



Universidade Federal de Pernambuco
Centro de Informática

Graduação em Engenharia da Computação

**Caos na Dinâmica de Neurônios
Quânticos Sem Peso**

Fernando Maciano de Paula Neto

Trabalho de Graduação

Recife
08 de abril de 2014

Universidade Federal de Pernambuco
Centro de Informática

Fernando Maciano de Paula Neto

Caos na Dinâmica de Neurônios Quânticos Sem Peso

Trabalho apresentado ao Programa de Graduação em Engenharia da Computação do Centro de Informática da Universidade Federal de Pernambuco como requisito parcial para obtenção do grau de Bacharel em Engenharia da Computação.

Orientadora: *Profa. Teresa Bernarda Ludermir*
Co-orientador: *Prof. Wilson Rosa de Oliveira Jr.*

Recife
08 de abril de 2014

*A todos aqueles que se dedicam em deixar um legado
positivo à humanidade.*

Agradecimentos

Este trabalho encerra um ciclo de atividades ricas e desafiadoras.

Agradeço a todos aqueles que alimentaram o nosso processo de descobrimento, de ciência, de incertezas em cada etapa de busca pelo conhecimento. Àqueles que mostraram o caminho, que resolveram equações ou inequações, e aos que criaram problemas para que nós resolvéssemos. Aos mais sábios professores e aos em formação, nosso obrigado pelo desafio e pela maturidade despertada em nós.

À família que acreditou no meu potencial até quando não acreditei, que me acolheu nos momentos de dificuldade em que os desafios pareciam ter saturado o domínio das certezas. A minha mãe e a meu pai (in memoriam) e a meus irmãos, meu carinho, companherismo e agradecimentos eternos.

Aos meus professores orientadores Teresa e Wilson a quem tenho muita admiração, respeito e inspiração. Ao doutorando Adenilton, meus agradecimentos por também me orientar quando me coube.

Reitero meus agradecimentos a todos aqueles que persistem na labuta pelo conhecimento, de forma generosa, humilde e compassiva à humanidade. Que faz de sua busca em resolver problemas uma oportunidade de mudanças. Alguns conscientes desse processo outros não.

Obrigado.

Cuide dos meios. O fim cuidará de si mesmo.

—GANDHI

Resumo

Complexidade é um conceito encontrado diariamente em diversos contextos e sistemas que funcionam ao nosso redor. Ela surge da iteração massiva através de diferentes partes de sistemas não-lineares ou em um fenômeno físico que é intrinsecamente complexo. Exemplos deles são sistemas biológicos, fenômenos relacionados ao tempo, turbulências em fluidos, retrodifusão de superfícies marítimas, e multicaminhos de sistema de comunicação móvel. Complexidade é uma parte inseparável do mundo dos sistemas dinâmicos não lineares. O objetivo desse trabalho é analisar as dinâmicas de dois modelos de neurônios artificiais quânticos, o qRAM e o $|\psi\rangle$ -RAM. Podemos verificar através de ferramentas matemáticas que as dinâmicas propostas apresentam sensibilidade significativa às condições iniciais e apresentam simetria representativa nos seus Conjuntos Julia.

Palavras-chave: Não-Linearidade, Caos, Computação Quântica, Redes Neurais Quânticas Sem Peso

Sumário

1	Introdução	1
2	Sistemas Dinâmicos	3
2.1	Orbitais	3
	Tipos de orbitais	3
	Atração ou Repulsão	4
2.2	Bifurcações	4
2.3	Diagrama do Orbital	5
2.4	Conjunto Julia	6
2.5	Mandelbrot	7
3	Computação Quântica	11
3.1	Bit Quântico	11
3.2	Operador Quântico	11
3.3	Paralelismo	12
4	Neurônios Artificiais	13
4.1	Neurônio Clássico	13
4.2	Redes Neurais Clássicas Sem Peso	13
4.2.1	Nó RAM	14
4.2.2	Nó Lógico Probabilístico (Probabilistic Logic Node) PLN	14
4.2.3	MPLN - PLN multivalorado (Multi-valued PLN)	15
4.2.4	pRAM - Probabilistic RAM	15
4.3	Neurônio Quântico Sem Peso	16
4.3.1	qRAM	17
	Exemplo 1 do qRAM	19
	Exemplo 2 do qRAM	19
4.3.2	qPLN	20
	Exemplo 1 do qPLN	20
	Exemplo 2 do qPLN	22
4.3.3	Quantum Multi-valued PLN: qMPLN	22
4.3.4	$ \psi\rangle$ - RAM	23
4.3.5	Nó PQM	24
	4.3.5.1 Memórias Quânticas Probabilísticas MQPs	24

5	Modelos de Dinâmicas para Neurônios Quânticos Sem Peso	27
5.1	Modelo de dinâmica simples para o qRAM (mds-qRAM)	28
5.1.1	Medindo o qubit de saída	28
5.1.2	Análise	32
5.2	Modelo de dinâmica quântica do Kiss et al.	39
5.3	Modelo de dinâmica simples com estado puro para qRAM (mdsep-qRAM)	40
5.4	Modelo de dinâmica elitista não linear de estado puro para qRAM (mdenlep-qRAM)	41
5.4.1	Análise	42
5.5	Modelo de dinâmica simples para o $ \psi\rangle$ -RAM (mds- $ \psi\rangle$ -RAM)	46
5.6	Modelo de dinâmica não-linear de estado puro para o $ \psi\rangle$ -RAM (mdnlep- $ \psi\rangle$ -RAM)	49
5.6.1	Análise	50
5.7	Comparação Quantitativa dos modelos	52
5.7.1	Definições	52
5.7.2	Medida de comparação	54
6	Conclusão	57

CAPÍTULO 1

Introdução

Tem crescido as pesquisas, nos últimos 30 anos, na área de análise em sistemas dinâmicos não-lineares [1] [2] [10], principalmente para entender como sistemas dinâmicos se comportam diante de condições iniciais diferentes [11]. Tal interesse provém principalmente do fato de que a iteração massiva de diferentes partes de sistemas não-lineares, ou em um fenômeno físico que é intrinsecamente complexo, compõe o comportamento de diversos fenômenos encontrados ao nosso redor, como os sistemas biológicos, fenômenos relacionados ao tempo, turbulências em fluidos, retrodifusão de superfícies marítimas e multicaminhos de sistema de comunicação móvel [12]. Para cientistas, a disponibilidade de processamento computacional em alta velocidade e, particularmente, a computação gráfica permitiram grandes avanços na área [13]. Complexidade é um conceito encontrado diariamente em diversos contextos e sistemas que funcionam ao nosso redor e é uma parte inseparável do mundo dos sistemas dinâmicos não lineares. Como acontece em sistemas lineares, a modelagem ajuda a entender o comportamento desses sistemas.

Entender a dinâmica de modelos da inteligência artificial, seus graus de complexidade, parâmetros, bifurcações e caos dá ao cientista entendimento de como esses sistemas se correlacionam com indivíduos de mesmo tipo (formando redes) ou com agentes externos (formando sistemas). O objetivo deste estudo é estudar a dinâmica dos modelos de neurônios quântico sem peso que gere CAOS.

No Capítulo (2) iremos introduzir alguns conceitos sobre os Sistemas Dinâmicos e algumas ferramentas importantes de suas análises que usamos nos modelos propostos. A Computação Quântica é explicada no Capítulo (3), onde introduzimos os conceitos básicos para entendimento desse trabalho. Uma visita à literatura dos neurônios artificiais sem peso é feita no Capítulo (4) assim como seus modelos quantizados. Para elucidar os modelos, mostramos alguns exemplos simples e outros complexos de seus funcionamentos na prática.

Finalmente no Capítulo (5) propomos modelos de dinâmicas com esses neurônios quânticos artificiais. Após a descrição analítica de cada modelo, um conjunto de experimentos é realizado e depois uma análise deles é elaborada. No final do capítulo, propomos duas medidas comparativas de modelos de dinâmica δ e δ_{abs} de forma que os modelos possam ser avaliados e comparados quantitativamente. A conclusão e considerações finais são feitas no Capítulo (6).

CAPÍTULO 2

Sistemas Dinâmicos

Há diversas ferramentas e conceitos dentro da área de Sistemas Dinâmicos que ajudam o projetista a entender o comportamento de seu sistema. Eles serão apresentados nesse capítulo para que auxilie no entendimento e na avaliação dos modelos que serão apresentados nesse trabalho.

2.1 Orbitais

Existem diversos problemas na ciência e na matemática que envolvem iteração. Iteração significa repetir um processo várias vezes. Em dinâmica, este processo que é repetido é a aplicação de uma função. O resultado da aplicação de uma função em um tempo exatamente anterior é usado como entrada nessa mesma função no tempo atual.

Dado $x_0 \in \mathbf{R}$, define-se o orbital para x_0 de uma função F a sequência de pontos $x_0, x_1 = F(x_0), x_2 = F^2(x_0), \dots, x_n = F^n(x_0), \dots$. O ponto x_0 é chamado de "semente" do orbital.

Quando uma função possui uma constante, é normal representá-la como $F_c(z)$ para c uma constante. Como exemplo, temos $F_c(z) = z^2 + c$, onde $F_2(z) = z^2 + 2$, para $c = 2$.

Tipos de orbitais

Muitos tipos de orbitais podem ser encontrados em sistemas dinâmicos. O mais importante tipo de orbital é o *fixed point*, ponto fixo. O ponto fixo é o ponto x_0 que satisfaz $F(x_0) = x_0$. Nota-se que $F^2(x_0) = F(F(x_0)) = F(x_0)$ e, de forma geral, $F^n(x_0) = x_0$. O orbital de um ponto fixo é uma sequência constante x_0, x_0, x_0, \dots . Pontos fixos podem ser encontrados geometricamente pela análise da intersecção do gráfico de uma função com a linha diagonal:

$$Q_c(z) = z \tag{2.1}$$

Um outro tipo importante de orbital é o periódico ou cíclico. O ponto x_0 é periódico se $F^n(x_0) = x_0$ para algum $n > 0$. O menor n é chamado de *prime period*, período principal. Nota-se que se x_0 é periódico com o período principal n , então o orbital de x_0 é apenas uma sequência repetida de números: $x_0, F(x_0), \dots, F^{(n-1)}(x_0), x_0, F(x_0), \dots, F^{(n-1)}(x_0), \dots$

Um ponto x_0 é chamado de *eventually fixed*, eventualmente fixo, se x_0 não é um ponto fixo ou periódico, mas algum ponto no seu orbital de x_0 é fixo ou periódico. Por exemplo, -1 é eventualmente fixo para $F(x) = x^2$, pois $F(-1) = 1$, que é fixo.

Quando um orbital está longe de ser fixo, podendo iteragir sobre números randômicos, é bem provável que ele tenha *comportamento caótico*.

Atração ou Repulsão

Existem principalmente dois tipos de pontos fixos, os que atraem e os que repelem. Um ponto fixo x_0 de uma função F é um ponto fixo atrator se $|F'(x_0)| < 1$. O ponto x_0 é um ponto repelente se $|F'(x_0)| > 1$. Finalmente, se $|F'(x_0)| = 1$, o ponto fixo é chamado de neutro ou indiferente. Para a função, por exemplo, $F(x) = x^2$, que possui os pontos fixos 0 e 1, o que acontece com os orbitais na vizinhança desses valores é que se $|x_0| < 1$, então o orbital de x_0 rapidamente se aproxima de zero. Por exemplo, o orbital de 0.1 é 0.1, 0.01, 0.0001, 0.00000001, De fato, se algum x_0 com $0 \leq x_0 < 1$, não importa tão próximo de 1, o orbital sempre tende para longe de 1 e para perto de 0. O orbital de 0.9 é 0.9, 0.81, 0.6561, 0.430467..., 0.0117, Mais precisamente, se $0 \leq x_0 < 1$ então $F^n(x_0) \rightarrow 0$, com $n \rightarrow \infty$. Por outro lado, se $x_0 > 1$, então o orbital vai para longe de 1. Por exemplo 1.1, 1.21, 1.4641..., ..., 445.79, Então, se $x_0 > 1$, nós temos que $F^n(x_0) \rightarrow \infty$, quando $n \rightarrow \infty$. Ou seja, pontos próximos de 0 são atraídos para zero e pontos próximos de 1 tem suas órbitas repelidas, que se afastam de 1.

2.2 Bifurcações

O objetivo do entendimento sobre bifurcações está aliado à necessidade de como a dinâmica de uma função Q_c muda quando seu parâmetro c varia. Para isso, num primeiro momento, deseja-se encontrar os pontos fixos de Q_c . Eles são obtidos, como dito na Equação (2.1), igualando $Q_c(x) = x$ e encontrando os valores de x que satisfazem essa equação.

Para $Q_c(x) = x^2 + c$, temos $x^2 + c = x$, em que $x_1 = \frac{1}{2}(1 + \sqrt{1 - 4c})$ e $x_2 = \frac{1}{2}(1 - \sqrt{1 - 4c})$ são as duas raízes que satisfazem essa equação. Existem, então, dois pontos fixos de Q_c . Considerando ainda essa mesma função como exemplo, podemos verificar que x_1 e x_2 são reais se e somente se $1 - 4c \geq 0$, ou $c \leq 1/4$. Quando $c > 1/4$, Q_c não tem pontos fixos. Quando $c = 1/4$, tem-se que $x_1 = x_2 = 1/2$. Finalmente, quando $c < 1/4$, então x_1 e x_2 são reais e distintos. Nota-se que se tem $x_1 > x_2$ nesse caso.

Retornando o caso de $c > 1/4$, verifica-se graficamente que Q_c é uma parábola voltada para cima. Quando $c > 1/4$, esse gráfico não se encontra com a linha diagonal $y = x$. Essa análise de gráfico mostra que todos os orbitais de Q_c quando $c > 1/4$, tende para infinito.

Quando c decrementa de $1/4$, essa situação muda e acontece a primeira bifurcação. Bifurcação, então, significa a divisão em dois, uma separação, que é exatamente o que acontece com os pontos fixos de Q_c . Para $c > 1/4$, Q_c não tem ponto fixo. Para $c = 1/4$, Q_c tem exatamente um ponto fixo; mas para $c < 1/4$, esse ponto fixo se divide em dois, um em x_1 e outro em x_2 .

Como explicado na Seção (2.1), podemos verificar se um tal ponto fixo é atrator, repelente ou neutro. Para $Q_c(x) = x^2$, $Q'_c(x) = 2x$, onde $Q'_c(x_1) = 1 + \sqrt{1 - 4c}$ e $Q'_c(x_2) = 1 - \sqrt{1 - 4c}$. Nota-se que se $c = 1/4$, $Q'_c(x_1) = 1$, mas $Q'_c(x_1) > 1$ para $c < 1/4$, desde que $\sqrt{1 - 4c} > 0$, para esses valores de c . Por isso, x_1 é um ponto fixo neutro quando $c = 1/4$ mas é repelente quando $c < 1/4$.

Nessa mesma função, analisar a dinâmica para o ponto fixo x_2 é um pouco mais complicado. Se $c = 1/4$, $Q'_c(x_2) = 1$ (aqui, claro, $x_2 = x_1 = 1/2$). Quando c é um pouco abaixo de $1/4$, $Q'_c(x_2) < 1$, então x_2 se torna um atrator. Mas procurando todos os valores de c para $|Q'_c(x_2)| < 1$, vê-se que:

$$-1 < Q'_c(x_2) < 1 \quad (2.2)$$

ou

$$-1 < 1 - \sqrt{1 - 4c} < 1 \quad (2.3)$$

Resolvendo essas inequações:

$$\begin{aligned} 2 &> \sqrt{1 - 4c} > 0 \\ 4 &> 1 - 4c > 0 \\ -3/4 &< c < 1/4 \end{aligned} \quad (2.4)$$

Então, x_2 é um ponto fixo atrator para Q_c quando $-3/4 < c < 1/4$. É fácil verificar que, quando $c = -3/4$, $Q'_c(x_2) = 1 - \sqrt{1 + 3} = -1$, então x_2 é neutro e quando $c < -3/4$, $Q'_c(x_2) < -1$, então x_2 é repelente.

Dessa forma, além de analisar quem são os pontos fixos, pode-se verificar se são atratores, repelentes ou neutros e precisar onde são os pontos onde a função não tem ponto fixo, onde ela possui um ponto fixo e onde os pontos fixos se dividem. Toda essa análise tem tudo a ver com a dinâmica assintótica dos orbitais dessas funções, como foi detalhadamente descrito acima.

2.3 Diagrama do Orbital

A imagem pode ser uma das mais instrutivas, ou até mesmo a mais intrigante, ferramenta de análise de sistemas dinâmicos. O digrama do orbital permite capturar a dinâmica de uma função Q_c para diferentes valores de c , em uma só figura. O resultado dá uma boa análise da dinâmica de sua família completa assim como a idéia de como Q_c faz sua transição para o caos.

No diagrama de orbital, plota-se o parâmetro c no eixo horizontal e plotamos o orbital assintótico de 0 no eixo vertical. Por assintótico, compreende-se que não serão os valores gerados nas primeiras iterações e sim aqueles que já superaram o comportamento transiente da função. Pode-se verificar o diagrama do Orbital da função $Q_c(z) = z^2 + c$ na Figura 2.1, encontrado facilmente pelo Algoritmo (1).

Algorithm 1: Algoritmo para encontrar o Diagrama do Orbital

```

1 for  $c$  entre  $c_{min}$  e  $c_{max}$  do
2   for  $n$  entre 0 e 200 do
3     Compute  $Q_c^n(0)$ ;
4     if  $n > 50$  then
5       Desenhe no gráfico  $(c, Q_c^n(0))$ ;
6     end
7   end
8 end

```

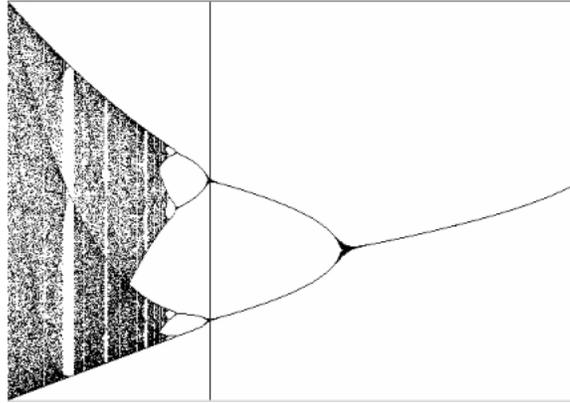


Figura 2.1: Diagrama do Orbital $Q_c(z) = z^2 + c$, onde $-2 < c < 0.25$

2.4 Conjunto Julia

Conjunto Julia é o lugar onde todo o comportamento caótico de uma função complexa ocorre [13]. Por exemplo, a função de mapa quadrático $Q_0(z) = z^2$ é caótica no círculo unitário, pois se $|z| < 1$ então $|Q_0^n(z)| \rightarrow 0$, quando $n \rightarrow +\infty$, e se $|z| > 1$ então $|Q_0^n(z)| \rightarrow +\infty$ quando $n \rightarrow +\infty$. Sendo $Q_0(z)$ a notação de orbital 0 para $Q(z)$.

Através do *filled julia set*, conjunto Julia preenchido, é possível identificar figuras fractais cuja natureza é bastante peculiar. O conjunto julia preenchido de Q_c é o conjunto de pontos cujos orbitais são limitados. O conjunto Julia de Q_c é o limite, a fronteira, do conjunto Julia preenchido. $K_0 = \{z \mid |z| \leq 1\}$ e $J_0 = \{z \mid |z| = 1\}$ são, respectivamente, Conjunto Julia Preenchido e Conjunto Julia. Um exemplo de um conjunto Julia preenchido pode ser observado na Figura (2.2).

Um algoritmo simplificado para encontrar o conjunto Julia preenchido pode ser o descrito no Algoritmo 2.

Algorithm 2: Algoritmo para encontrar uma aproximação do Conjunto Julia Preenchido

- 1 Escolha um número máximo de iterações N .
 - 2 Para cada ponto z no grid, compute os primeiros N pontos no orbital de z . Se $|Q_c^i(z)| > \max\{|c|, 2\}$, para algum $i \leq N$, então páre a iteração e pinte z de branco.
 - 3 Se $|Q_c^i(z)| \leq \max\{|c|, 2\}$, para todo $i \leq N$, então pinte z de preto.
 - 4 Os pontos pintados de branco representam o orbital que escapou, e os pontos pretos não, pelo menos nas N primeiras iterações.
 - 5 Os pontos pretos, então, serão uma aproximação do conjunto Julia preenchido.
-

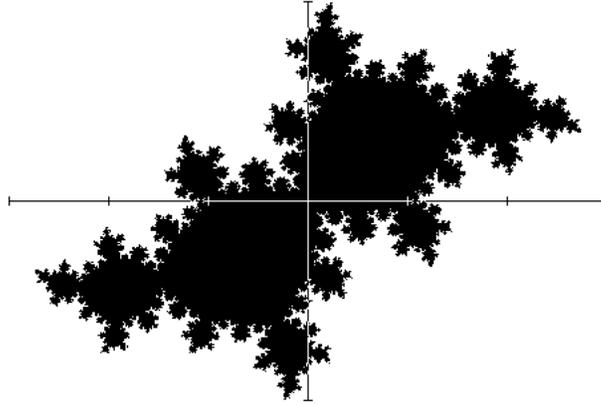


Figura 2.2: Conjunto Julia preenchido para $Q_c(z) = z^2 + c$, $c = -0.39054 - 0.58679i$

2.5 Mandelbrot

É o dicionário de todos os conjuntos Julia. O conjunto Mandelbrot M consiste de todos os valores de c tais que o conjunto Julia preenchido K_c é "conectado". Isso significa que o sistema é testado para valores de c diferentes, com orbitais de 0. Se a dinâmica desse orbital é convergente, ele pertence ao conjunto, caso contrário, ele não pertence ao conjunto. Ou seja: $M = \{c \in \mathbb{C} \mid |Q_c^n(0)| \not\rightarrow \infty\}$. A rotina para encontrar os valores de c que satisfazem o conjunto Mandelbrot é ilustrado no Algoritmo (3).

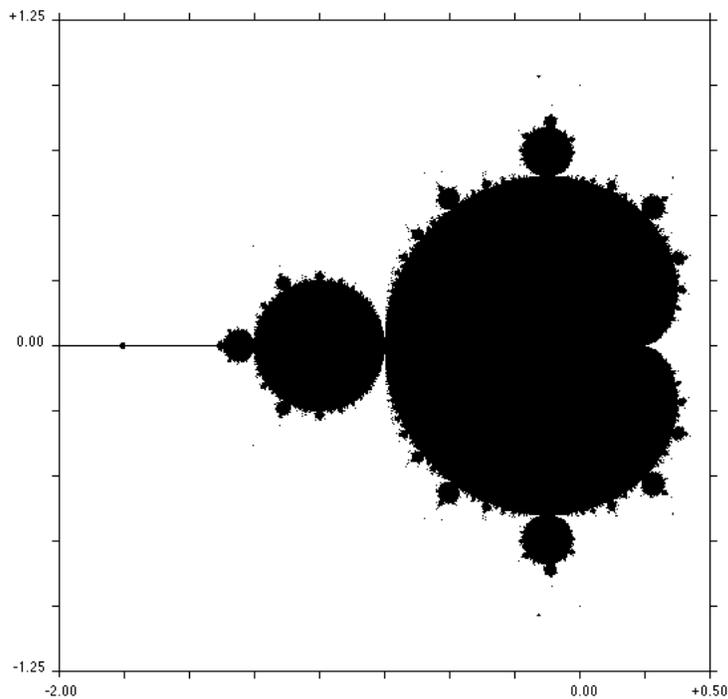


Figura 2.3: Conjunto Mandelbrot para $Q_c(z) = z^2 + c$

Algorithm 3: Algoritmo para encontrar uma aproximação do Conjunto Mandelbrot

- 1 Escolha um número máximo de iterações N .
- 2 Para cada ponto c no grid, compute os primeiros N pontos no orbital 0 de Q_c . Se $|Q_c^i(0)| > \max\{|c|, 2\}$, para algum $i \leq N$, então páre a iteração e pinte c de branco.
- 3 Se $|Q_c^i(0)| \leq \max\{|c|, 2\}$, para todo $i \leq N$, então pinte c de preto.
- 4 Os pontos pretos, então, serão uma aproximação do conjunto Mandelbrot.

É possível fazer uma associação do Conjunto Mandelbrot com Bifurcações. Na função $Q_c(z) = z^2 + c$, por exemplo, para acharmos os valores de c em que Q_c tem um ponto fixo atrator ou neutro fazemos:

$$\begin{aligned} Q_c(z) &= z^2 + c = z \\ Q'_c(z) &= |2z| \leq 1 \end{aligned} \quad (2.5)$$

A primeira equação é verdade para os valores de z que são fixos e a segunda para os atratores, como foi dito nas seções anteriores. A segunda equação também diz que z_c se encontra em cima ou dentro do disco limitado de raio $1/2$. No círculo $|Q'_c(z_c)| = 1$, então z_c é um ponto fixo neutro. Se escrevemos este círculo como $z = \frac{1}{2}e^{i\theta}$, então podemos encontrar a equação dos valores de c em que Q_c tem um ponto fixo, isolando-o da equação anterior:

$$c = \zeta(\theta) = \frac{1}{2}e^{i\theta} - \frac{1}{4}e^{2i\theta} \quad (2.6)$$

Esta função forma uma curva representada na Figura (2.4) exatamente o cardióide preto mais largo da Figura (2.3). O interior dele é formado pelos valores de c em que Q_c tem um ponto fixo atrator. Os limites dessa função no eixo real são os dois pontos $c = \frac{1}{4}$ (o ponto de bifurcação) e $c = -\frac{3}{4}$ (ponto de bifurcação), ou então, considerando $z = \frac{1}{2}e^{i\theta}$, esses pontos são $\theta = 0$ e π , respectivamente.

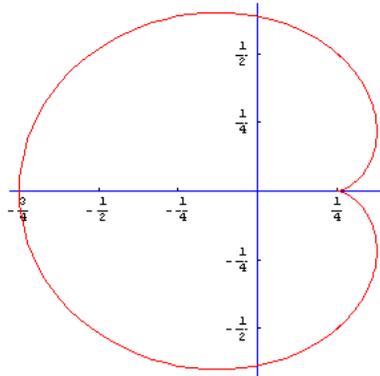


Figura 2.4: Curva $\zeta(\theta) = \frac{1}{2}e^{i\theta} - \frac{1}{4}e^{2i\theta}$ formada pelos valores de c que fazem a função Q_c ter um ponto fixo neutro

A região de atração de 2 ciclos pode ser encontrada fazendo $Q_c^2(z) = z$ e também achando suas raízes. Através deste cálculo, pode-se encontrar o círculo à esquerda do cardióide da Figura (2.3), com limites no eixo real $c = -\frac{1}{4}$ e $c = -\frac{3}{4}$.

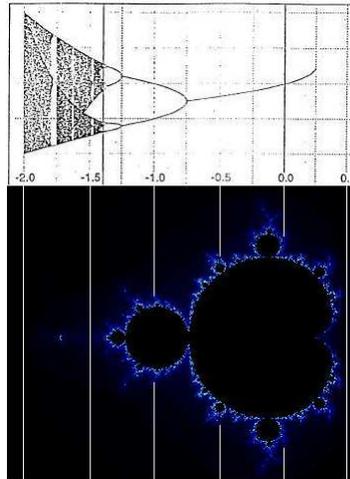


Figura 2.5: Comparação do Conjunto Mandelbrot e do Diagrama de Orbital da função $Q_c(z) = z^2 + c$. Mostra-se que o diagrama coincide com a representação do Mandelbrot de acordo com a quantidade de pontos fixos e suas bifurcações. Os pontos em que os "lóbulos" (círculos e cardioides) se encontram com o eixo real são os pontos fixos e onde os sistemas se bifurcam.

Computação Quântica

3.1 Bit Quântico

Um bit quântico é um vetor unitário bidimensional. O vetor $|0\rangle = [1, 0]^T$ e $|1\rangle = [0, 1]^T$ formam sua base computacional. Qualquer qubit $|\psi\rangle$ pode ser escrito como a equação (3.1), onde α e β são números complexos e $|\alpha|^2 + |\beta|^2 = 1$. Produto tensorial são usados para compor sistemas $|ij\rangle = |i\rangle \otimes |j\rangle$.

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \quad (3.1)$$

3.2 Operador Quântico

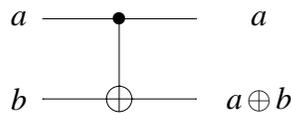
Operador quântico \mathbf{U} sobre n qubits é uma matriz unitária complexa $2^n \times 2^n$. Por exemplo, alguns operadores sobre 1 qubit são Identidade \mathbf{I} , NOT \mathbf{X} e Hadamard \mathbf{H} , descritos na Equação (3.2) e Equação (3.3). A representação combinacional desses operadores unitários formam o chamado circuito quântico.

$$\mathbf{I} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad \begin{array}{l} \mathbf{I}|0\rangle = |0\rangle \\ \mathbf{I}|1\rangle = |1\rangle \end{array} \quad \mathbf{X} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \quad \begin{array}{l} \mathbf{X}|0\rangle = |1\rangle \\ \mathbf{X}|1\rangle = |0\rangle \end{array} \quad (3.2)$$

$$\mathbf{H} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad \begin{array}{l} \mathbf{H}|0\rangle = 1/\sqrt{2}(|0\rangle + |1\rangle) \\ \mathbf{H}|1\rangle = 1/\sqrt{2}(|0\rangle - |1\rangle) \end{array} \quad (3.3)$$

O operador identidade \mathbf{I} gera a saída exatamente como a entrada; Operador \mathbf{X} funciona como o NOT clássico na base computacional; Hadamard \mathbf{H} gera superposição de estados. O operador CNOT tem 2 entradas e 2 saídas e inverte o segundo qubit se o primeiro é 1, como mostrado na Figura 3.1.

Figura 3.1: Operador CNOT



Os operadores da computação quântica podem ser enxergados como transformadores lineares, ou seja, matrizes que operam sobre uma base vetorial. Essas matrizes necessitam ser unitárias e inversíveis e tem o formato genérico:

$$U(\alpha, \beta, \gamma, \delta) = e^{i\alpha} \begin{bmatrix} e^{-i\beta/2} & 0 \\ 0 & e^{i\beta/2} \end{bmatrix}^* \begin{bmatrix} \cos(\gamma/2) & -\sin(\gamma/2) \\ \sin(\gamma/2) & \cos(\gamma/2) \end{bmatrix}^* \begin{bmatrix} e^{-i\delta/2} & 0 \\ 0 & e^{i\delta/2} \end{bmatrix} \quad (3.4)$$

onde $\alpha, \beta, \gamma, \delta$ são números reais. A segunda matriz é uma rotação ordinária, a primeira e última matrizes podem ser visualizadas como rotações em planos diferentes. Através dessa decomposição, pode-se construir de forma exata qualquer porta quântica de um qubit [8].

3.3 Paralelismo

Paralelismo é uma importante característica da computação quântica. Se alguém tem um operador \mathbf{U}_f sobre $n + 1$ qubits de forma que $\mathbf{U}_f|x_i, 0\rangle = |x_i, f(x_i)\rangle$ para cada x_i na base computacional, então se pode calcular todos os valores simultaneamente aplicando \mathbf{U}_f no estado descrito em (3.5), que resultará no estado descrito na Equação (3.6).

$$|\psi\rangle = \sum_{i=0}^{n-1} \alpha_i |x_i, 0\rangle \quad (3.5)$$

$$\mathbf{U}_f|\psi\rangle = \sum_{i=0}^{n-1} \alpha_i |x_i, f(x_i)\rangle \quad (3.6)$$

Neurônios Artificiais

4.1 Neurônio Clássico

Um Perceptron Clássico é uma função matemática para representar uma operação inspirada por neurônios biológicos. O resultado de um Perceptron simples é 1, ou HIGH, se o produto interno entre o vetor de entrada e o vetor de peso é maior ou igual que um valor de limiar, e 0, ou LOW, caso contrário.

O Perceptron simples tem a função degrau como função de ativação. Outros modelos, como Adaline, incorporaram funções lineares e incrementaram seu poder computacional, como descrito em (4.1). Ele é usado em problemas de classificação e como aproximadores de função.

$$y = f\left(\sum_{i=0} x_i * w_i\right) \quad (4.1)$$

Uma Rede Neural é um conjunto de perceptrons. As saídas de cada perceptron são entradas para outros Perceptrons. Uma rede neural pode ter mais de uma camada de perceptrons, incrementando seu poder computacional e resolvendo problemas não lineares. Atualmente, as redes neurais podem ser usadas como previsão para séries temporais, problemas de classificação, reconhecimento de padrões e muitos outros problemas reais.

4.2 Redes Neurais Clássicas Sem Peso

As Redes Neurais Sem Peso (RNSP), que foram propostas inicialmente por Igor Aleksander [19], são modelos de computação neural que possuem entrada e saída binárias e não há pesos entre seus nós. Funcionam como look-up tables, tabelas de busca, e seu aprendizado, que pode ser supervisionado ou não-supervisionado, é a mudança do conteúdo endereçado por uma entrada. Há um rápido processamento do aprendizado principalmente devido a capacidade de não modificar o conteúdo anteriormente aprendido - nos modelos com peso, o treinamento de uma dada entrada-saída pode modificar o valor aprendido e armazenado nos pesos da entrada-saída anteriormente apresentada. Por ser flexível, de fácil implementação paralela e a possibilidade de implementação comercialmente acessível com as memórias de acesso aleatório, RAMs, as RNSP foram bastante investigadas e resolvem muitos problemas por causa de sua generalização e identificação de padrões [18].

As RNSP podem ser identificadas erroneamente como Redes Neurais Booleanas (RNB). Uma RNB é um modelo de rede neural em que entradas e saídas são booleanas mas ainda existem pesos entre seus nós que são adaptados de acordo com seu treinamento. Tais pesos

podem ser sempre booleanos, transformando-se em uma RNSP, porém, os pesos podem não ser booleanos, como a rede de Hopfield [20]. Então, uma RNSP é uma RNB com pesos booleanos e uma RNB não é uma RNSP.

Nesta seção, iremos apresentar os principais modelos propostos na literatura de Redes Neurais Sem Peso. Uma análise detalhada de outros modelos pode ser encontrada em [18].

4.2.1 Nó RAM

Um nó RAM não foi concebido inicialmente sob inspiração biológica, pois seu funcionamento não envolve uma explicação sob ponto de vista de bio-comportamento de sinapses e propagação de sinal neural e elétrico. Sua inspiração veio da concepção do uso facilitado de memórias RAMs pela engenharia para resolver problemas de reconhecimento de padrões.

O funcionamento de um nó RAM envolve então um acesso às posições de memória de uma RAM, onde o endereçamento é feito pela entrada. N bits de entrada são capazes de endereçar, então, 2^N posições de memória. Dado um sinal $x = x_1x_2x_3\dots x_N$, a saída do nó RAM é o bit $y = C[x]$, 0 ou 1, armazenado na posição de memória x . Tal comportamento pode então ser visto como o mapeamento de uma função lógica, onde o nó RAM computa qualquer função booleana de suas entradas.

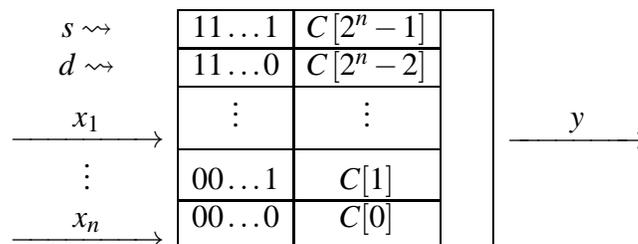


Figura 4.1: A figura mostra uma representação do nó RAM. Ao apresentar uma entrada x , o nó RAM devolve como saída o bit y armazenado no endereço $C[x]$.

O aprendizado do nó RAM é entendido como o processo que atualiza o bit na posição de memória de cada entrada de treinamento. O nó RAM então pode computar todas as funções binárias de suas entradas enquanto o nó com peso pode computar apenas funções linearmente separáveis. Nesse modelo de nó RAM, não é possível haver generalização, ou seja, não há capacidade de extrapolar o que foi apreendido na etapa de aprendizado para padrões que não foram apresentados durante essa fase. O nó RAM básico produz as saídas corretas apenas para os padrões vistos na etapa de treinamento. Porém, há outros modelos de redes sem peso que são capaz de generalizar e são compostos por nós RAMs que consideram distância *hamming* de padrões ou máscaras [21].

4.2.2 Nó Lógico Probabilístico (Probabilistic Logic Node) PLN

Um dos problemas identificados no uso do nó RAM é que o Nó, ao dar como saída a resposta 0, não se era possível identificar se aquele padrão identificado pela RAM era realmente zero ou se

aquela posição de memória não havia sido modificada durante a fase de treinamento, já que todas as posições de memória eram zeradas antes que o nó pudesse aprender. Em outras palavras, encontrar um valor de saída zero significava ou que o vetor de entrada é um contraexemplo da classe correspondente ou que nenhuma informação foi apresentada sobre aquele padrão durante o treinamento. Foi a partir dessa limitação que veio a inspiração para a construção do nó PLN, proposto por Kan e Aleksander [22].

Para solucionar esse problema, foi proposto que os possíveis valores armazenados na memória seriam 0, 1 e u . Ao acessar uma posição de memória que armazenasse o valor u , um gerador probabilístico faria o nó PLN dar como saída um número aleatório numa distribuição uniforme $U \sim (0, 1)$, conforme a Equação (4.2). O valor u (undefined) torna possível identificar os estados que não foram atualizados na etapa de treinamento.

$$y = \begin{cases} 0, & \text{if } C[x] = 0 \\ 1, & \text{if } C[x] = 1 \\ U(0, 1), & \text{if } C[x] = u \end{cases} \quad (4.2)$$

Antes do desenvolvimento deste modelo, só existiam modelos neurais sem peso de uma única camada, formada por vários nós, com uma unidade de Discriminador, que fazia a ponderação (soma) das saídas de cada nó. Para resolver um determinado problema de classificação, então, cada classe era treinada em um discriminador e em uma camada de nós com seus exemplos. Ou seja, quando se desejasse encontrar a classe de um padrão que se deseja saber em qual classe ele pertence, um exemplo não rotulado, apresentaria tal padrão para cada um dos discriminadores e o classificaria para aquela classe cujo valor do discriminador fosse maior. Com o surgimento do PLN, Aleksander [23] propôs uma arquitetura multicamadas, chamada de pirâmides. As pirâmides são uma estrutura tripla em que cada camada geralmente tem um número fixo de neurônios com fan-out 1 e baixo fan-in. Pirâmides tem menos funcionalidade que um único neurônio armazenando o mesmo número de bits de entrada. Porém, se comparada com o único nó, a pirâmide reduz o tamanho de memória necessária e aumenta a generalização [24].

4.2.3 MPLN - PLN multivalorado (Multi-valued PLN)

O PLN foi sofisticado em [25] e [26], onde apareceu a definição do MPLN ou m-state PLN. O diferencial do MLPN é que armazena em uma posição de memória um range de probabilidade maior mas ainda assim discreto, e a saída do MLPN pode ser, além de uma função de distribuição de probabilidade, uma função linear ou sigmoïdal [26].

Ou seja, um nó MLPN, por exemplo, pode aprender a dar como saída 1 com probabilidade 15% quando uma determinada posição de memória for acessada.

4.2.4 pRAM - Probabilistic RAM

Em [27], Taylor propôs um modelo de neurônios ruidosos que apresentam comportamento característico e incorpora muitas propriedades conhecidas de neurônios reais. A evolução de tais modelos foi mostrada ser equivalente às redes neurais de ruído pRAM [28]. O pRAM é uma extensão do modelo do PLN, assim como o MLPN, porém armazena probabilidades contínuas, entre $[0,1]$, nas posições de memória. A entrada x endereça o valor da frequência

$y = C[x]$ poder dar como saída o valor 1. Vale-se salientar que esta frequência/probabilidade é contínua no pRAM.

Dada uma entrada com N bits no pRAM, como já foi visto anteriormente nos outros modelos, tem-se 2^N posições de memórias endereçáveis, porém, no pRAM armazena a probabilidade contínua que o bit de saída seja 1. Ou seja, a média dos valores de saída para uma dada entrada I é conhecida, que é $y = C[I]$.

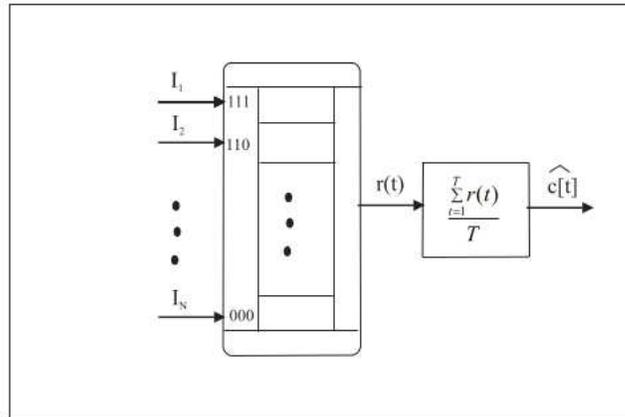


Figura 4.2: Nó pRAM. A posição de memória $C[x]$, para uma dada entrada x , armazenará um valor contínuo que será a probabilidade da saída ser 1.

O pRAM foi estendido para converter entradas contínuas em saídas binárias. Tal extensão é conhecida como integrando pRAM (i-p-RAM) [29]. Em resumo, esse mapeamento de entrada contínua em saída binária é feita considerando a entrada contínua como um array $x = x_1, x_2, \dots, x_N$, onde $x \in [0, 1]^N$. Neste caso, cada valor x_i ($i=1, \dots, N$) pode ser representado por uma sequência de pulsos I_j tais que x_j define a probabilidade que 1 ocorra na j -ésima posição de I , $P(I_j = 1) = x_j$. Ou seja, ao contrário do pRAM ou qualquer outro nó sem peso, no i-p-RAM não acessa a apenas um valor de entrada na tabela de memória, para que se faça a associação da probabilidade de saída. Mas, a cada $t = 1 \dots T$ passos de tempo, uma rajada de posições será acessada e contribuirá estocasticamente com o valor de saída $r=1$ ou $r=0$. Mais detalhes podem ser encontrados em [18].

4.3 Neurônio Quântico Sem Peso

Existem diversos modelos de neurônios quânticos existentes na literatura que abordam estratégias intrinsecamente quânticas ou apenas de inspiração quântica. O não-parallelismo nos modelos ou a utilização de operadores não-unitários e não-lineares fazem com que, ao serem projetados, devam ser analisados cuidadosamente quanto a sua viabilidade física de implementação. Nós propomos uma análise desses modelos e pode ser encontrada em [9].

Nesta seção, apresentamos modelos de neurônios quânticos que foram inspirados nas redes sem peso, apresentadas na seção anterior.

4.3.1 qRAM

De Oliveira et al. em [30] e [31] definiram uma quantização do neurônio RAM descrito em [4] através de uma técnica chamada *Mathematical Quantization* [41]. Ela significa que "A *idéia fundamental da quantização matemática é que conjuntos sejam substituídos por espaços de Hilbert*". Os elementos de conjuntos são colocados um a um em relação com as bases do Espaço de Hilbert.

O nó RAM armazena em uma posição um único bit. A descrição inicial do nó qRAM mostra que a matriz A representa o comportamento de um RAM clássico com, sem perda de generalidade, uma entrada e uma saída:

Weaver:01

$$A = \begin{pmatrix} I & 0 \\ 0 & X \end{pmatrix} \quad \text{where} \quad \begin{array}{l} A|00\rangle = |0\rangle I|0\rangle \\ A|10\rangle = |1\rangle X|0\rangle \end{array} \quad (4.3)$$

Esta matriz comporta-se como uma célula capaz de retornar 0 ou 1 dependendo do valor do primeiro bit. Uma RAM de n -bits possui uma coleção de 2^n desses A 's, conseqüentemente 2^n seletores e um qubit de saída $|o\rangle = |0\rangle$. Uma representação em forma de circuito é feita na Figura (4.3). As matrizes A 's estão implícitas na notação do circuito, que escolhe o operador U_0 se a entrada $|ab\rangle$ for $|00\rangle$ ou escolhe U_1 , se a entrada for $|ab\rangle = |01\rangle$, e assim conseqüentemente. Essa notação é descrita pelo autor como uma q-ROM, já que ela aparece fixa no circuito e não pode ser alterada.

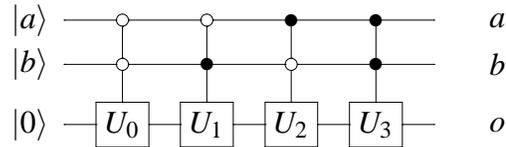


Figura 4.3: Nó q-RAM em descrição de circuito quântico, onde $U_i, i = 0, 1, \dots, 2^n - 1$, pode ser os operadores I ou X

Treinar uma ROM representada pela figura (4.3), é escolher qual a matriz U_i em cada uma das i posições. Essa RAM é estendida para uma nova versão que inclui seletores, mostrada na Figura (4.4). Dessa forma, o treinamento acontece modificando os valores dos seletores. Da Silva et al. propõem um aprendizado clássico e quântico em [6] para o qRAM.

Uma rede neural com nós q-RAM é representado na Figura (4.5).

O neurônio qRAM pode ser visto do ponto de vista completamente matricial através de um

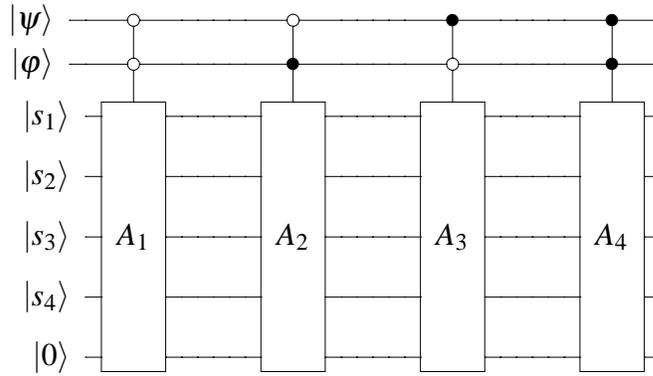


Figura 4.4: qRAM node

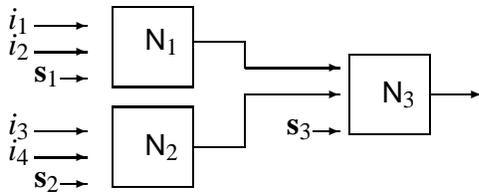


Figura 4.5: Rede qRAM de 2 camadas

operador unitário:

$$qRAM = \begin{pmatrix} \mathbf{I} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \mathbf{I} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \mathbf{X} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathbf{X} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathbf{I} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \mathbf{X} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \mathbf{I} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mathbf{X} \end{pmatrix} \quad (4.4)$$

O funcionamento do nó qRAM é entendido como a escolha de um seletor por parte do qubit de entrada. Tal seletor será o qubit de controle da operação CNOT em que o qubit de saída é o qubit alvo. Por exemplo, dada a entrada $|i\rangle=|0\rangle$, o neurônio irá escolher o primeiro seletor, $|s_0\rangle$, para ser aplicado à porta CNOT junto com o qubit de saída; dada a entrada $|i\rangle=|1\rangle$, o neurônio irá escolher o segundo seletor, $|s_1\rangle$, para ser aplicado com o qubit de saída à porta CNOT, e assim por diante. Em uma entrada em superposição, o neurônio pode escolher seletores em superposição e, provavelmente, gerar saídas em superposição. O neurônio N é representado

pelo operador na equação (4.5).

$$N = \sum_{i=0}^{2^n-1} |i\rangle_n \langle i|_n A_{s_i} o \quad (4.5)$$

Exemplo 1 do qRAM

Dada uma qRAM com o qubit de entrada $|\psi\rangle = |0\rangle$, seletores $|s_0\rangle = |1\rangle$ e $|s_1\rangle = \frac{1}{2}|0\rangle + \frac{1}{2}|1\rangle$, e registrador de saída, por definição, $|o\rangle = |0\rangle$.

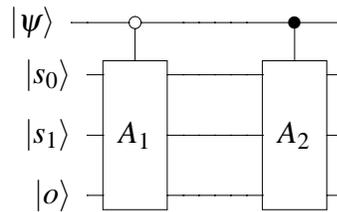


Figura 4.6: Nó qRAM do Exemplo 1

Por seu funcionamento, como a entrada está na base computacional, ela irá escolher apenas um seletor. Como o seu valor é $|0\rangle$, então irá escolher o seletor s_0 para aplicar o operador CNOT. Como o valor no seletor escolhido é $|1\rangle$, ele irá aplicar o operador X no qubit $|0\rangle$:

$$CNOT|1\rangle|0\rangle = |1\rangle|1\rangle \quad (4.6)$$

Como operador, a operação do qRAM pode ser visto como:

$$\begin{aligned} qRAM|0\rangle|1\rangle & \left(\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle \right) |0\rangle \\ & = |0\rangle|1\rangle \left(\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle \right) |1\rangle \\ & = \frac{1}{\sqrt{2}} (|0\rangle|1\rangle|0\rangle|1\rangle + |0\rangle|1\rangle|1\rangle|1\rangle) \end{aligned}$$

Exemplo 2 do qRAM

Dada uma qRAM com o qubit de entrada $|\psi\rangle = \frac{1}{2}|0\rangle - \frac{1}{2}|1\rangle$, seletores $|s_0\rangle = -\frac{1}{2}|0\rangle - \frac{1}{2}|1\rangle$ e $|s_1\rangle = |0\rangle$, e registrador de saída, por definição, $|o\rangle = |0\rangle$. Algebricamente, como operador, a

operação do qRAM sobre esses qubits pode ser vista como:

$$\begin{aligned}
qRAM(|\psi\rangle|s_0\rangle|s_1\rangle|0\rangle) &= \\
&= qRAM\left(\left(\frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle\right) \otimes \left(-\frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle\right) \otimes |0\rangle \otimes |0\rangle\right) \\
&= qRAM\left(\frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle\right) \otimes \left(\left(-\frac{1}{\sqrt{2}}|0\rangle \otimes |0\rangle \otimes |0\rangle\right) + \left(-\frac{1}{\sqrt{2}}|1\rangle\right) \otimes |0\rangle \otimes |0\rangle\right) \\
&= qRAM\left(-\frac{1}{2}|0\rangle|0\rangle|0\rangle|0\rangle - \frac{1}{2}|0\rangle|1\rangle|0\rangle|0\rangle + \frac{1}{2}|1\rangle|0\rangle|0\rangle|0\rangle + \frac{1}{2}|1\rangle|1\rangle|0\rangle|0\rangle\right) \\
&= -\frac{1}{2}|0\rangle|0\rangle|0\rangle|0\rangle - \frac{1}{2}|0\rangle|1\rangle|0\rangle|1\rangle + \frac{1}{2}|1\rangle|0\rangle|0\rangle|0\rangle + \frac{1}{2}|1\rangle|1\rangle|0\rangle|0\rangle
\end{aligned}$$

4.3.2 qPLN

De Oliveira et al. apontam que se a matriz U_i , na figura (4.3) puder ser também um operador Hadamard, H , essa ROM pode então simular uma PLN, já que a saída antes da medição pode ser $|0\rangle$, $|1\rangle$, ou um estado em superposição com amplitudes equiprováveis em ambos os estados $|0\rangle$ e $|1\rangle$.

A propriedade randômica do PLN é garantida pela implementação da porta $H|0\rangle = |u\rangle$, por princípios quânticos [8].

O comportamento do modelo q-PLN em circuito pode ser descrito pela matriz A abaixo:

$$A = \begin{pmatrix} I & 0 & 0 & 0 \\ 0 & X & 0 & 0 \\ 0 & 0 & H & 0 \\ 0 & 0 & 0 & U \end{pmatrix} \quad \text{onde} \quad \begin{aligned} A|000\rangle &= |00\rangle I|0\rangle \\ A|010\rangle &= |01\rangle X|0\rangle \\ A|100\rangle &= |10\rangle H|0\rangle \\ A|110\rangle &= |11\rangle U|0\rangle \end{aligned} \quad (4.7)$$

Em forma de circuito quântico significa que para N entradas, o q-PLN terá $2 * 2^n$ seletores e um qubit de saída $|o\rangle = |0\rangle$, conforme demonstra a Equação (4.7). Se os seletores estiverem na base computacional, seu comportamento é de aplicar no qubit de saída o operador Identidade (se forem $|s\rangle = |00\rangle$), aplicarem o operador X (se forem $|s\rangle = |01\rangle$), aplicarem o operador Hadamard (se forem $|s\rangle = |10\rangle$), ou, por fim, aplicarem o operador unitário U (se forem $|s\rangle = |11\rangle$). O poder computacional quântico parece se sobrepôr ao clássico quando tais seletores estão em superposição de estados, ativando mais de uma operação e fazendo processamento paralelo. Um algoritmo de treinamento para o q-PLN é proposto em [30].

O neurônio trabalha com um conjunto de $2 * 2^n$ seletores, $|s\rangle$, onde n é um número de qubits de entradas $|i\rangle$, um qubit $|o\rangle$ de saída e 2^n matrizes A .

Exemplo 1 do qPLN

Dado um nó qPLN com o qubit de entrada $|\psi\rangle = |0\rangle$, seletores $|s_1^0\rangle = |1\rangle$ e $|s_2^0\rangle = |0\rangle$, $|s_1^1\rangle = |0\rangle$ e $|s_2^1\rangle = |1\rangle$ e registrador de saída, por definição, $|o\rangle = |0\rangle$.

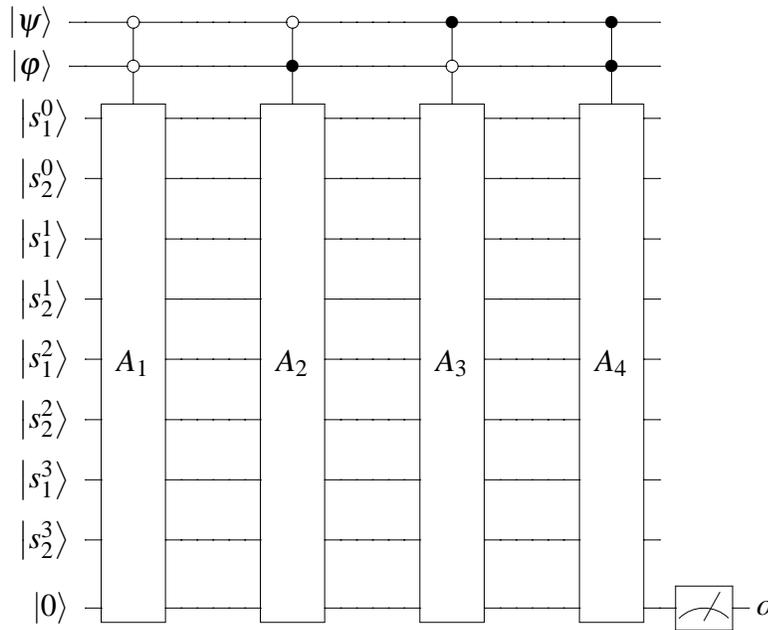


Figura 4.7: Nó qPLN

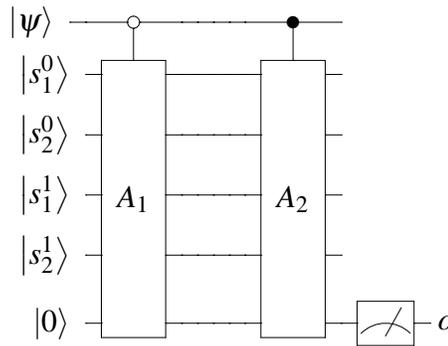


Figura 4.8: Nó qPLN do Exemplo 1

Como a entrada está na base computacional, o nó irá escolher o primeiro par de seletores, já que a entrada é $|0\rangle$. Pela operação descrita na Equação (4.7), como o valor dos seletores escolhidos é $|10\rangle$, a operação a ser aplicada na saída é H , Hadamard:

$$A|1\rangle|0\rangle|0\rangle = |1\rangle|0\rangle \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \quad (4.8)$$

A operação do qPLN pode ser vista algebricamente como:

$$\begin{aligned}
& qPLN(|0\rangle|1\rangle|0\rangle|0\rangle|1\rangle|0\rangle) \\
&= |0\rangle|1\rangle|0\rangle|0\rangle|1\rangle \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \\
&= \frac{1}{\sqrt{2}}(|0\rangle|1\rangle|0\rangle|0\rangle|1\rangle|0\rangle) + \frac{1}{\sqrt{2}}(|0\rangle|1\rangle|0\rangle|0\rangle|1\rangle|1\rangle)
\end{aligned}$$

Exemplo 2 do qPLN

Dada um nó qPLN com o qubit de entrada $|\psi\rangle = \frac{1}{2}|0\rangle - \frac{1}{2}|1\rangle$, seletores $|s_1^0\rangle = |0\rangle$ e $|s_2^0\rangle = |1\rangle$, $|s_1^1\rangle = |1\rangle$ e $|s_2^1\rangle = |1\rangle$ e registrador de saída, por definição, $|o\rangle = |0\rangle$. Algebricamente, como operador, a operação do qPLN sobre esses qubits pode ser vista como:

$$\begin{aligned}
& qPLN(|\psi\rangle|s_1^0\rangle|s_2^0\rangle|s_1^1\rangle|s_2^1\rangle|0\rangle) = \\
&= qPLN\left(\frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle\right)|01\rangle|11\rangle|0\rangle \\
&= qPLN\left(\frac{1}{\sqrt{2}}(|0\rangle|01\rangle|11\rangle|0\rangle - |1\rangle|01\rangle|11\rangle|0\rangle)\right) \\
&= \frac{1}{\sqrt{2}}(|0\rangle|01\rangle|11\rangle H|0\rangle - |1\rangle|01\rangle|11\rangle U|0\rangle) \\
&= \frac{1}{\sqrt{2}}(|0\rangle|01\rangle|11\rangle \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) - |1\rangle|01\rangle|11\rangle U|0\rangle) \\
&= \frac{1}{2}(|0\rangle|01\rangle|11\rangle|0\rangle + |0\rangle|01\rangle|11\rangle|1\rangle) + \frac{1}{\sqrt{2}}|1\rangle|01\rangle|11\rangle U|0\rangle
\end{aligned}$$

4.3.3 Quantum Multi-valued PLN: qMPLN

A diferença do MPLN e PLN clássicos é que o primeiro tem a capacidade de armazenar não somente 0, 1 e u , mas pode armazenar um valor de uma probabilidade de o bit de saída ser 1.

O modelo quantizado q-MPLN é apresentado em [30] e é uma extensão do modelo do q-PLN. Ao invés da matriz A ser formada pelos operadores I , X , H , ela é formada por matrizes U_p definidas como:

$$U_p = \begin{pmatrix} \sqrt{1-p} & -\sqrt{p} \\ \sqrt{p} & \sqrt{1-p} \end{pmatrix} \quad (4.9)$$

onde p é o valor de probabilidade armazenado naquela posição de memória. Com esta notação, $U_0|0\rangle = |0\rangle$, $U_1|0\rangle = |1\rangle$ e $U_{\frac{1}{2}}|0\rangle = |u\rangle$ que são os correspondentes ao modelo do q-PLN. O modelo geral q-MLPN tem p variando no domínio complexo entre $[0,1]$. Por exemplo, se a probabilidade de se obter o valor 1 como saída for $p = 0.25$, então:

$$U_{0.25} = \begin{pmatrix} \sqrt{0.75} & -\sqrt{0.25} \\ \sqrt{0.25} & \sqrt{0.75} \end{pmatrix} \quad (4.10)$$

que retornará $|1\rangle$ com 0.25 de probabilidade, como esperado.

A figura representativa do q-MPLN é a mesma do qPLN (Figura (4.7)), considerando apenas a matriz universal A do q-MPLN na forma abaixo:

$$A = \begin{pmatrix} \mathbf{U}_{p1} & 0 & 0 & 0 \\ 0 & \mathbf{U}_{p2} & 0 & 0 \\ 0 & 0 & \mathbf{U}_{p3} & 0 \\ 0 & 0 & 0 & \mathbf{U}_{p4} \end{pmatrix} \quad (4.11)$$

4.3.4 $|\psi\rangle$ - RAM

Em [32], da Silva et al. preocuparam-se em generalizar o qPLN para que ele pudesse implementar o conjunto universal de operadores quânticos e, assim, se tornasse um Operador Quântico Universal. Foi provado em [32] que o nó qPLN com entradas fixas, ou seja, com o terceiro registrador sendo sempre $|0\rangle$ não podiam implementar o operador Hadamard e que o teorema de clonagem de qubits era violado. Tal teorema fala da inviabilidade teórica de circuitos quânticos clonarem estados quânticos arbitrários. Essa prova se dá considerando que é possível clonar um estado de entrada se o qubit $|o\rangle$ for sempre considerado $|0\rangle$. Logo, a diferença fundamental para o qPLN é que o $|\psi\rangle$ -RAM permite colocar qualquer valor no terceiro registrador de seu nó.

Então, o nó $|\psi\rangle$ -RAM foi definido pela Equação (4.12), como possuindo uma entrada $|i\rangle$ de n qubits, seletores $|s\rangle$ com $2 * 2^n$ qubits e a saída $|o\rangle$ com 1 qubit. O estado $|i\rangle$ descreve a entrada do nó $|\psi\rangle$ -RAM e o estado $|s\rangle|o\rangle$ descreve o estado quântico do nó. A saída pode ser inicializada com qualquer valor ou usada como um registrador de entrada.

$$\mathbf{N} = \sum_{i=0}^{2^n-1} |i\rangle_n \langle i|_n A_{s_i,o} \quad (4.12)$$

Dessa forma, pode-se implementar os operadores Hadamard e Toffoli (portas quânticas universais) com o nó $|\psi\rangle$ -RAM, dessa forma, podendo ser um aproximador de qualquer circuito quântico existente.

As figuras (4.9) e (4.10) mostram a simulação dos operadores Hadamard e Toffoli respectivamente. Como dito, é necessário que o terceiro registrador não possua valor fixo.

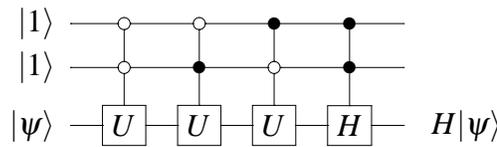


Figura 4.9: Nó $|\psi\rangle$ -ROM simulando o operador Hadamard

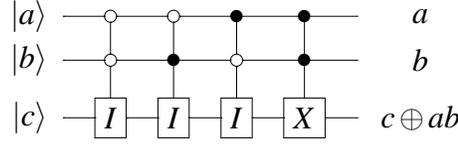


Figura 4.10: Nó $|\psi\rangle$ -ROM simulando a porta Toffoli

4.3.5 Nó PQM

Os modelos anteriormente citados são quantizações dos modelos de redes neurais quânticas sem peso e são interpretados como matrizes, cujo tamanho cresce exponencialmente de acordo com a quantidade de entradas. Da Silva et al. propuseram o modelo PQM [16] em que se usa um estado ou um registrador quântico cujo tamanho cresce linearmente com a entrada, inspirado no modelo das Memórias Quânticas Probabilísticas (MQP) propostas por Trugenberger [33].

Um outro ponto a favor destacado das MQPs é que um único nó MQP é capaz de fazer generalização, enquanto um único nó qRAM não é capaz, trabalhando em modo probabilístico.

Da Silva et al. também propõem uma mudança no algoritmo [34] de Trugenberger que faz com que seu algoritmo não seja limitado a reconhecimento de padrões linearmente independentes. Para explicar o nó MQP, iremos tratar antes das memórias MQP.

4.3.5.1 Memórias Quânticas Probabilísticas MQPs

Uma memória MQP é um estado quântico $|m\rangle$ que representa os j padrões p^j em superposição para serem armazenados. O algoritmo que cria esta superposição é detalhado em [33]. Com este algoritmo, 2^n padrões podem ser armazenados em n qubits. Eles são armazenados e salvos em $|m\rangle$:

$$|m\rangle = \frac{1}{\sqrt{p}} \sum_{j=1}^p |p^j\rangle \quad (4.13)$$

O algoritmo que recupera a informação requer 3 registradores: o primeiro \mathbf{i} irá receber o padrão de entrada, o segundo registrador \mathbf{m} onde os padrões foram salvos e irá conter a memória $|m\rangle$ e o terceiro \mathbf{c} é um qubit $|c\rangle$ auxiliar inicializado com $\mathbf{H}|0\rangle$.

$$|\psi_0\rangle = \frac{1}{\sqrt{2p}} \sum_{k=1}^p |i_1, \dots, i_n; p_1^k, \dots, p_n^k; 0\rangle + \frac{1}{\sqrt{2p}} \sum_{k=1}^p |i_1, \dots, i_n; p_1^k, \dots, p_n^k; 1\rangle \quad (4.14)$$

Depois da parte determinística de recuperação, o estado quântico é indicado pela equação (4.15).

$$\begin{aligned}
|\psi\rangle &= \frac{1}{\sqrt{2^p}} \sum_{k=1}^p \cos \frac{\pi}{2n} d_H(i, p^k) |i_1, \dots, i_n; p_1^k; \dots; p_n^k; 0\rangle \\
&+ \frac{1}{\sqrt{2^p}} \sum_{k=1}^p \sin \frac{\pi}{2n} d_H(i, p^k) |i_1, \dots, i_n; p_1^k; \dots; p_n^k; 1\rangle
\end{aligned} \tag{4.15}$$

onde $d_H(i, p^k)$ é a distância Hamming entre i e p^k . A medição do registrador c diz se o padrão é reconhecido (se $|c\rangle = |0\rangle$) ou não reconhecido (se $|c\rangle = |1\rangle$). Este processo pode ser repetido T vezes, já que a medição é probabilística, fazendo com que quanto maior for a quantidade de vezes medida, mais próxima da certeza de se saber se aquela entrada pertence àquela classe. Se o qubit $|c\rangle$ for $|0\rangle$, classifica-se i como reconhecido e mede-se o registrador de memória a fim de identificá-lo.

Retornando ao nó MQP, verifica-se que ele também irá fazer o processo de armazenamento no registrador $|m\rangle$ dos padrões em superposição descrito anteriormente porém não fará a medição desse qubit, pois medir é colapsar e perder informações de seus estados, obrigando refazer o processo de armazenamento de padrões em superposição.

Um nó MQP de n qubits de entrada tem três registradores: o primeiro é o registrador \mathbf{i} com n qubits que é a entrada do nó; o segundo é o registrador \mathbf{m} que armazena a memória do nó, através do algoritmo detalhado por Trugenberger, e o terceiro registrador é o \mathbf{c} com um único qubit. A medição desse último registrador, como foi explicado anteriormente, dirá se o padrão apresentado na entrada pertence ou não àquele padrão. O nó MQP então possui $2^n + 1$ qubits.

O aprendizado do nó MQP é então a criação da superposição $|m\rangle$ de todos os padrões a serem armazenados como treinamento citado anteriormente, mas detalhado em [33]. O custo de treinamento é linear em relação à quantidade de padrões no conjunto de treinamento, $\Theta(n_2)$.

Figura 4.11: Nó MQP

Para avaliar um novo padrão ao nó, a entrada é aplicada ao nó MQP que pode ser visto como um operador quântico descrito em forma de circuito pela Figura (4.11). A medição do registrador \mathbf{c} irá determinar se o nó pertence ou não ao padrão sobre as seguintes probabilidades:

$$\begin{cases} P(|c\rangle = |0\rangle) = \sum_{k=1}^p \frac{1}{p} \cos^2\left(\frac{\pi}{2n} d_H(i, p^k)\right) \\ P(|c\rangle = |1\rangle) = \sum_{k=1}^p \frac{1}{p} \sin^2\left(\frac{\pi}{2n} d_H(i, p^k)\right) \end{cases} \tag{4.16}$$

Uma desvantagem do nó qRAM é que um único nó não é capaz de fazer generalização. Com o nó MQP, como a distância entre a entrada e os padrões armazenados em $|m\rangle$ levam em conta a distância Hamming, então, ele é capaz de generalizar pois é possível que essa distância seja muito grande, grande, mediana, baixa, etc. Essa valoração de distância contínua é refletida no qubit de saída, que dará a probabilidade daquele padrão pertencer ao conjunto armazenado.

Uma limitação do nó MQP é sua saída probabilística. Isso pode ser minimizado executando o nó um número grande de vezes e valorando a entrada com a saída mais frequente [35].

Modelos de Dinâmicas para Neurônios Quânticos Sem Peso

. Muitos algoritmos quânticos seguem um comportamento ciclicamente iterativo. O algoritmo de busca de Grover repete seus passos em $\Theta(\sqrt{n})$ vezes [5]. Tradicionalmente, essa repetição de passos é vista como a repetição física do conjunto de operadores e portas quânticas $\Theta(\sqrt{n})$ vezes. Mas ela pode ser entendida como um conjunto de operadores que, através de um "loop" fechado, reaplica sua saída à entrada e os qubits são medidos depois de $\Theta(\sqrt{n})$ ciclos dessa iteração. Apesar de alguns algoritmos quânticos terem propriedades iterativas, eles são desenhados com portas quânticas acíclicas, apesar da importância e motivação de utilizar-se redes cíclicas.

O que se motiva estudar redes cíclicas na computação quântica é que o comportamento iterativo é muito utilizado na computação tradicional. Os comandos "faça enquanto" e "repita" são subrotinas frequentemente utilizadas nos programas e pode ser complicado implementá-las por métodos acíclicos. O desenho da soluções em hardware já pode ser feito para ter circuitos de realimentação, como os de sistemas de controle.

Outra razão para querer estudar redes cíclicas de portas quânticas é o fato de que alguns sistemas físicos podem requerer um comportamento de rede cujas entradas e saídas sejam dependentes. Isso acontece em sistemas de controle e pode ser exemplificado por um robot quântico [36] [37] interagindo com um ambiente para navegação ou identificação, onde um computador quântico controla suas operações. Entender do ponto de vista acíclico esta arquitetura robótica é muito difícil dado que o robô quântico englobaria muitas variáveis e um volume de dados em que a decisão atual estaria associada com a decisão anterior. Um outra razão para estudos cíclicos de redes é entender sua relação com o problema da Parada. Em redes acíclicas, é possível determinar se um algoritmo parará, porém, este comportamento não é provado para o comportamento cíclico de complexidade arbitrária. Este problema é equivalente ao problema da parada das Máquinas de Turing [39].

Em [39] é possível entender a dinâmica de redes cíclicas quânticas, sob o ponto de vista dos operadores, sua análise de fase, pela extração de seus auto-valores, auto-vetores e auto-estados.

No contexto de processamento de informação quântica, Lloyd [38] demonstrou que informação em rede cíclica se beneficia onde nenhuma medição é feita. Porém, a unitariedade das transformações quânticas geralmente impede a sensibilidade exponencial às condições iniciais, embora haja exceções [17], assim como a medição de sistemas quânticos afeta sua dinâmica [15] e um comportamento não-linear pode emergir do sistema. Este comportamento não-linear é de interesse de quem analisa sistemas dinâmicos na busca de comportamento caótico.

Neste capítulo, apresentamos modelos de dinâmicas caóticas que iteram sobre neurônios quânticos. Como referência de dinâmica, analisamos a dinâmica proposta por Kiss et al. através da utilização de um operador não-linear criador por Bechmann et al [14]. Essa transformação embora seja não-linear (e não preserve o traço) é fisicamente possível de ser realizada num laboratório, conforme explica o autor, pois a única dificuldade de sua implementação seria por deficiência inerentemente tecnológica.

Inicialmente, um modelo de dinâmica simples é proposto onde se verifica alto grau de sensibilidade paramétrica, além de um comportamento sensível às valorações das casas decimais. Além disso, através desse modelo simples, podemos verificar que os valores dos seletores podem dar ao nó um comportamento convergente ou não em sua dinâmica. Os demais modelos analisados foram inspirados em um modelo não-linear quadrático proposto por Kiss et al.

5.1 Modelo de dinâmica simples para o qRAM (mds-qRAM)

Analisar a dinâmica de um modelo é fazer com que sua saída seja realimentada como entrada e acompanhar a valoração e comportamento desta saída. O desafio, na computação quântica, é de que "medir" a saída para realimentá-la na entrada é perder informação. Se, antes de medir, as amplitudes assumiam valor complexo e estavam em superposição quântica, depois de medir um estado $\alpha|0\rangle + \beta|1\rangle$, iremos enxergar com probabilidades $|\alpha|^2$ e $|\beta|^2$ os estados respectivamente $|0\rangle$ e $|1\rangle$. Neste modelo, então, investigamos a dinâmica do qRAM extraindo o valor do qubit de saída $|o\rangle$ através de uma "medição" das amplitudes. Esta dinâmica acontece com a seguinte realimentação:

$$|\psi\rangle_{t+1} = qRAM(|\psi\rangle_t) \quad (5.1)$$

onde $|\psi\rangle = |i\rangle|s_0\rangle|s_1\rangle|o\rangle$.

5.1.1 Medindo o qubit de saída

Apesar dos operadores do qRAM serem em suma operadores CNOT (Não-controlados), onde o qubit de saída é sempre o qubit-alvo e as entradas escolhem os seletores que serão qubits-de-controle, não se é possível determinar qual o estado quântico do qubit de saída isolado após a operação. Apresentamos a prova deste teorema abaixo:

Teorema 5.1.1. *Qubits em superposição que são controlados pela família de operadores do tipo U-Controlado são emaranhados.*

Prova: Se considerarmos os qubits $|u\rangle$ e $|v\rangle$:

$$|u\rangle = \begin{pmatrix} a \\ b \end{pmatrix} \quad (5.2)$$

$$|v\rangle = \begin{pmatrix} c \\ d \end{pmatrix} \quad (5.3)$$

O produto tensorial entre eles é:

$$|u\rangle|v\rangle = \begin{pmatrix} ac \\ ad \\ bc \\ bd \end{pmatrix} \quad (5.4)$$

Se aplicarmos a este qubit a operação do U-Controlado ao segundo qubit, onde U é:

$$U(\alpha) = \begin{pmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{pmatrix} \quad (5.5)$$

Então:

$$A_U|u\rangle|v\rangle = \begin{pmatrix} ac \\ ad \\ bc * \cos(\alpha) - bd * \sin(\alpha) \\ bc * \sin(\alpha) + bd * \cos(\alpha) \end{pmatrix} \quad (5.6)$$

O qubit $|u\rangle$ não deveria ter sido modificado pela operação e parece intuitivo achar que o segundo qubit $|v\rangle$ seja recuperável na sua forma de estado quântico. Provaremos que não é possível, a não ser que haja restrição no operador U.

Considerando que apenas o segundo qubit deveria ter sido modificado, o qubit $|u\rangle$ é o mesmo inicial. O segundo tem valor desconhecido $|v\rangle_{t+1} = x|0\rangle + y|1\rangle$.

O produto tensorial entre eles dá: $|uv_{t+1}\rangle = ax|00\rangle + ay|01\rangle + bx|10\rangle + by|11\rangle$. Igualando $A_U|u\rangle|v\rangle$ com $|uv_{t+1}\rangle$, para qubits de que não estejam na base computacional, temos que:

$$\begin{aligned} c &= x \\ d &= y \\ c * \cos(\alpha) - d * \sin(\alpha) &= x \\ c * \sin(\alpha) + d * \cos(\alpha) &= y. \end{aligned} \quad (5.7)$$

Como podemos ver, os termos em função do primeiro qubit somem das equações e há condições somente sobre o segundo qubit. Uma delas é que os qubits sejam iguais antes e depois da aplicação do operador U. Isso só acontece se $\alpha = 0$, ou seja $U = \text{Identidade}$ e que $c = d = x = y$, ou seja, os qubits $|u\rangle = |v\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ devem ser iguais e possuir amplitudes iguais nos seus estados.

Isolando $(\cos(\alpha) - 1)$ na terceira e quarta equações anteriores, uma outra condição é imposta, a de que $y^2 = -x^2$.

Ou seja, para ser possível extrair o qubit operado é necessário que o U-controlado seja o operador identidade e que os qubits sejam iguais e em amplitudes iguais. Logo, não é possível extrair o qubit alvo quando um par de qubits são aplicados em um operador U-Controlado, como é o caso do qRAM e dos outros modelos de neurônios quânticos.

Considerando esta prova, mostraremos qual a estratégia proposta para medição do qubit de saída. Ao aplicarmos um estado quântico $|\psi\rangle$ num operador qRAM, teremos após disso um

outro estado quântico. A idéia proposta então é extrair o qubit de saída pela medição desse novo estado quântico. Por exemplo, se temos um estado de 2 qubits $|\phi\rangle = a_0|00\rangle + a_1|01\rangle + a_2|10\rangle + a_3|11\rangle$, a teoria nos diz que o segundo qubit assumirá com probabilidade $|a_0|^2 + |a_2|^2$ o valor $|0\rangle$ e $|a_1|^2 + |a_3|^2$ o valor $|1\rangle$.

Neste caso, se temos um estado quântico de 4 qubits $|\psi_{saida}\rangle$, então ele terá o seguinte formato:

$$|\psi\rangle = \begin{pmatrix} \psi_0 \\ \psi_1 \\ \dots \\ \psi_{15} \end{pmatrix} \begin{matrix} |0000\rangle \\ |0001\rangle \\ \dots \\ |1111\rangle \end{matrix} \quad (5.8)$$

Descobrir as probabilidades de encontrar $|0\rangle$ e $|1\rangle$ no último qubit é considerar, então, a soma das probabilidades dos estados $|xxx0\rangle$ e $|xxx1\rangle$, respectivamente. Para isso, soma-se as normas ao quadrado das posições pares desse vetor e atribui-se à amplitude do estado $|0\rangle$ a raiz dessa soma. O mesmo se faz com as posições ímpares e atribui-se à amplitude do estado $|1\rangle$ a raiz dessa soma.

$$\alpha^2 = \sum_{ipar} |\psi_i|^2 \quad (5.9)$$

$$\beta^2 = \sum_{impar} |\psi_i|^2 \quad (5.10)$$

$$|o\rangle_{saida} = \alpha|0\rangle + \beta|1\rangle \quad (5.11)$$

Se considerarmos que a entrada $|i\rangle$ esteja na forma:

$$|i\rangle = \begin{pmatrix} i_0 \\ i_1 \end{pmatrix} \quad (5.12)$$

Que os seletores $|s_0\rangle$ e $|s_1\rangle$ são:

$$|s_0\rangle = \begin{pmatrix} s_0^0 \\ s_1^0 \end{pmatrix} \quad (5.13)$$

$$|s_1\rangle = \begin{pmatrix} s_1^0 \\ s_1^1 \end{pmatrix} \quad (5.14)$$

E que o qubit de saída esteja no formato:

$$|o\rangle = \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \quad (5.15)$$

O estado de entrada será:

$$|\psi\rangle = \begin{pmatrix} i_0 * s_0^0 * s_1^0 * \alpha \\ i_0 * s_0^0 * s_1^0 * \beta \\ i_0 * s_0^0 * s_1^1 * \alpha \\ i_0 * s_0^0 * s_1^1 * \beta \\ i_0 * s_0^1 * s_1^0 * \alpha \\ i_0 * s_0^1 * s_1^0 * \beta \\ i_0 * s_0^1 * s_1^1 * \alpha \\ i_0 * s_0^1 * s_1^1 * \beta \\ i_1 * s_0^0 * s_1^0 * \alpha \\ i_1 * s_0^0 * s_1^0 * \beta \\ i_1 * s_0^0 * s_1^1 * \alpha \\ i_1 * s_0^0 * s_1^1 * \beta \\ i_1 * s_0^1 * s_1^0 * \alpha \\ i_1 * s_0^1 * s_1^0 * \beta \\ i_1 * s_0^1 * s_1^1 * \alpha \\ i_1 * s_0^1 * s_1^1 * \beta \end{pmatrix} \quad (5.16)$$

Quando aplicamos a entrada $|\psi\rangle$ ao qRAM, temos a seguinte saída:

$$qRAM|\psi\rangle = \begin{pmatrix} i_0 * s_0^0 * s_1^0 * \alpha \\ i_0 * s_0^0 * s_1^0 * \beta \\ i_0 * s_0^0 * s_1^1 * \alpha \\ i_0 * s_0^0 * s_1^1 * \beta \\ i_0 * s_0^1 * s_1^0 * \beta \\ i_0 * s_0^1 * s_1^0 * \alpha \\ i_0 * s_0^1 * s_1^1 * \beta \\ i_0 * s_0^1 * s_1^1 * \alpha \\ i_1 * s_0^0 * s_1^0 * \alpha \\ i_1 * s_0^0 * s_1^0 * \beta \\ i_1 * s_0^0 * s_1^1 * \beta \\ i_1 * s_0^0 * s_1^1 * \alpha \\ i_1 * s_0^1 * s_1^0 * \alpha \\ i_1 * s_0^1 * s_1^0 * \beta \\ i_1 * s_0^1 * s_1^1 * \beta \\ i_1 * s_0^1 * s_1^1 * \alpha \end{pmatrix} \quad (5.17)$$

O que significa que as amplitudes do qubit de saída podem ser medidas e extraídas:

$$\alpha^2 = |i_0 s_0^1 s_1^0 \alpha|^2 + |i_0 s_0^1 s_1^1 \alpha|^2 + |i_1 s_0^0 s_1^1 \alpha|^2 + |i_1 s_0^1 s_1^1 \alpha|^2 + |i_0 s_0^0 s_1^0 \beta|^2 + |i_0 s_0^0 s_1^1 \beta|^2 + |i_1 s_0^0 s_1^0 \beta|^2 + |i_1 s_0^1 s_1^0 \beta|^2 \quad (5.18)$$

$$\beta^2 = |i_0 s_0^0 s_1^0 \alpha|^2 + |i_0 s_0^0 s_1^1 \alpha|^2 + |i_1 s_0^0 s_1^0 \alpha|^2 + |i_1 s_0^1 s_1^0 \alpha|^2 + |i_0 s_0^1 s_1^0 \beta|^2 + |i_0 s_0^1 s_1^1 \beta|^2 + |i_1 s_0^0 s_1^1 \beta|^2 + |i_1 s_0^1 s_1^1 \beta|^2 \quad (5.19)$$

Após extrair o qubit $|o\rangle$, experimentamos (a) realimentá-lo nele mesmo e (b) nele mesmo e na entrada.

5.1.2 Análise

Ao realimentar a saída $|o\rangle$ nela mesma, verificamos um comportamento convergente (Figuras (5.1), (5.2), (5.4), (5.5), (5.6)) e não-convergente (Figura (5.3)) a depender dos valores dos seletores.

Ao realimentar a saída $|o\rangle$ no qubit de saída e no qubit de entrada $|i\rangle$, verificamos também um comportamento convergente, assintótico (Figuras (5.7), (5.9), (5.10), (5.11)) e não-assintótico (Figura (5.8)) a depender dos valores dos seletores.

Tais resultados demonstraram uma sensibilidade considerável em valores iniciais e dependência dos valores decimais para ditar o seu comportamento. É possível verificar que o comportamento extrapola seu padrão anterior simplesmente porque atingiu valores nas casas decimais suficientes para modificar o seu comportamento. Isto é visto, por exemplo, na Figura (5.11), que demonstra que apesar de o valor parecer ter convergido em 0.5 nas 10^a e 40^a iterações, o comportamento é alterado e muda seu valor de convergência para 0.0.

As figuras que demonstraram não-convergência parecem representar uma modulação em amplitude AM.

Através da exploração das possibilidades das amplitudes do qubit de entrada e fixando os seletores podemos ver o comportamento convergente ou não da dinâmica, na Figura (5.12).

Nas Figuras (5.13), (5.14), (5.15) e (5.16), pode-se verificar o comportamento da convergência dado a fixação da entrada e a variação das amplitudes dos seletores. Vemos que valores mínimos ou máximo em ambos os seletores fazem, em sua grande maioria, com que o comportamento divergente aconteça. Considerou-se como convergente o critério do $x_n = x_{n-1}$ para n consideravelmente grande. Os experimentos mostraram que um n suficiente seria 50.

Os experimentos foram executados na linguagem Python 2.7.3 que também utiliza o GCC 4.6.3, em um computador 64 bits, processador Intel®Core™i3 CPU M 350 @ 2.27GHz x 4.

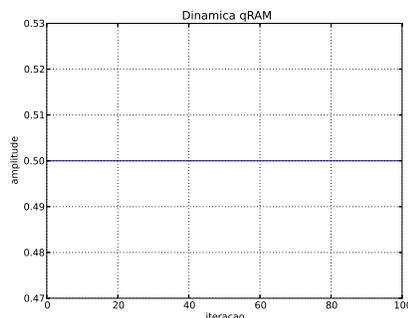


Figura 5.1: Modelo simples de dinâmica - saída realimentada para ela mesma $|\Psi\rangle = |1\rangle$, $|s_0\rangle = |0\rangle$, $|s_1\rangle = |0\rangle$, $|o\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$

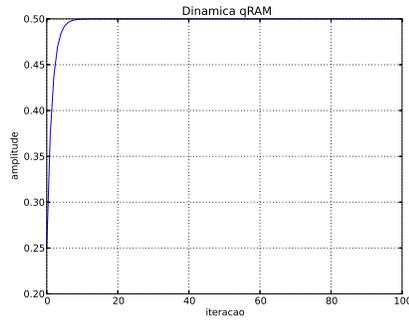


Figura 5.2: Modelo simples de dinâmica - saída realimentada para ela mesma $|\Psi\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$, $|s_0\rangle = |0\rangle$, $|s_1\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$, $|o\rangle = |0\rangle$

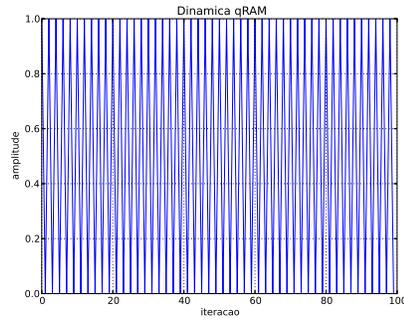


Figura 5.3: Modelo simples de dinâmica - saída realimentada para ela mesma e na entrada $|\Psi\rangle = |1\rangle$, $|s_0\rangle = |0\rangle$, $|s_1\rangle = |1\rangle$, $|o\rangle = |0\rangle$

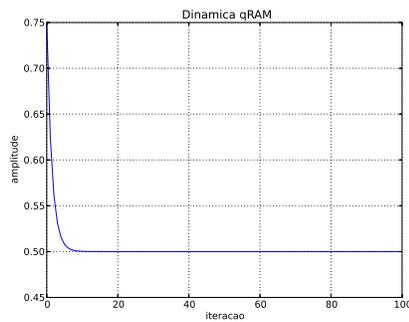


Figura 5.4: Modelo simples de dinâmica - saída realimentada para ela mesma $|\Psi\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$, $|s_0\rangle = |0\rangle$, $|s_1\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$, $|o\rangle = |1\rangle$

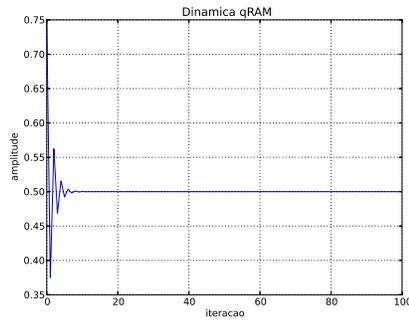


Figura 5.5: Modelo simples de dinâmica - saída realimentada para ela mesma $|\Psi\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$, $|s_0\rangle = |1\rangle$, $|s_1\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$, $|o\rangle = |0\rangle$

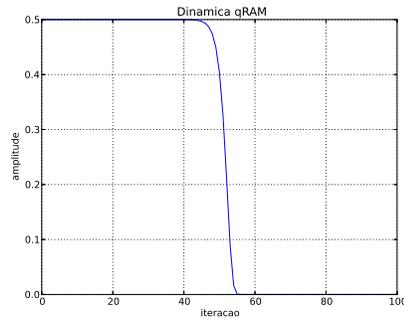


Figura 5.6: Modelo simples de dinâmica - saída realimentada para ela mesma e na entrada $|\Psi\rangle = |1\rangle$, $|s_0\rangle = |0\rangle$, $|s_1\rangle = |0\rangle$, $|o\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$

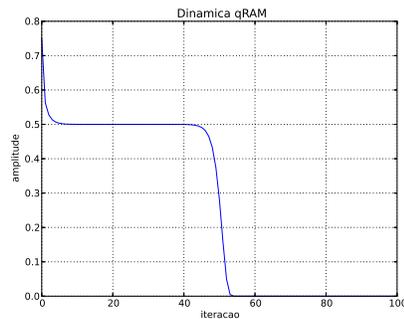


Figura 5.7: Modelo simples de dinâmica - saída realimentada para ela mesma e na entrada $|\Psi\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$, $|s_0\rangle = |0\rangle$, $|s_1\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$, $|o\rangle = |1\rangle$

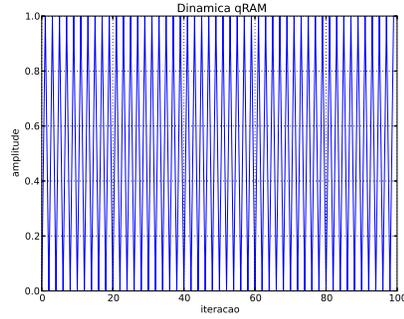


Figura 5.8: Modelo simples de dinâmica - saída realimentada para ela mesma e na entrada $|\Psi\rangle = |1\rangle, |s_0\rangle = |1\rangle, |s_1\rangle = |1\rangle, |o\rangle = |1\rangle$

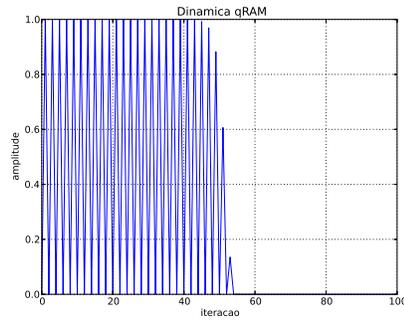


Figura 5.9: Modelo simples de dinâmica - saída realimentada para ela mesma e na entrada $|\Psi\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle), |s_0\rangle = |1\rangle, |s_1\rangle = |1\rangle, |o\rangle = |1\rangle$

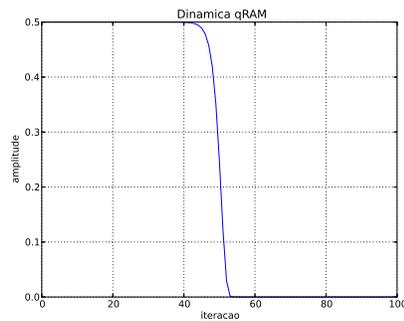


Figura 5.10: Modelo simples de dinâmica - saída realimentada para ela mesma e na entrada $|\Psi\rangle = |1\rangle, |s_0\rangle = |1\rangle, |s_1\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle), |o\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$

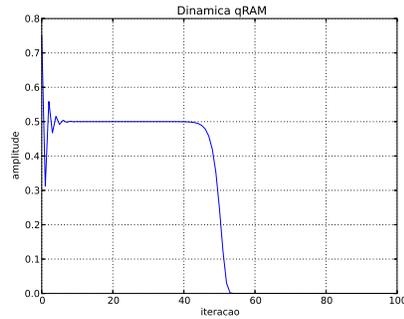


Figura 5.11: Modelo simples de dinâmica - saída realimentada para ela mesma e na entrada $|\Psi\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$, $|s_0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$, $|s_1\rangle = |1\rangle$, $|o\rangle = |0\rangle$

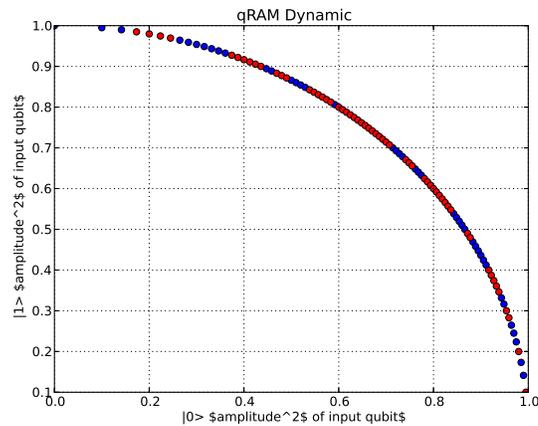


Figura 5.12: Modelo simples de dinâmica - análise do comportamento convergente quando a saída é realimentada para ela mesma e na entrada. Os eixos representam as variações das amplitudes do qubit de entrada para seletores fixos: $|s_0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$, $|s_1\rangle = |1\rangle$, $|o\rangle = |0\rangle$. Pontos em azul representam convergência e vermelho divergência

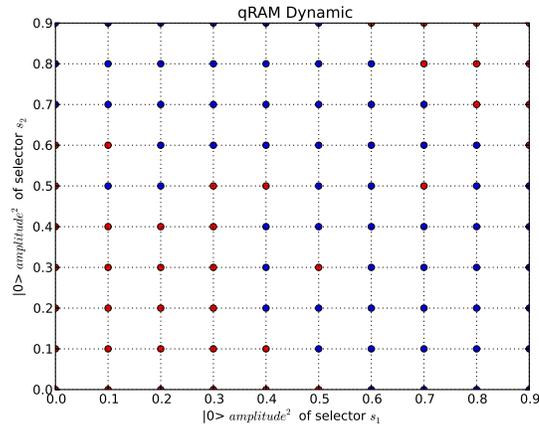


Figura 5.13: Modelo simples de dinâmica - análise do comportamento convergente quando a saída é realimentada para ela mesma e na entrada. Os eixos x e y representam a amplitude do qubit $|0\rangle$ dos seletores s_0 e s_1 respectivamente para um qubit de entrada fixo. $|i\rangle = 0.4472135955|0\rangle + 0.894427191|1\rangle$. Pontos em azul representam convergência e vermelho divergência

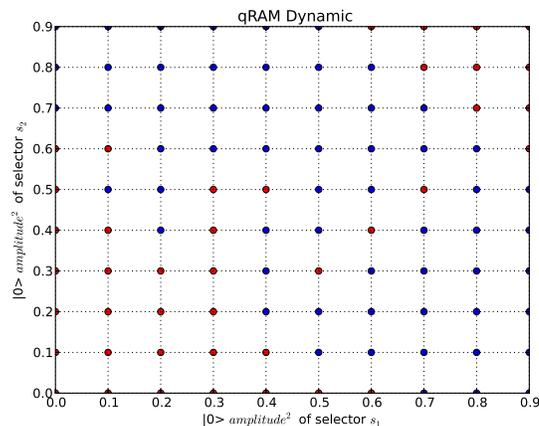


Figura 5.14: Modelo simples de dinâmica - análise do comportamento convergente quando a saída é realimentada para ela mesma e na entrada. Os eixos x e y representam a amplitude do qubit $|0\rangle$ dos seletores s_0 e s_1 respectivamente para um qubit de entrada fixo. $|i\rangle = 0.632455532034|0\rangle + 0.774596669241|1\rangle$. Pontos em azul representam convergência e vermelho divergência

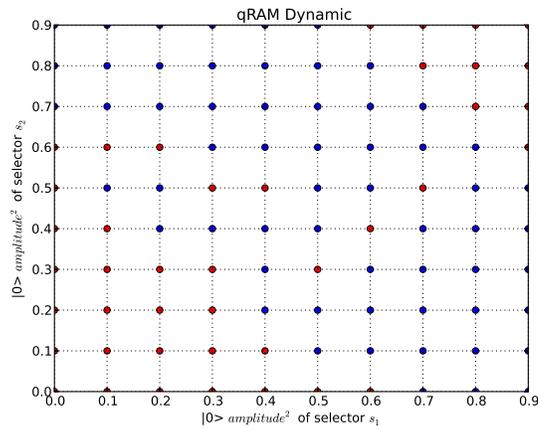


Figura 5.15: Modelo simples de dinâmica - análise do comportamento convergente quando a saída é realimentada para ela mesma e na entrada. Os eixos x e y representam a amplitude do qubit $|0\rangle$ dos seletores s_0 e s_1 respectivamente para um qubit de entrada fixo. $|i\rangle = 0.774596669241|0\rangle + 0.632455532034|1\rangle$. Pontos em azul representam convergência e vermelho divergência

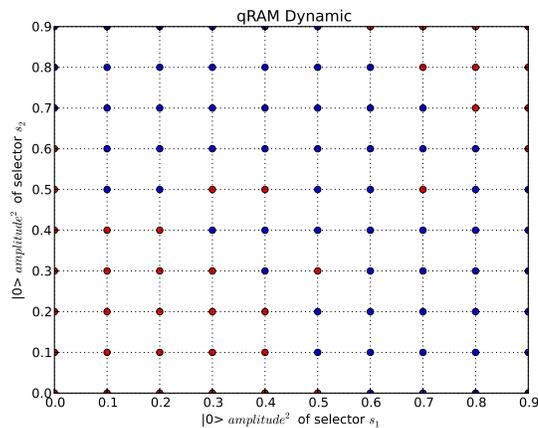


Figura 5.16: Modelo simples de dinâmica - análise do comportamento convergente quando a saída é realimentada para ela mesma e na entrada. Os eixos x e y representam a amplitude do qubit $|0\rangle$ dos seletores s_0 e s_1 respectivamente para um qubit de entrada fixo. $|i\rangle = 0.894427191|0\rangle + 0.4472135955|1\rangle$. Pontos em azul representam convergência e vermelho divergência

5.2 Modelo de dinâmica quântica do Kiss et al.

Kiss et al [15] propõem uma análise de dinâmica de um qubit utilizando o operador de transformação não-linear, proposto em [14]. Esse operador tem como utilidade já conhecida a possibilidade de diferenciar de forma otimizada entre estados de spin-1/2 não-ortogonais. Ela é mapeada pela Equação (5.20), onde N é o fator de normalização $1/\sum \rho_{ij}^2$.

$$\rho = S\rho, \quad \rho_{ij} \xrightarrow{S} N\rho_{ij}^2 \quad (5.20)$$

Então, se temos um único qubit, a transformação S proposta por Bechmann seria:

$$|\psi\rangle_{entrada} = \alpha|0\rangle + \beta|1\rangle \xrightarrow{S} |\psi\rangle_{saida} = N(\alpha^2|0\rangle + \beta^2|1\rangle) \quad (5.21)$$

Kiss et al. [15] propõem incluir essa transformação S para simular uma dinâmica de um qubit. Através dessa dinâmica, torná-la altamente não trivial com a inclusão do operador U genérico de rotação do qubit no espaço de Hilbert, com variáveis x e ϕ , como demonstrado na equação (5.22).

$$U(x, \phi) = \begin{pmatrix} \cos(x) & \sin(x)e^{i\phi} \\ -\sin(x)e^{-i\phi} & \cos(x) \end{pmatrix} \quad (5.22)$$

Fazemos abaixo, de forma detalhada, o processo proposto por Kiss et al, sem pular as etapas algébricas que são omitidas em seu artigo inicial. Considerando um estado inicial puro do qubit, com a seguinte notação:

$$\psi = N(z|0\rangle + |1\rangle) \quad (5.23)$$

onde N é o fator de normalização do qubit, $N = \frac{1}{\sqrt{1+|z|^2}}$. Um operador quadrático S [14] e um operador de rotação genérico são aplicados. As aplicações desses operadores são descritas abaixo:

$$\begin{aligned} |\psi\rangle &= N(z|0\rangle + |1\rangle) \xrightarrow{S} |\psi_2\rangle = N(z^2|0\rangle + |1\rangle) \\ U|\psi_2\rangle &= |\psi_3\rangle = N((z^2\cos(x) + \sin(x)e^{i\phi})|0\rangle + (-\sin(x)e^{-i\phi}z^2 + \cos(x))|1\rangle) \end{aligned} \quad (5.24)$$

Esse estado $|\psi_3\rangle$ é o estado quântico de saída após a primeira iteração do sistema dinâmico. Para recuperar o valor de z após essa dinâmica, é preciso deixar o estado quântico na forma original:

$$|\psi_{saida}\rangle = N(z'|0\rangle + |1\rangle) \quad (5.25)$$

Se pegarmos $|\psi_3\rangle$ e dividirmos o valor da amplitude em $|0\rangle$ pelo valor da amplitude em $|1\rangle$, chegaremos ao novo valor de z :

$$z' = \frac{z^2\cos(x) + \sin(x)e^{i\phi}}{-\sin(x)e^{-i\phi}z^2 + \cos(x)} \quad (5.26)$$

Nota-se que o fator de normalização não precisa ser levado em conta pois se anulará nessa divisão entre as amplitudes.

Considerando $p = \tan(x)e^{i\phi}$, temos a fórmula analítica deste modelo de dinâmica, estudada em [15]:

$$F_p(z) = \frac{z^2 + p}{-p^*z^2 + 1} \quad (5.27)$$

5.3 Modelo de dinâmica simples com estado puro para qRAM (mdsep-qRAM)

O qRAM possui uma estrutura quântica que dado n qubits de entrada, necessita-se a inclusão de mais $2^n + 1$ estados quânticos (2^n para os seletores e 1 para o qubit de saída). O processo de medição do qubit de saída também é realizado neste modelo. O diferencial é que, após a medição desse último qubit, um valor único representativo deste estado é extraído. A partir de seu valor $|o\rangle = \alpha|0\rangle + \beta|1\rangle$, conseguido pela medição, destacaremos o valor z no formato puro sugerido por Kiss et al:

$$|o\rangle_{puro} = N(z|0\rangle + |1\rangle) \quad (5.28)$$

Através deste formato, é possível ter uma variável em que, na dinâmica, possa-se extrairlo e verificar seu comportamento. É fácil imaginar que para isso é necessário fazer a divisão $z = \frac{\alpha}{\beta}$. Se β é 0, então a divisão vai à infinito e a dinâmica perde seu sentido. Dessa forma, consideramos que quando $\beta = 0$, $z = \alpha$.

As dinâmicas com este formato de estado puro apresentam comportamento bastante interessante: a grande maioria dos experimentos, os valores convergem para 0.5 e não houve, nos exemplos colhidos, situação em que a dinâmica não tenha convergido. Alimentar a saída $|o\rangle$ no seu qubit $|o\rangle$ e no de entrada $|i\rangle$ não modifica o comportamento. Não se notou a diferença entre os resultados de ambos os processos.

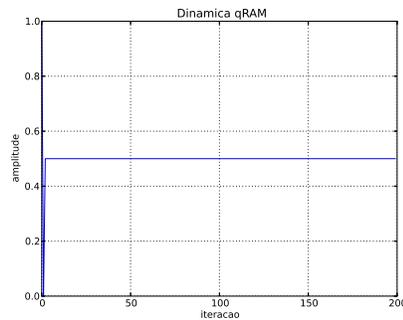


Figura 5.17: Modelo simples de dinâmica - saída realimentada para ela mesma e na entrada $|\Psi\rangle = |1\rangle, |s_0\rangle = |0\rangle, |s_1\rangle = |1\rangle, |o\rangle = |0\rangle$

O Conjunto Julia Preenchido deste modelo é simples como mostra a Figura (5.19). A execução deste algoritmo é lenta, pois se trata de uma figura rica em pontos dentro do círculo

5.4 MODELO DE DINÂMICA ELITISTA NÃO LINEAR DE ESTADO PURO PARA QRAM (MDENLEP-QRAM)

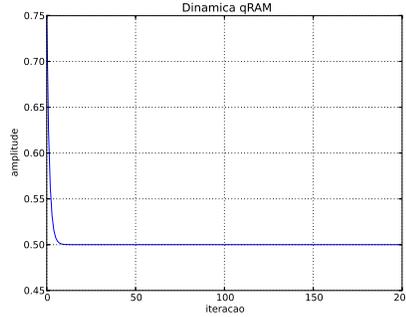


Figura 5.18: Modelo simples de dinâmica - saída realimentada para ela mesma e na entrada $|\Psi\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$, $|s_0\rangle = |0\rangle$, $|s_1\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$, $|o\rangle = |1\rangle$

unitário (pretos), permanecendo todas as iterações de sua dinâmica nele - o que a torna lenta. Nos experimentos realizados, parece que incluir o operador não-linear de Bechmann [14] não modifica o comportamento do conjunto Julia Preenchido.



Figura 5.19: Modelo simples de dinâmica - Na imagem, representação gráfica do Conjunto Julia Preenchido

5.4 Modelo de dinâmica elitista não linear de estado puro para qRAM (mdenlep-qRAM)

Com o objetivo de não precisar fazer medição do qubit de saída, como é o funcionamento das dinâmicas anteriores, esse modelo é proposto. Mas então como recuperar um valor único z dos estados quânticos como no modelo do Kiss se não temos como enxergar o qubit de saída $|o\rangle$, como foi provado pelo Teorema (5.1.1).

Neste modelo, para nossa simplificação, apenas alguns estados iniciais são considerados para a dinâmica - por isso o nome elitista. Dessa forma, podemos acompanhar a dinâmica sem que os qubits sejam emaranhados e o valor z complexo possa ser resgatado. O modelo pode ser dividido em duas etapas. Na primeira versão desse modelo, mdenlep-qRAM-1 há o uso de um operador não-linear de soma de amplitude e o uso de um operador não-linear quadrático. A segunda versão do modelo mdenlep-qRAM-2 considera a ordem contrária das duas etapas anteriores, ou seja, primeiro o operador não-linear quadrático é utilizado e depois é aplicado o operador não-linear de soma de amplitudes.

Na etapa de soma de amplitude, um operador não-linear X inclui um valor imaginário c nas amplitudes de $|0\rangle$ e $|1\rangle$, como descrito:

$$|\psi\rangle = N(z|0\rangle + |1\rangle) \xrightarrow{X} |\psi_x\rangle = N((z+c)|0\rangle + (1+c)|1\rangle) \quad (5.29)$$

A segunda etapa inclui o operador não-linear de purificação de estados misturados [14] descrito no início dessa seção. Para um único qubit, o processo se dá desta forma:

$$\begin{aligned} |\psi_x\rangle &= N((z+c)|0\rangle + (1+c)|1\rangle) \xrightarrow{S} \\ |\psi_{xs}\rangle &= N((z+c)^2|0\rangle + (1+c)^2|1\rangle) \end{aligned} \quad (5.30)$$

Como o operador qRAM exige $2^n + n + 1$ qubits no total, como foi dito, para nossa simplificação, consideraremos o modelo de estado inicial puro. Para um qRAM de quantidade de qubits de entrada $n=1$, apenas os estados abaixo são considerados:

$$\begin{aligned} |\psi\rangle &= N(z|0100\rangle + |0101\rangle + z|0110\rangle + \\ &+ |0111\rangle + z|1010\rangle + |1011\rangle + z|1110\rangle + |1111\rangle) \end{aligned} \quad (5.31)$$

Tais estados foram escolhidos pois serão os únicos afetados quando o qRAM for aplicado devido a natureza do operador CNOT que caracteriza o qRAM. As demais amplitudes são zeradas para simplificar nosso processo de captura do z após cada iteração da dinâmica. Essa simplificação do problema permite que o qRAM seja modelado como a aplicação da porta CNOT para 4 pares de qubits no formato de estado quântico puro.

Se repetimos a operação explicada anteriormente de isolamento do valor de z na sua forma analítica, encontraremos que nosso modelo mdenlep-qRAM-1 resulta em:

$$F_c(z) = \frac{(1+c)^2}{(z+c)^2} \quad (5.32)$$

O modelo mdenlep-qRAM-2 inverte a ordem de aplicação dos operadores não-lineares, gerando a expressão analítica facilmente encontrada:

$$F_c(z) = \frac{1+c}{z^2+c} \quad (5.33)$$

5.4.1 Análise

O comportamento do sistema é visto pelo Conjunto Julia preenchido nas Figuras (5.20)-(5.21) para o modelo 1 e Figuras (5.22)-(5.25) para o modelo proposto 2.

Os Diagramas de Orbitais são visualizados para valores de c imaginários nas Figuras (5.26) e (5.27) e para os modelos de dinâmica 1 e 2 respectivamente e é possível identificar caos em alguns dos valores de c .



Figura 5.20: Conjunto Julia Preenchido para o MCD-q1 aplicado ao qRAM, $c=0.0 + 3.0j$

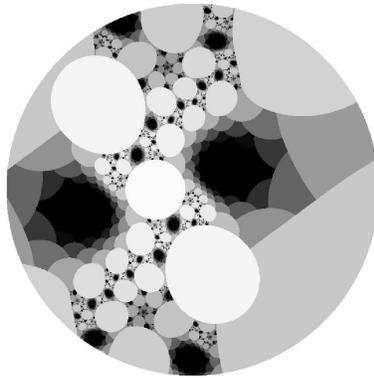


Figura 5.21: Conjunto Julia Preenchido para o MCD-q1 aplicado ao qRAM, $c=0.5 + 1.0j$

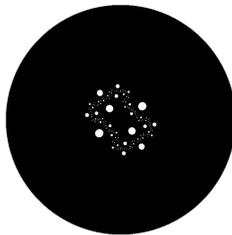


Figura 5.22: Conjunto Julia Preenchido para o MCD-q2 aplicado ao qRAM, $c=0.0 + 0.5j$

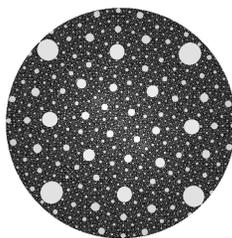


Figura 5.23: Conjunto Julia Preenchido para o MCD-q2 aplicado ao qRAM, $c=0.0 + 10.0j$

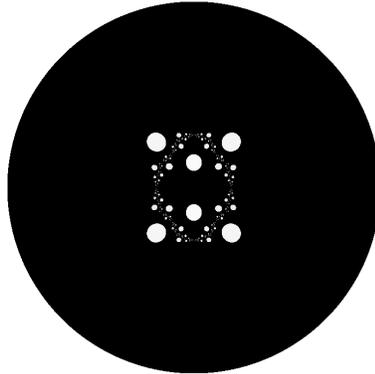


Figura 5.24: Conjunto Julia Preenchido para o MCD-q2 aplicado ao qRAM, $c=0.5 + 0.0j$

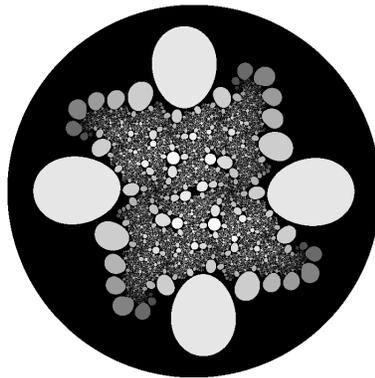


Figura 5.25: Conjunto Julia Preenchido para o MCD-q2 aplicado ao qRAM, $c=0.5 + 1.0j$

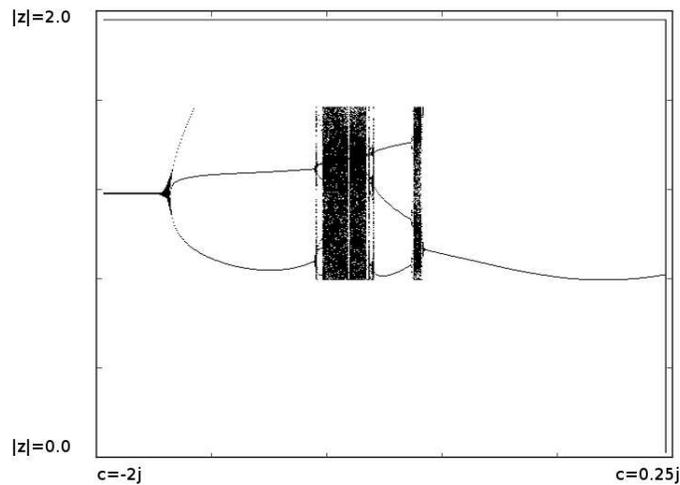


Figura 5.26: Diagrama Orbital para o MCD-q2 aplicado ao qRAM. Depois de várias iterações, os valores absolutos de z são plotados no intervalo de $-2j < c < 0.25j$

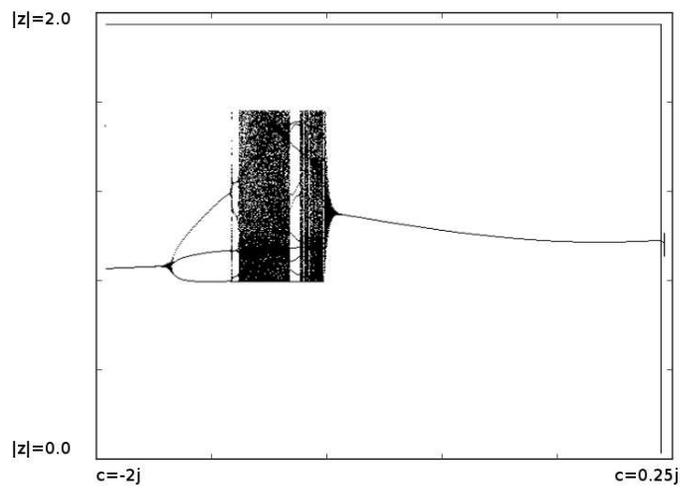


Figura 5.27: Diagrama Orbital para o MCD-q1 aplicado ao qRAM. Depois de várias iterações, os valores absolutos de z são plotados no intervalo de $-2j < c < 0.25j$

5.5 Modelo de dinâmica simples para o $|\psi\rangle$ -RAM (mds- $|\psi\rangle$ -RAM)

O $|\psi\rangle$ -RAM possui mais funções a serem executadas por padrão do que o qRAM. $|\psi\rangle$ -RAM permite inserir o operador quântico genérico U_t em suas operações, abrindo margem para que haja algum parâmetro livre para a simulação de sua dinâmica, sem necessidade de impor um parâmetro extra. Abaixo, encontram-se duas formas propostas de modelar a dinâmica de um nó $|\psi\rangle$ -RAM: (a) a saída $|o\rangle$ é sempre realimentada nela mesma e (b) a saída $|o\rangle$ é realimentada nele mesmo e no qubit de entrada. Neste modelo, analisamos o comportamento da amplitude do qubit de saída. Em ambos os casos, a saída é realimentada após o processo de medição explicado nos capítulos anteriores, sendo, então, uma realimentação sempre no domínio dos reais. A dinâmica tem os seguintes passos:

$$\begin{aligned} |\psi_{sp}\rangle &= |i\rangle|s\rangle|o_t\rangle * |\psi\rangle - RAM_U \\ |o_{t+1}\rangle &= \text{Medição}(|o\rangle) \end{aligned} \quad (5.34)$$

Para U_α :

$$U(\alpha) = \begin{pmatrix} \cos(\alpha) & \sin(\alpha) \\ -\sin(\alpha) & \cos(\alpha) \end{pmatrix} \quad (5.35)$$

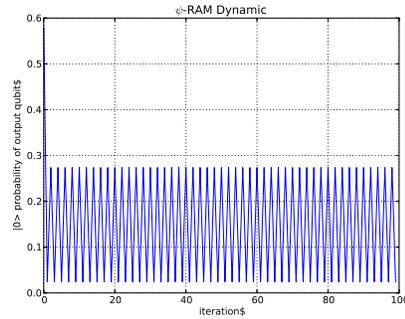


Figura 5.28: mds- $|\psi\rangle$ -RAM - Análise da amplitude do modelo do $|\psi\rangle$ -RAM, onde $|i\rangle = |1\rangle, |s_0\rangle = |0\rangle, |s_1\rangle = |0\rangle, |s_2\rangle = |1\rangle, |s_3\rangle = |1\rangle, |o\rangle = |0\rangle$ e $\alpha_U = \frac{\sqrt{2}}{2}$

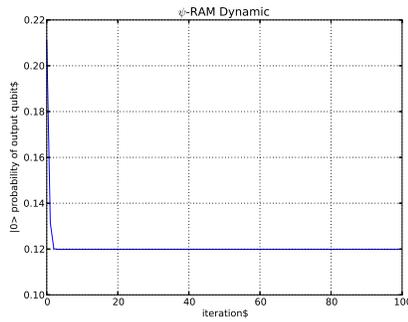


Figura 5.29: mds- $|\psi\rangle$ -RAM - Análise da amplitude do modelo do $|\psi\rangle$ -RAM, realimentando a saída apenas na saída, onde $|i\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$, $|s_0\rangle = |1\rangle$, $|s_1\rangle = |1\rangle$, $|s_2\rangle = |0\rangle$, $|s_3\rangle = |0\rangle$, $|o\rangle = |1\rangle$ e $\alpha_U = \frac{\sqrt{2}}{2}$

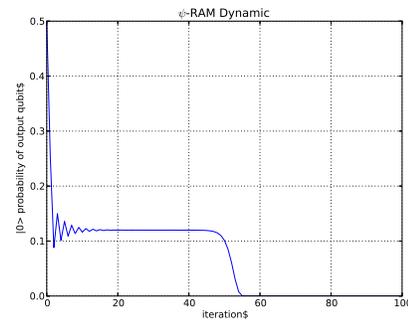


Figura 5.30: mds- $|\psi\rangle$ -RAM - Análise da amplitude do modelo do $|\psi\rangle$ -RAM, realimentando a saída na saída e no qubi de entrada onde $|i\rangle = |0\rangle$, $|s_0\rangle = |0\rangle$, $|s_1\rangle = |0\rangle$, $|s_2\rangle = |1\rangle$, $|s_3\rangle = |1\rangle$, $|o\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ e $\alpha_U = \frac{\sqrt{2}}{2}$

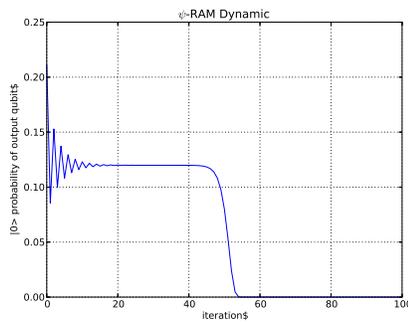


Figura 5.31: mds- $|\psi\rangle$ -RAM - Análise da amplitude do modelo do $|\psi\rangle$ -RAM, realimentando a saída na saída e no qubi de entrada onde $|i\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$, $|s_0\rangle = |0\rangle$, $|s_1\rangle = |0\rangle$, $|s_2\rangle = |1\rangle$, $|s_3\rangle = |1\rangle$, $|o\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ e $\alpha_U = \frac{\sqrt{2}}{2}$

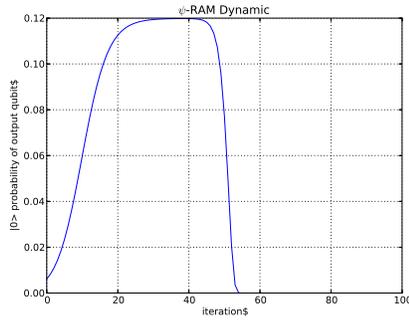


Figura 5.32: mds- $|\psi\rangle$ -RAM - Análise da amplitude do modelo do $|\psi\rangle$ -RAM, realimentando a saída na saída e no qubi de entrada onde $|i\rangle = |0\rangle, |s_0\rangle = |1\rangle, |s_1\rangle = |1\rangle, |s_2\rangle = |0\rangle, |s_3\rangle = |0\rangle, |o\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ e $\alpha_U = \frac{\sqrt{2}}{2}$

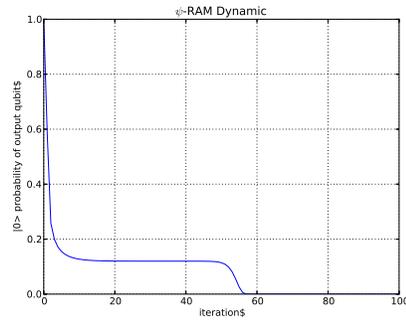


Figura 5.33: mds- $|\psi\rangle$ -RAM - Análise da amplitude do modelo do $|\psi\rangle$ -RAM, realimentando a saída na saída e no qubi de entrada onde $|i\rangle = |1\rangle, |s_0\rangle = |1\rangle, |s_1\rangle = |1\rangle, |s_2\rangle = |1\rangle, |s_3\rangle = |1\rangle, |o\rangle = |0\rangle$ e $\alpha_U = \frac{\sqrt{2}}{2}$

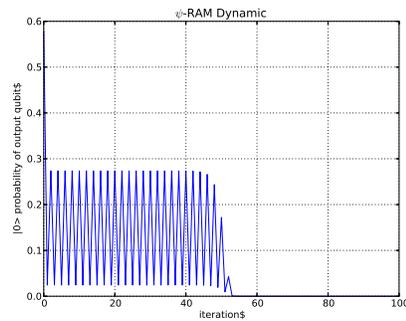


Figura 5.34: mds- $|\psi\rangle$ -RAM - Análise da amplitude do modelo do $|\psi\rangle$ -RAM, realimentando a saída na saída e no qubi de entrada onde $|i\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle), |s_0\rangle = |1\rangle, |s_1\rangle = |1\rangle, |s_2\rangle = |1\rangle, |s_3\rangle = |1\rangle, |o\rangle = |0\rangle$ e $\alpha_U = \frac{\sqrt{2}}{2}$

5.6 Modelo de dinâmica não-linear de estado puro para o $|\psi\rangle$ -RAM (mdnlep- $|\psi\rangle$ -RAM)

O $|\psi\rangle$ -RAM permite inserir o operador quântico genérico U_α em suas operações, abrindo margem para que haja algum parâmetro livre para a simulação de sua dinâmica, sem necessidade de impor um operador extra. Neste modelo, analisamos a dinâmica sobre o ponto de vista do Conjunto Julia Preenchido. Do ponto de vista experimental, verificamos que sem o operador não-linear S, a dinâmica é simples e não há representações significativas, logo sua presença é fundamental para a dinâmica. Abaixo, encontram-se duas formas propostas de modelar a dinâmica de um nó $|\psi\rangle$ -RAM: (a) a saída $|o\rangle$ é sempre realimentada nela mesma e a entrada $|\psi\rangle$ e os seletores $|s\rangle$ são sempre os estados $|1\rangle$ e (b) o resultado da saída $|o\rangle$ é realimentado nos qubits de entrada e seletores. Em ambos os casos, a saída é realimentada após o processo de medição explicado nos capítulos anteriores, sendo, então, uma realimentação sempre no domínio dos reais. Apenas o operador não-linear S [14] é utilizado, de forma que a dinâmica tenha os seguintes passos:

$$\begin{aligned}
|o\rangle &= N(z_t|0\rangle + 1|1\rangle) \xrightarrow{S} \\
|o_s\rangle &= N(z_t^2|0\rangle + 1^2|1\rangle) \\
|\psi_{sp}\rangle &= |i\rangle|s\rangle|o_s\rangle * |\psi\rangle - RAM_U \\
|o_d\rangle &= Medicao(|\psi_{sp}\rangle) \\
|o_d\rangle &= \alpha|0\rangle + \beta|1\rangle \\
z_{t+1} &= \frac{\alpha}{\beta}
\end{aligned} \tag{5.36}$$

Para U:

$$U(t) = \begin{pmatrix} \cos(t) & \sin(t) \\ -\sin(t) & \cos(t) \end{pmatrix} \tag{5.37}$$

Vale-se salientar que a dinâmica não apresenta CAOS quando a saída é escolhida inicialmente como $|0\rangle$. Iniciar a dinâmica com esta valoração parece restringir o comportamento a um subespaço de soluções em que todos os orbitais $a + bj$, com $a \in [-2,2]$ e $b \in [-2,2]$, pertençam ao conjunto de Julia Preenchido. Sendo assim, nossa dinâmica já acontece colocando os valores iniciais no qubit de saída $|o\rangle = N(z|0\rangle + |1\rangle)$

Quando apenas a saída é realimentada, e a entrada e seletores são escolhidos como $|1\rangle$, a dinâmica do nó quântico acontece como se o operador U estivesse sendo aplicado completamente a um único qubit. A única diferença entre o modelo anteriormente detalhado do Kiss et al. é que as amplitudes dos qubits são valores reais. Como se a fórmula fosse:

$$F_c(z) = \frac{|z^2 + c|}{|-z^2 p * + 1|} \tag{5.38}$$

Por simetria, podemos verificar que a entrada pode ser colocada como qualquer valor em superposição, pois ela está apenas ponderando a participação de cada uma das duplas de sele-

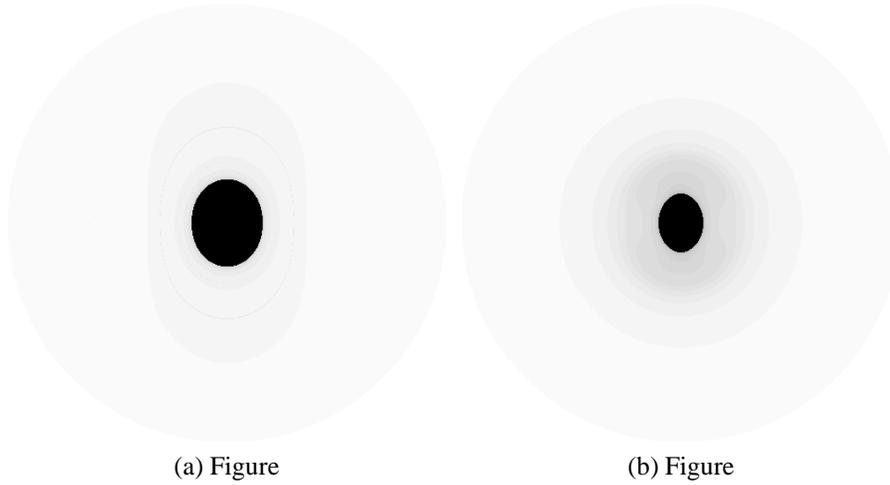


Figura 5.35: $mdnlep-|\psi\rangle$ -RAM - Conjunto Julia Preenchido para o Modelo de Dinâmica Não-linear ψ -RAM para $t_U = 0.1$. À esquerda, o conjunto da saída realimentada apenas na saída. À direita, a saída é a realimentada em todos os qubits de entrada

tores e, como ambos são iguais $|11\rangle$, ponderam os mesmos valores, que são os resultados do operador U no qubit de saída.

Quando a saída é realimentada em todos os qubits de entrada e seletores, o comportamento da função a ser visto de forma analítica é extensa.

5.6.1 Análise

As sensibilidades à dinâmica é muito mais expressiva no modelo $|\psi\rangle$ -RAM, como podemos ver nas Figuras (5.28)-(5.34). Nos parâmetros escolhidos, houve em todos os casos convergência absoluta. Esse modelo é expressivamente sensível ao processo da dinâmica, demonstrando comportamento não-linear significativo.

Através da plotagem do Conjunto Julia Preenchido, nas Figuras (5.35)-(5.38), podemos verificar que a saída realimentada nela mesma provoca mais destaque no conjunto. Podemos ver simetria e sensibilidade à condição inicial.

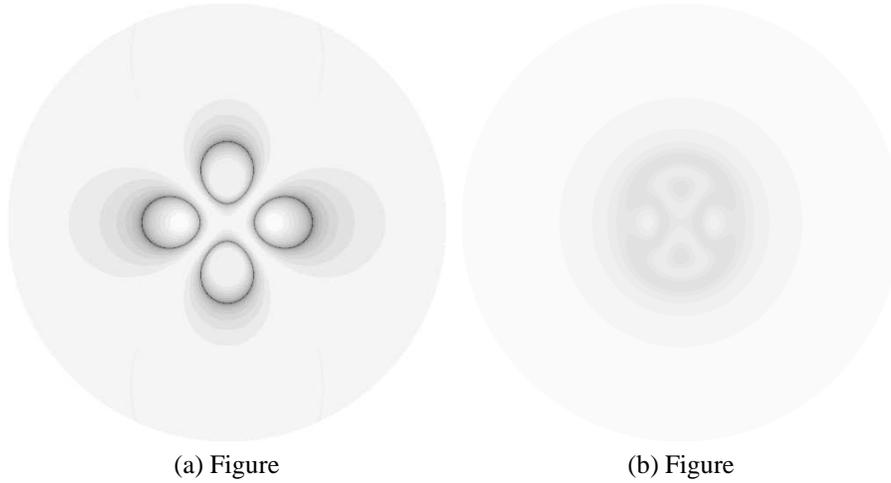


Figura 5.36: mdnlep- $|\psi\rangle$ -RAM - Conjunto Julia Preenchido para o Modelo de Dinâmica Não-linear ψ -RAM para $t_U = 0.9$. À esquerda, o conjunto da saída realimentada apenas na saída. À direita, a saída é a realimentada em todos os qubits de entrada

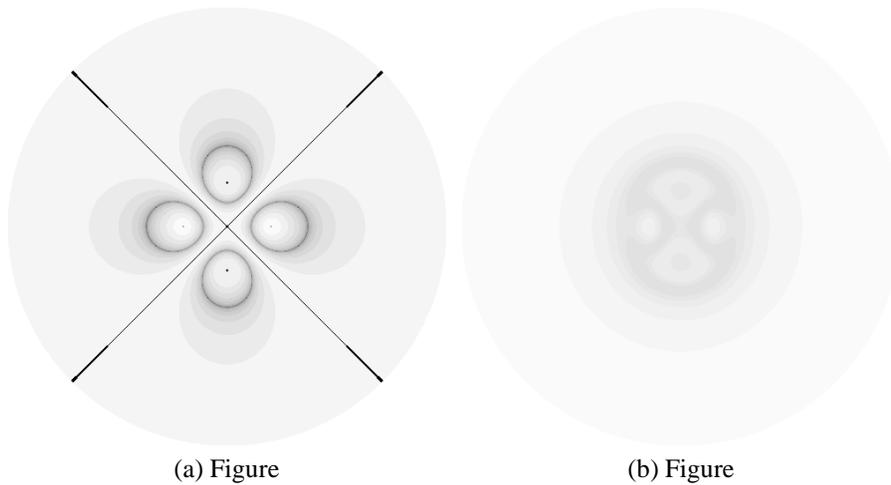


Figura 5.37: mdnlep- $|\psi\rangle$ -RAM - Conjunto Julia Preenchido para o Modelo de Dinâmica Não-linear ψ -RAM para $t_U = 1.0$. À esquerda, o conjunto da saída realimentada apenas na saída. À direita, a saída é a realimentada em todos os qubits de entrada

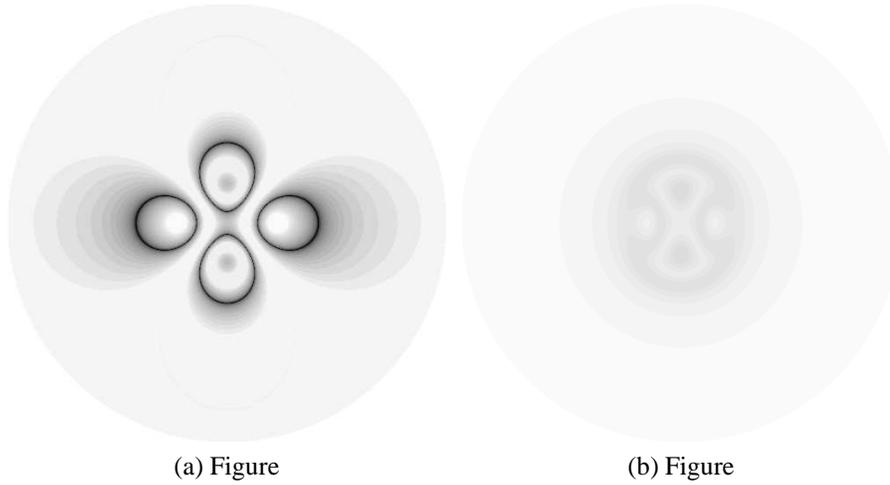


Figura 5.38: $\text{mdnlep-}|\psi\rangle$ -RAM - Conjunto Julia Preenchido para o Modelo de Dinâmica Não-linear ψ -RAM para $t_U = 0.8$. À esquerda, o conjunto da saída realimentada apenas na saída. À direita, a saída é a realimentada em todos os qubits de entrada

5.7 Comparação Quantitativa dos modelos

5.7.1 Definições

Modelo	Neurônio Quântico	Resumo
Modelo de dinâmica simples (mdsq-RAM)	qRAM	Foi verificado que a amplitude do qubit de saída é sensível às condições iniciais através de uma dinâmica de medição com realimentação da saída (a) na própria saída; (b) na própria saída e no qubit de entrada
Modelo de dinâmica simples com estado puro (mdsep-qRAM)	qRAM	Foi verificado que a amplitude do qubit de saída é sensível às condições iniciais através de uma dinâmica de medição com realimentação da saída (a) na própria saída; (b) na própria saída e na entrada. O diferencial neste modelo é que o qubit que é realimentado está no formato $N(z 0\rangle + 1\rangle)$. Por possuir um parâmetro z , gerou-se o conjunto Julia desse modelo que é simples.
Modelo de dinâmica elitista não linear de estado puro para qRAM (mdenlep-qRAM)	qRAM	Não é preciso aplicar o operador de medição neste modelo, mas é elitista porque restringe os estados iniciais a apenas aqueles atingíveis pelos operadores do não-controlado do qRAM. Possui alta sensibilidade paramétrica apresentada pelo Conjunto Julia Preenchido. Um operador não linear X é apresentado para que haja parametrização de uma constante c no modelo. A ordem de aplicação do operador não linear S e do operador não-linear X modifica o comportamento do sistema.
Modelo de dinâmica simples para o $ \psi\rangle$ -RAM (mds- $ \psi\rangle$ -RAM)	$ \psi\rangle$ -RAM	Foi verificado que a amplitude do qubit de saída é sensível às condições iniciais através da realimentação dele (a) na própria saída; (b) na própria saída e na entrada. Essa dinâmica tem comportamento não-linear rico em mudanças de convergência por causa das valorações decimais nos valores dos qubits durante a dinâmica. Por possuir o operador U com o parâmetro t , gerou-se o conjunto Julia desse modelo, demonstrando simetria e resultados representativos. Isso acontece mesmo em uma dinâmica simples, apenas com a realimentação do qubit de saída.
Modelo de dinâmica não-linear de estado puro para o $ \psi\rangle$ -RAM (mdnlep- $ \psi\rangle$ -RAM)	$ \psi\rangle$ -RAM	Foi verificado que o Conjunto Julia Preenchido da dinâmica do $ \psi\rangle$ -RAM é sensível à variação paramétrica e rica em simetria, na presença de operadores não-lineares.

5.7.2 Medida de comparação

Para compararmos os modelos propostos de forma quantitativa, propomos uma medida de variação δ . Essa medida representa o quanto há mudança e variação de valores da amplitude durante uma dinâmica. Ela é definida como o somatório das derivadas nos pontos amostrados da dinâmica:

$$\delta = \sum_t \frac{\partial \alpha(t)}{\partial t} \quad (5.39)$$

Ou na sua forma absoluta:

$$\delta_{abs} = \sum_t \left| \frac{\partial \alpha(t)}{\partial t} \right| \quad (5.40)$$

Onde,

$$\frac{\partial \alpha(t)}{\partial t} = \lim_{\Delta t \rightarrow 0} \frac{\alpha(t + \Delta t) - \alpha(t)}{\Delta t} \quad (5.41)$$

Sendo α a amplitude do qubit de saída que varia com a dinâmica do sistema.

Para cada configuração de um dado modelo de dinâmica, que possui $|i\rangle$, $|s\rangle$ e $|o\rangle$ como parâmetros iniciais, é possível extrair um δ e um δ_{abs} no final de sua dinâmica.

A fim de comparar os modelos, escolhemos algumas configurações iniciais que os registradores $|i\rangle$, $|s\rangle$ e $|o\rangle$ podem assumir. Os valores possíveis para esses qubits foram $H|0\rangle$, $|0\rangle$ e $|1\rangle$, ou seja, 81 configurações possíveis. Logo, para cada modelo, temos 81 valores de δ e 81 valores de δ_{abs} .

Desses 81 valores, extraímos os valores máximos, mínimos, a média e a variância dessa amostragem e listamos abaixo. Tanto para o δ quanto para o δ_{abs} .

Modelo	Realimentação	Máximo(δ)	Mínimo(δ)	Média(δ)	Variância(δ)
mdsq-RAM	$ o\rangle$ em $ o\rangle$	1.0	-1.0	0	0.179012
mdsq-RAM	$ o\rangle$ em $ o\rangle$ e em $ i\rangle$	1.0	-1.0	-0.401235	0.181603
mdsep-qRAM	$ o\rangle$ em $ o\rangle$	0.5	-0.5	-0.0432099	0.069121
mdsep-qRAM	$ o\rangle$ em $ o\rangle$ e em $ i\rangle$	0.5	-0.5	-0.037037	0.072702
mds- $ \psi\rangle$ -RAM	$ o\rangle$ em $ o\rangle$	0.339847	-0.669108	-0.088288	0.062114
mds- $ \psi\rangle$ -RAM	$ o\rangle$ em $ o\rangle$ e em $ i\rangle$	0	-1.0	-0.408257	0.067565

Tabela 5.1: Tabela dos valores máximos, mínimos, a média e variância das 81 amostras colhidas de δ para cada um dos modelos rodando as 81 configurações

Podemos verificar que os valores de δ , quando a configuração realimenta o qubit de saída no qubit de saída e no de entrada, mostra-se com variância maior que o caso da realimentação

Tabela 5.2: Tabela de resultados dos valores da medida de variação absoluta δ_{abs} para cada um dos modelos

Modelo	Realimentação	Máximo(δ_{abs})	Mínimo(δ_{abs})	Média(δ_{abs})	Variância(δ_{abs})
mdsq-RAM	$ o\rangle$ em $ o\rangle$	99.0	0.0	17.160494	1399.548316
mdsq-RAM	$ o\rangle$ em $ o\rangle$ e em $ i\rangle$	99.0	0.0	6.698724	504.231702
mdsep-qRAM	$ o\rangle$ em $ o\rangle$	1.5	0.0	0.265432	0.198064
mdsep-qRAM	$ o\rangle$ em $ o\rangle$ e em $ i\rangle$	1.5	0.0	0.256323	0.178565
mds- $ \psi\rangle$ -RAM	$ o\rangle$ em $ o\rangle$	41.780787	0.0	8.728907	226.476157
mds- $ \psi\rangle$ -RAM	$ o\rangle$ em $ o\rangle$ e em $ i\rangle$	21.830430	0.0	2.421053	29.071097

apenas na saída. Essa padrão inverte na métrica de variação absoluta δ_{abs} , onde a variância é ordens de grandeza maior para o caso de realimentação apenas na saída. Vale-se mencionar que a configuração com estado puro mantém os valores pequenos de média, máximo e variância, provavelmente por restringir o comportamento do qubit de saída no formato $N(z|0\rangle + |1\rangle)$. Para o caso de variação δ , o modelo de estado puro tem seu range de variação apenas em $[-0.5, 0.5]$.

Ter valores de média sempre negativos fazem acreditar que o comportamento da dinâmica da amplitude é tendenciosamente decrescente (Tabela (5.1)). Pode-se enxergar o quanto as variações de magnitude da amplitude são significativa olhando para a Tabela (5.2), onde vemos que a a variância é alta e que a média da variação não é tão baixa quanto a Tabela (5.1) faz acharmos que é, por conta da compensação do sinal.

CAPÍTULO 6

Conclusão

Através dos modelos de dinâmicas propostos, podemos investigar a sensibilidade dos neurônios quânticos sem peso às condições iniciais, sob a perspectiva do Caos. Os experimentos apresentados, pela visualização gráfica do conjunto Julia e dos gráficos de amplitude, corroboram com o esperado em relação à sensibilidade paramétrica e inicial.

Há muito o que ser feito na continuidade dessa pesquisa, visto a importância de entendimento desses modelos e a extrapolação de uma dinâmica mais robusta e genérica. Pode-se utilizar outras ferramentas de análise de modelos dinâmicos assim como fazer uma análise numérica mais profunda quanto à sensibilidade dessas condições iniciais. As aplicações da não linearidade e de métodos caóticos parecem ser promissores pelos resultados apresentados quando se utiliza mecânica quântica não linear para resolver problemas NP-completos em tempo polinomial [40]. Um trabalho que detecte caos através de redes neurais também é vislumbrado [3].

Referências Bibliográficas

- [1] H. M. Henrique, E. L. Lima e J. C. Pinto *A bifurcation study on neural network models for nonlinear dynamic systems*, Latin American Applied Research, 1998, vol 28, n 3, pags 187-200
- [2] M. Ghil, I. Zaliapin e B. Coluzzi. *Boolean delay equations: A simple way of looking at complex systems*, Physica D: Nonlinear Phenomena, 1998, vol 237, n 23, pags 2967-2986
- [3] A. M. Albano, A. Passamante, T. Hediger e M. E. Farrell *Using neural nets to look for chaos*, Physica D, 1992,n 1-4, pags 1-9
- [4] I. Aleksander, *Self-adaptive universal logic circuits*, Electronics Letters 2 (8) (1966) 321-322
- [5] L.K. Grover, *A fast quantum mechanical algorithm for database search*, in: Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing, STOC '96, ACM, 1996, pp. 212-219.
- [6] Adenilton J. da Silva, Wilson R. de Oliveira, Teresa B. Ludermir *Classical and superposed learning for quantum weightless neural networks* Neurocomputing Journal, 2011, pp. 52-60.
- [7] Daniel S. Abrams, Seth Lloyd *Nonlinear quantum mechanics implies polynomial-time solution for NP-complete and #P problems* Phys.Rev.Lett. 81 (1998) 3992-3995
- [8] M.A. Nielsen, I.L. Chuang *Quantum Computation and Quantum Information*, Cambridge University Press, 2000.
- [9] Paula Neto, F. M. ; Silva, A. J. ; Ludermir, T. B.; De Oliveira, Wilson Rosa. *Analysis of Quantum Neural Models*, 11th Brazilian Congress on Computational Intelligence CBIS, 2013, Ipojuca, PE. BRICS-CCI-CBIC 2013. Disponível em: http://cin.ufpe.br/~fmpn2/files/Analysis_of_Quantum_Neural_Models.pdf. [accessada em 22-Abril-2014].
- [10] Luis A. Aguirre and Christophe Letellier. *Modeling Nonlinear Dynamics and Chaos: A Review*, Mathematical Problems in Engineering, vol. 2009, Article ID 238960, 35 pages, 2009. doi:10.1155/2009/238960.
- [11] H. Poincaré. *Les Méthodes Nouvelles de la Mécanique Céleste*, Gauthier-Villars, Paris, 1892.

- [12] Haykin, Simon; Principe, J. *Making sense of a complex world [chaotic events modeling]*, Signal Processing Magazine, IEEE; vol.15; Issue: 3; Page 66 - 81, ISSN:1053-5888
- [13] Devaney, Robert L. *A first course in chaotic dynamical systems: theory and experiment*, 1948.
- [14] H. Bechmann-Pasquinucci and B. Huttner and N. Gisin, *Non-linear quantum state transformation of spin-1/2*, Physics Letters A, vol. 242, 4-5, Page 198 - 204, 1998. issn 0375-9601, doi [http://dx.doi.org/10.1016/S0375-9601\(98\)00189-3](http://dx.doi.org/10.1016/S0375-9601(98)00189-3).
- [15] T. Kiss, I. Jex, G. Alber, S. Vymetal, *Complex chaos in the conditional dynamics of qubits*, Phys. Rev. A 74, 040301(R) (2006)
- [16] A.J. Silva, W. de Oliveira, T.B. Ludermir, *A weightless neural node based on a probabilistic quantum memory*, in: Neural Networks, 2010. SBRN: 10. 11th Brazilian Symposium on 2010, pp. 259-264.
- [17] B. V. Chirikov et al., Physica D 33, 77 1988; M. V. Berry, in New Trends in Nuclear Collective Dynamics, edited by Y. Abe, H. Horiuchi, and K. Matsuyanagi, Springer Proceedings in Physics Vol. 58 Springer, Berlin, 1992, pp. 183-186; R. Blümel, Phys. Rev. Lett. 73, 428 1994; R. Schack, ibid. 75, 581 1995; R. Blümel, ibid. 75, 582 1995.
- [18] T. B. Ludermir, A. Carvalho, A. P. Braga, and M. C. P. Souto, *Weightless neural models: A review of current and past works*, Neural Computing Surveys, vol. 2, pp. 41-61, 1999.
- [19] I. Aleksander, *Self-adaptive universal logic circuits*, IEE Electronics Letters, 2:231, 1966
- [20] J. J. Hopfield. *Neural networks and physical systems with emergent collective computational properties*, In Proceedings of the National Academy of Sciences (USA), volume 79, pages 2554-2558, 1982
- [21] I. Aleksander. *Emergent intelligent properties of progressively structured pattern recognition nets*, Pattern Recognition Letters, 1:375-384, 1983
- [22] W. K Kan and I. Aleksander. *A probabilistic logic neuron network for associative learning*, In Proceedings of the IEEE First International Conference on Neural Networks (ICNN87), volume II, pages 541-548, San Diego, California, June 1987.
- [23] W. K Kan and I. Aleksander. *Canonical neural nets based on logic nodes*, In Proceedings of the IEE International Conference on Artificial Neural Network, pages 110-114, London, 1989
- [24] W. K Kan and I. Aleksander. *A probabilistic logic neuron network for associative learning*, In Proceedings of the IEEE First International Conference on Neural Networks (ICNN87), volume II, pages 541-548, San Diego, California, June 1987

- [25] C. Myers and I. Aleksander. *Learning algorithms for probabilistic logic nodes*, In Abstracts of the I Annual International Neural Networks Society Meeting (INNS88), page 205, Boston, 1988.
- [26] C. Myers and I. Aleksander. *Output functions for probabilistic logic nodes*, In Proceedings of the IEE International Conference on Neural Networks, pages 310-314, London, 1989.
- [27] J. G. Taylor. *Spontaneous behaviour in neural networks*, Journal of Theoretical Biology, 36:513-528.
- [28] D. Gorse and J. G. Taylor. *On the equivalence properties of noisy neural and probabilistic RAM nets*, Physics Letters A, 131(6):326-332, 1988.
- [29] D. Gorse and J. G. Taylor. *A continuous input RAM-based stochastic neural model*, Neural Networks, 4:657-665, 1991.
- [30] W. de Oliveira, A.J. Silva, T.B. Ludermir, A. Leonel, W. Galindo, J. Pereira. *Quantum logical neural networks*, in: Neural Networks, 2008. SBRN '08. 10th Brazilian Symposium on 2008, pp. 147-152.
- [31] W. de Oliveira. *Quantum RAM based neural networks*, in: European Symposium on Artificial Neural Networks 2009. Proceedings 2009, pp. 331-336.
- [32] A.J. Silva, T.B. Ludermir, W. de Oliveira, *On the universality of quantum logical neural networks*, in: 2012 Brazilian Symposium on Neural Networks.
- [33] C. Trugenberger, *Probabilistic quantum memories*, Phys. Rev. Lett., vol. 87, no. 6, p. 067901, Jul 2001.
- [34] C. Trugenberger, *Quantum pattern recognition*, Quantum Information Processing, vol. 1, pp. 471-493, 2002.
- [35] C. Trugenberger, *Phase transitions in quantum pattern recognition*, Phys. Rev. Lett., vol. 89, no. 27, p. 277903, Dec 2002.
- [36] P. Benioff, *Foundational Aspects of Quantum Computers and Quantum Robots*, Superlattices and Microstructures, 23, 407-417 (1998).
- [37] P. Benioff, *Quantum robots and environments*, Physical Review A 58, 893 (1998)
- [38] S. Lloyd, *Coherent quantum feedback*, Physical Review A 62, 022108/1 (2000)
- [39] P. Cabauy, P. Benioff, *Cyclic networks of quantum gates*, Physical Review A 68, 032315 (2002); DOI:10.1103/PhysRevA.68.032315
- [40] D. Abrams and S. Lloyd, *Nonlinear Quantum Mechanics Implies Polynomial-Time Solution for NP-Complete and #P Problems*, Physical Review Letters 81:992? 3995, (1998)
- [41] N. Weaver, *Mathematical Quantization*, Studies in Advanced Mathematics. Chapman & Hall/CRC, Boca Raton, Florida, 2001.