2] VIEIRA, Newton José. *Linguagens e Máquinas: Uma Introdução aos Fundamentos da Computação*. 1ª Edição. Belo Horizonte: Departamento de Ciência da Computação, Instituto de Ciências Exatas, Universidade Federal de Minas Gerais, 2004.
- [3] GEEKS FOR GEEKS. *Turing Machine as Comparator*. Geeks for Geeks, 2023. Disponível em: <https://www.geeksforgeeks.org/turing-machine-as-comparator/>
Acesso em: 20 de novembro de 2023