



UNIVERSIDADE FEDERAL DE SANTA MARIA
CENTRO DE TECNOLOGIA
ELC1008 - TEORIA DA COMPUTAÇÃO

Guilherme Fereira Da Silva

Leandro Brum da Silva Lacorte

**RELATÓRIO ACADÊMICO: IMPLEMENTAÇÃO DE UMA MÁQUINA DE
TURING QUÂNTICA (MTQ)**

Santa Maria, RS

2025

SUMÁRIO

1 FUNDAMENTAÇÃO TEÓRICA DA MÁQUINA DE TURING QUÂNTICA (MTQ)	3
2 ESTRATÉGIAS DE IMPLEMENTAÇÃO ADOTADAS.....	4
2.1 Estrutura de Dados.....	4
2.2 Representação da Superposição.....	4
2.3 Transições e Evolução.....	4
2.4 Medição.....	5
2.5 Implementação de Exemplos.....	5
2.6 Design Modular.....	5
3 CASOS DE TESTE E RESULTADOS.....	6
3.1 Algoritmo de Deutsch.....	6
3.2 Linguagem $A = \{0^n 1^n\}$	6
4 DISCUSSÃO SOBRE SUPERPOSIÇÃO E REVERSIBILIDADE.....	7
4.1 Superposição.....	7
4.2 Reversibilidade.....	7
4.3 Importância para Computação Quântica.....	8
5 CONCLUSÃO.....	8

1 FUNDAMENTAÇÃO TEÓRICA DA MÁQUINA DE TURING QUÂNTICA (MTQ)

A Máquina de Turing Quântica (MTQ) é uma extensão quântica do modelo clássico de Máquina de Turing, incorporando os princípios da **computação quântica**, tais como:

1. Superposição

- Diferente da MT clássica, que possui um único estado em um instante, a MTQ permite que múltiplas configurações coexistam simultaneamente.
- Cada configuração clássica é associada a uma **amplitude complexa** $\alpha \in \mathbb{C}$, representando sua contribuição na superposição:

$$|\Psi\rangle = \sum_i \alpha_i |c_i\rangle$$

onde $|c_i\rangle$ é uma configuração clássica.

- A probabilidade de observar uma configuração c_i após a medição é $P(c_i) = |\alpha_i|^2$.

2. Reversibilidade

- Todas transições quânticas devem ser unitárias, preservando a soma total das probabilidades ($\sum_i |\alpha_i|^2 = 1$).
- Isso garante que nenhuma informação é perdida, refletindo o princípio da reversibilidade da mecânica quântica.

3. Interferência

- Amplitudes de diferentes caminhos computacionais podem **interferir construtivamente ou destrutivamente**, alterando a probabilidade final de uma configuração.
- Permite algoritmos como Deutsch, Grover e Shor explorarem paralelismo quântico.

4. Medida quântica

- A medição colapsa a superposição em uma configuração clássica, escolhida probabilisticamente de acordo com as amplitudes.
- Esse colapso é o momento em que a MTQ fornece um resultado observável.

Exemplo de uso teórico no projeto:

- **Algoritmo de Deutsch:** decide se uma função $f:\{0,1\}\rightarrow\{0,1\}$ é constante ou balanceada em **uma única execução**, usando superposição e interferência.
- **Linguagem A = $\{0^n1^n\}$:** simula múltiplos caminhos de marcação simultâneos na fita, mostrando paralelismo intrínseco da MTQ.

2 ESTRATÉGIAS DE IMPLEMENTAÇÃO ADOTADAS

2.1 Estrutura de Dados

- **Configuração clássica** (`config_t`): estado atual, posição da cabeça e fita.
- **Configuração quântica** (`quantum_config_t`): configuração clássica + amplitude complexa.
- **Regras de transição** (`rule_t`): representam δ quântico, incluindo movimento da cabeça e multiplicação da amplitude.

2.2 Representação da Superposição

- Cada passo do simulador mantém um **vetor de** `quantum_config_t`, representando a superposição atual.
- Limite de `MAX_CONFIGS = 200` configurações para controlar crescimento exponencial.

2.3 Transições e Evolução

- Função `transicao()`:
 - Para cada configuração da superposição, aplica todas as regras compatíveis.

- Gera novas configurações, multiplicando amplitudes pela regra.
- Combina configurações iguais (`merge_equal_configs`) somando amplitudes para interferência.
- Normaliza amplitudes (`normalize`) para manter soma de probabilidades igual a 1.

2.4 Medição

- `medir()`:
 - Calcula probabilidades de cada configuração.
 - Sorteia uma configuração de acordo com $|amplitude|^2$.
 - Representa o **colapso quântico**, determinando saída observável.

2.5 Implementação de Exemplos

- **Deutsch**
 - Inicializa superposição de entrada $(|0\rangle + |1\rangle)/\sqrt{2}$.
 - Aplica oráculo `Uf` e Hadamard final.
 - Mede para decidir se função é constante ou balanceada.
- **Linguagem A**
 - Marca zeros e busca 1 correspondente em superposição, permitindo múltiplos caminhos simultâneos.
 - Demonstrou paralelismo quântico em reconhecimento de padrões.

2.6 Design Modular

- Headers (`mtq.h`, `quantum.h`, `util.h`) e arquivos de implementação (`quantum.c`, `util.c`) separados, permitindo:
 - Reuso de funções quânticas genéricas.

- Extensão para outras linguagens ou algoritmos.

3 CASOS DE TESTE E RESULTADOS

3.1 Algoritmo de Deutsch

- Entrada: função f constante zero ($f(x)=0, f(x)=0, f(x)=0$).
- Resultado esperado: função constante.
- Saída observada:

```
Amplitude escolhida: (1.000000+0.000000i)
Config medida: fita[b0] head=0 state=QF
Decisão: FUNCAO CONSTANTE
```

- Entrada: função f identidade ($f(0)=0, f(1)=1, f(0)=0, f(1)=1, f(0)=0, f(1)=1$).
- Resultado esperado: função balanceada.
- Saída observada:

```
Amplitude escolhida: (1.000000+0.000000i)
Config medida: fita[b1] head=0 state=QF
Decisão: FUNCAO BALANCEADA
```

Análise:

- Superposição inicial e interferência permitiram determinar a classe da função **em um passo**, confirmando eficiência quântica.

3.2 Linguagem $A = \{0^n 1^n\}$

- Entrada: 000111
- Processo:

- Cabeça marca **0**, move à direita, encontra **1** correspondente, retorna ao próximo **0**.
- Superposição gerencia diferentes caminhos de marcação simultaneamente.
- Resultado final: estado final QF com fita completamente marcada.
- Probabilidade 1 após medição, indicando aceitação da entrada.
- Entrada inválida: **00111**
- Resultado: estado final diferente de QF ou cabeças em posições inválidas, indicando rejeição.

Análise:

- O simulador reproduziu corretamente comportamento determinístico esperado da linguagem clássica.
- Superposição permitiu explorar múltiplos caminhos de verificação em paralelo.

4 DISCUSSÃO SOBRE SUPERPOSIÇÃO E REVERSIBILIDADE

4.1 Superposição

- Permite que múltiplas configurações coexistam e evoluam simultaneamente.
- No algoritmo de Deutsch, possibilita **paralelismo quântico**, avaliando $f(0)$ e $f(1)$ ao mesmo tempo.
- Na linguagem A, possibilita marcar zeros simultaneamente, acelerando a exploração de caminhos possíveis.

4.2 Reversibilidade

- Cada regra é projetada para **não destruir informação**.
- Evolução quântica unitária mantém norma da superposição igual a 1.

- Essencial para interferência: caminhos reversíveis podem interferir construtiva ou destrutivamente, alterando probabilidades finais.

4.3 Importância para Computação Quântica

- Sem superposição e reversibilidade, algoritmos quânticos perdem vantagem sobre métodos clássicos.
- Permite eficiência exponencial em problemas como Deutsch, Simon, Grover e Shor.
- Demonstra princípios de computação quântica em uma implementação educacional controlada.

5 CONCLUSÃO

A implementação da MTQ:

- Demonstra conceitos fundamentais de **computação quântica**: superposição, interferência, reversibilidade e medição probabilística.
- Permite simular algoritmos quânticos (Deutsch) e reconhecimento de linguagens clássicas (0^n1^n) com controle de superposição.
- Oferece um framework modular e extensível para futuras implementações de algoritmos quânticos ou outras linguagens formais.
- Os testes realizados confirmam a correção das regras, evolução quântica e comportamento esperado, validando a abordagem.