



# Otimização Quântica Híbrida (QAOA) para Problema Simples Inspirado no Posicionamento de Turbinas Eólicas: Uma Prova de Conceito

VIII  $\langle \text{WECIQ} | \text{WCQ} \rangle$

---

Marcos Vinícius Cândido Henriques<sup>1</sup>

<sup>1</sup>Universidade Federal Rural do Semi-Árido / Campus Angicos

Florianópolis/SC, 8–12 de dezembro de 2025

# Richard P. Feynman (1982)



“É possível fazer isso [simular efeitos quântico-mecânicos] com um novo tipo de computador — um computador quântico? ... Não é uma máquina de Turing, mas uma máquina de um tipo diferente.”

Richard P. Feynman, Simulating Physics with Computers, International Journal of Theoretical Physics, Vol. 21,

Nos. 6/7, 1982.

# Sumário

Breve Revisão da Computação Quântica

Motivação e objetivos

Fundamentos

Metodologia

Resultados

Conclusões e próximos passos

Agradecimentos e referências

# Bits clássicos vs qubits

## Bits (clássicos)

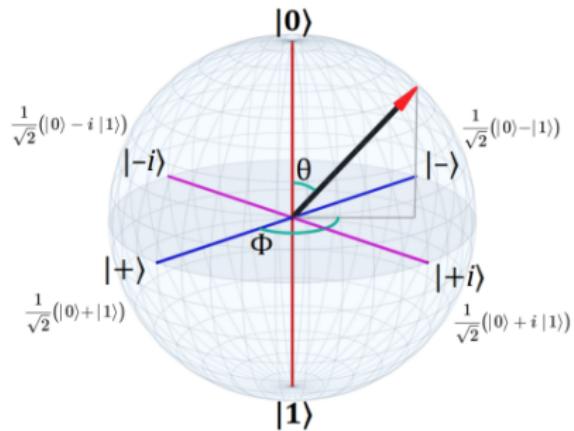
- ▶ Estados discretos: 0 *ou* 1.
- ▶ Informação determinística; podem ser copiados livremente.
- ▶ Operações lógicas (AND, OR, NOT) sobre valores binários.

## Qubits (quânticos)

- ▶ Estados base:  $|0\rangle$ ,  $|1\rangle$ .
- ▶ Superposição:  $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ , com  $|\alpha|^2 + |\beta|^2 = 1$ .
- ▶ Medida: colapsa para 0 ou 1 com probabilidades  $|\alpha|^2$  e  $|\beta|^2$ .
- ▶ Fase e interferência; **emaranhamento** entre múltiplos qubits.

# Geometria e escalabilidade

- ▶ Parametrização na *Esfera de Bloch*:  
 $|\psi\rangle = \cos \frac{\theta}{2} |0\rangle + e^{i\varphi} \sin \frac{\theta}{2} |1\rangle.$
- ▶  $n$  qubits codificam amplitudes sobre  $2^n$  estados; bits clássicos guardam apenas um estado por vez.
- ▶ Ganhos vêm de interferência e correlações quânticas (não de “testar tudo” explicitamente).



Esfera de Bloch: representação de 1 qubit

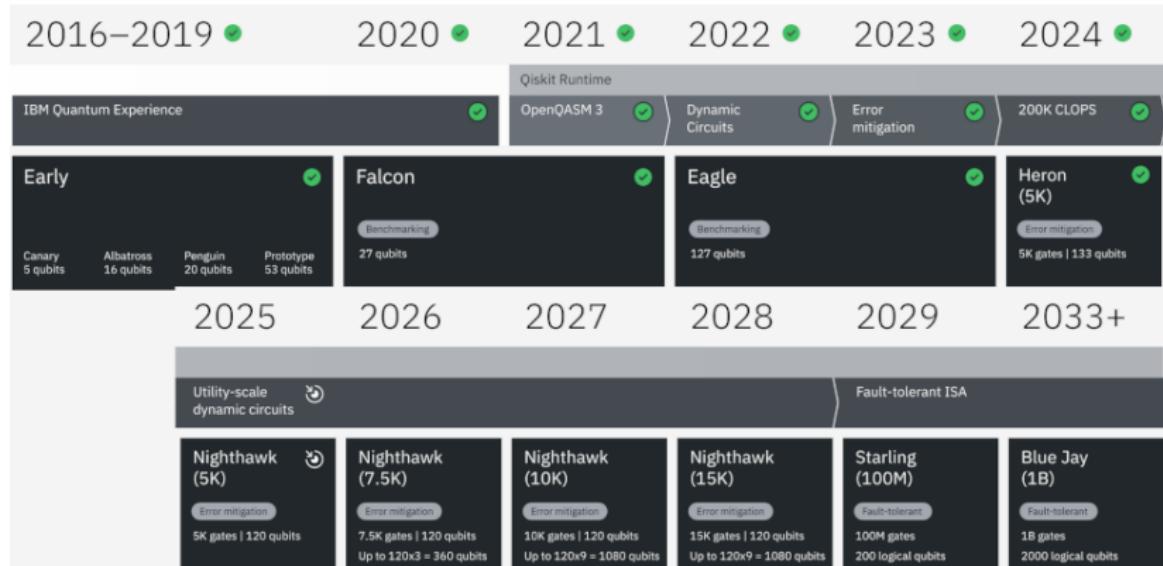
# Simulação clássica: recursos vs qubits (AWS)

Quão rápido explodem os recursos necessários para simular, em computadores clássicos, circuitos com dezenas de qubits?

Número de qubits	Memória necessária (GiB)	Número de instâncias	Número total de núcleos
36	2.199	16	576
37	4.398	32	1.152
38	8.796	64	2.304
39	17.592	128	4.608
40	35.184	256	9.216
41	70.369	512	18.432
42	140.737	1.024	36.864
43	281.475	2.048	73.728
44	562.950	4.096	147.456

Simulating 44-Qubit quantum circuits using AWS ParallelCluster (Dr. Fabio Baruffa; Pavel Lougovski), 30 AUG 2022, AWS Center for Quantum Computing, AWS.

# Roadmap IBM Quantum



Roadmap de hardware IBM Quantum — Crédito: IBM.

# Benchmarks em Otimização Quântica (KOCH et al., 2025)

Koch et al. (2025) propõem dez classes de problemas de otimização difíceis para algoritmos clássicos e ligados a aplicações práticas, visando benchmarks *sistemáticos, justos e comparáveis* para métodos de otimização quântica. Os autores fazem parte do *Quantum Optimization Working Group*, liderado pelo IBM Quantum.

Nº	Problema	Descrição
1	Market	<i>Split multi-dimensional subset-sum</i> : particionamento de itens com múltiplas dimensões/restricções.
2	LABS	Sequências binárias de baixa autocorrelação (minimiza picos de autocorrelação).
3	Birkhoff	Decomposição de Birkhoff mínima (combinação de permutações para matrizes duplamente estocásticas).
4	Steiner	Empacotamento de árvores de Steiner em grafos (aplicações em VLSI/roteamento).
5	Sports	Agendamento de torneios esportivos (STS) com restrições e balanceamento.
6	Portfolio	Otimização de portfólio multi-período com custos de transação.
7	Independent Set	Máximo conjunto independente não ponderado (MIS) em grafos.
8	Network	Projeto de redes de comunicação sob custos/capacidades e confiabilidade.
9	Routing	Roteamento de veículos com capacidade (CVRP).
10	Topology	Projeto de topologia de grafos (problema Nô-Grau-Diâmetro).

KOCH, Thorsten et al. Quantum Optimization Benchmark Library—The Intractable Decathlon. arXiv preprint arXiv:2504.03832, 2025.

# Contexto e motivação

- ▶ Crescente participação da energia eólica no RN e no Brasil.
- ▶ Efeito de esteira (wake): redução da potência e aumento da turbulência a jusante.
- ▶ Decisão de posicionamento das turbinas impacta diretamente **fator de capacidade** e **LCOE**.
  - ▶ **Fator de capacidade:** energia gerada / energia a 100% da potência nominal.
  - ▶ **LCOE:** custo nivelado por MWh ao longo da vida do projeto.



Turbinas da Vattenfall no Mar do Norte (2011). Crédito: NOAA.

## Modelo de esteira

- ▶ **Padrão (Jensen–Katic)**: déficit de velocidade decai com a distância a jusante e com a expansão da esteira. Forma típica:

$$1 - \frac{u}{u_0} = \frac{2a}{(1 + kx/r_0)^2}$$

onde  $u_0$ : velocidade a montante (inflow),  $u$ : velocidade a  $x$  a jusante,  $a$ : fator de indução axial,  $k$ : coeficiente de decaimento da esteira,  $r_0$ : raio do rotor,  $x$ : distância a jusante.

- ▶ **Neste trabalho (prova de conceito)**: adotamos um **modelo simples** no qual a potência disponível **decai linearmente** com a distância a partir da turbina a montante (até um alcance fixo). Quando esteiras se sobrepõem, as penalidades são somadas.
- ▶ Objetivo: ter um **cenário didático** para demonstrar o QAOA híbrido, com penalidades fáceis de calibrar e execução rápida.

## Objetivos do trabalho

- ▶ Formular o problema de layout como otimização combinatória.
- ▶ Aplicar o **QAOA** para maximizar geração com penalidades por interferência.
- ▶ Avaliar cenários simples (3x3, 4x4) e discutir escalabilidade.

# O que é o QAOA? (1/2)

- QAOA = *Quantum Approximate Optimization Algorithm* (Algoritmo Quântico de Otimização Aproximada).

O QAOA é inspirado no Teorema Adiabático.

## Teorema adiabático de Born e Fock (1928)

Um sistema físico permanece em seu autovetor instantâneo se uma perturbação atuar sobre ele **lentamente** o suficiente e se existir uma **lacuna (gap)** entre o correspondente autovalor e o restante do espectro do Hamiltoniano.<sup>1</sup>

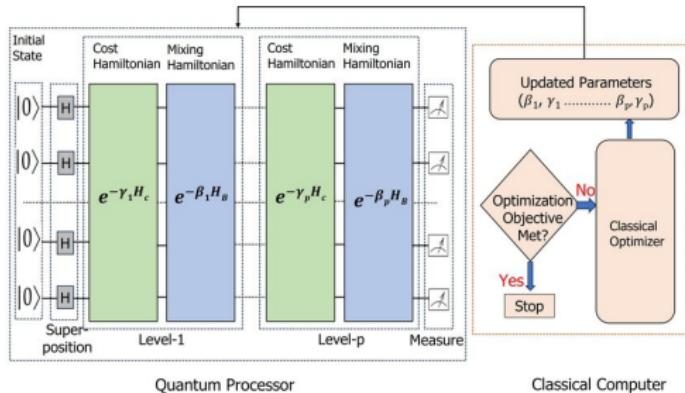
## Consequência direta do teorema adiabático

Se começamos no **estado fundamental** de um Hamiltoniano dependente do tempo e o Hamiltoniano evolui **lentamente** o suficiente, e dado tempo suficiente, o estado final será o **estado fundamental** do Hamiltoniano final.

<sup>1</sup>Born, M.; Fock, V. A. (1928). "Beweis des Adiabatensatzes". Zeitschrift für Physik A. 51 (3–4): 165–180.

## O que é o QAOA? (2/2)

- ▶ Algoritmo variacional para **otimização combinatória** em hardware quântico ruidoso (NISQ).
- ▶ Estado parametrizado: alterna camadas de **custo** ( $H_C$ ) e **mistura** ( $H_M$ ) com profundidade  $p$ .
- ▶ **Híbrido**: os parâmetros ( $\gamma, \beta$ ) são **otimizados em computador clássico**; o circuito é avaliado em dispositivo/simulador quântico.
- ▶ As medidas retornam *bitstrings* candidatos; escolhemos o(s) com melhor valor de custo.



## Modelo de esteiras e grafo de conflitos

- ▶ Penalidade por pares de turbinas com **interferência** acima de limiar (matriz de interferência).
- ▶ Mapeamento para um grafo: vértices = posições; arestas = penalidades/esteiras.
- ▶ Função custo: termo de **benefício** por turbina ativa e **penalidade** por conflitos.

Qubits: estados representando ausência ou presença de turbina

$|0\rangle$



$|1\rangle$



$$|+\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$$



# Termo de 1-qubit (recompensa)

Definição do termo por posição  $i$ :

$$H_i^{(1)} = -\frac{s_i}{2} (I - Z_i)$$

- ▶  $I$ : matriz identidade.
- ▶  $Z_i$ : operador de Pauli-Z no qubit  $i$ .
- ▶  $s_i$ : benefício/*score* associado à posição  $i$ .
- ▶ Ação de  $Z$  na base computacional:
  - ▶  $Z|0\rangle = |0\rangle$  (*autovalor +1*)
  - ▶  $Z|1\rangle = -|1\rangle$  (*autovalor -1*)

Contribuições de energia do termo  $H_i^{(1)}$ :

- ▶  $E(|0\rangle) = 0$
- ▶  $E(|1\rangle) = -s_i$ 
  - ▶ estado  $|1\rangle$  é **recompensado**



# Termo de 2-qubits (esteira em $|11\rangle$ )

Definição do termo para o par  $(i,j)$ :

$$\begin{aligned} H_{ij}^{(2)} &= \frac{w_{ij}}{4} (Z_i Z_j - Z_i - Z_j + I) \\ &\equiv \frac{w_{ij}}{4} (I - Z_i)(I - Z_j) \end{aligned}$$

- ▶  $I$ : matriz identidade.
- ▶  $Z_i, Z_j$ : operadores de Pauli-Z nos qubits  $i$  e  $j$ .
- ▶  $Z_i Z_j$ : produto tensorial  $Z \otimes Z$  agindo no par  $(i,j)$ .
- ▶  $w_{ij}$ : penalidade de esteira para o par  $(i,j) \in \mathcal{W}$ .

Contribuições de energia do termo  $H_{ij}^{(2)}$  por estado do par  $(i,j)$ :

- ▶  $E(|00\rangle) = E(|01\rangle) = E(|10\rangle) = 0$
- ▶  $E(|11\rangle) = w_{ij}$ 
  - ▶ penaliza apenas o estado  $|11\rangle$ .



# Hamiltoniano de custo total

Soma dos termos de 1-qubit e 2-qubits construídos no código:

$$H_C = \sum_{i \in V} \frac{s_i}{2} (Z_i - I) + \sum_{(i,j) \in \mathcal{W}} \frac{w_{ij}}{4} (Z_i Z_j - Z_i - Z_j + I)$$

- ▶  $I$ : identidade.
- ▶  $Z_i, Z_j$ : operadores de Pauli-Z nos qubits  $i$  e  $j$ .
- ▶  $Z_i Z_j = Z \otimes Z$  é o produto tensorial no par  $(i, j)$ .
- ▶  $s_i$ : benefício/score da turbina na posição  $i$ .
- ▶  $w_{ij}$ : penalidade de esteira do par  $(i, j) \in \mathcal{W}$ .
- ▶  $V$ : conjunto de posições (vértices);  
 $\mathcal{W}$ : arestas com wake.

Formulação QUBO (em Pauli-Z):

$$H_C = \sum_{i < j} Q_{ij} Z_i Z_j + \sum_i b_i Z_i$$

onde

$$Q_{ij} = \frac{w_{ij}}{4}$$

$$b_i = \frac{s_i}{2} - \frac{1}{4} \sum_{(i,j) \in \mathcal{W}} w_{ij}$$

# Código (Python + Qiskit): Hamiltoniano de custo

Exemplo em Python utilizando a biblioteca Qiskit.

```
def create_cost_hamiltonian():
    pauli_list = []
    const_offset = 0.0

    for i in range(optimizer.n_positions):
        pauli_list.append(("Z", [i], score[i]/2))
        const_offset += -score[i]/2

    for (i, j), wake_penalty in wake_penalties.items():
        pauli_list.append(("ZZ", [i, j], wake_penalty/4))
        pauli_list.append(("Z", [i], -wake_penalty/4))
        pauli_list.append(("Z", [j], -wake_penalty/4))
        const_offset += wake_penalty/4

    if abs(const_offset) > 0:
        pauli_list.append(("I", [], const_offset))

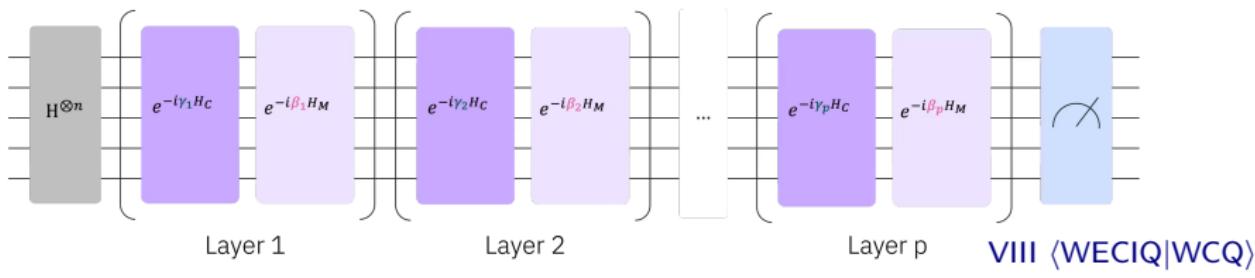
    return SparsePauliOp.from_s sparse_list
    (pauli_list, num_qubits=optimizer.n_positions)
```

# Hamiltoniano $\Rightarrow$ circuito quântico

- ▶ O Hamiltoniano  $H_C$  traz a definição quântica do problema; a partir dele, construímos um **círcuito** para amostrar boas soluções.
- ▶ QAOA inspira-se em *quantum annealing* e aplica **camadas alternadas** de operadores no circuito.
- ▶ Iniciar no estado  $H^{\otimes n}|0\rangle$  e conduzir o sistema ao estado fundamental de  $H_C$  aplicando  $e^{-i\gamma_k H_C}$  e  $e^{-i\beta_k H_M}$ , com  $\gamma_1, \dots, \gamma_p$  e  $\beta_1, \dots, \beta_p$ .
- ▶  $H_M$ : Hamiltoniano de **mistura (mixer)**; gera exploração entre configurações.

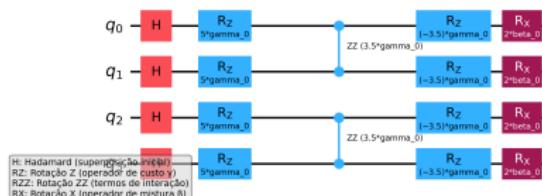
$$H_M = \sum_{i \in V} X_i$$

- ▶ O circuito é **parametrizado** por  $\{\gamma_i, \beta_i\}$ ; testamos valores diferentes e amostramos o estado resultante.

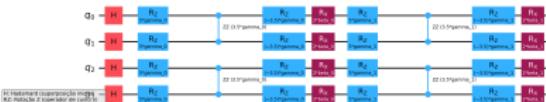


# Circuitos quânticos (QAOA)

**Círculo Quântico QAOA - 2x2**  
4 qubits | 1 camadas | 2 parâmetros

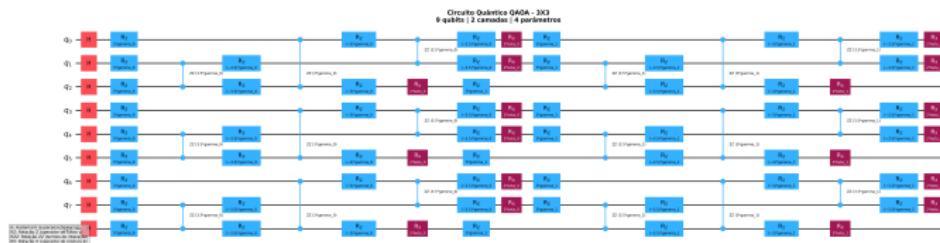


**Círculo Quântico QAOA - 2x2**  
4 qubits | 2 camadas | 4 parâmetros



2x2, 4 qubits, 2 camadas

2x2, 4 qubits, 1 camada



3x3, 9 qubits, 2 camadas

- ▶ **Grades avaliadas:** 3x3, 4x4.
- ▶ **Parâmetros:** Profundidade  $p$ , step size do COBYLA (rhobeg) e range de  $\gamma$  e  $\beta$ .
- ▶ **Métricas:** energia total aproximada, número de conflitos e custo final.
- ▶ **Simulações:** combinações de  $p \in \{1, 2, 3\}$  e rhobeg  $\in \{0.3, 0.5, 0.7\}$ .
- ▶ **Recompensa** (score) por turbina instalada: **5** (3x3 e 4x4).
- ▶ **Penalização de esteira** (ajustada por dimensão):
  - ▶ Grid 3x3: base\_penalty=8.0, distance\_decay=1.0
  - ▶ Grid 4x4: base\_penalty=12.0, distance\_decay=4.0

# Estratégia de Primitivas Qiskit no QAOA

## Diferenciação das Primitivas

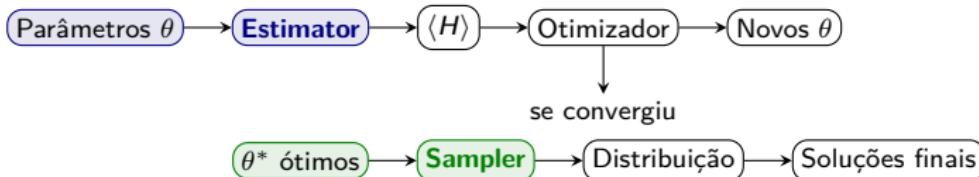
- ▶ **Estimator**: calcula valores esperados de observáveis  $\Rightarrow$  retorna **números reais**.
- ▶ **Sampler**: amostra estados quânticos  $\Rightarrow$  retorna **distribuições de bitstrings**.

## Durante Loop de Otimização (função de custo)

- ▶ APENAS Estimator
- ▶ Calcula  $\langle \psi | H_{\text{cost}} | \psi \rangle$
- ▶ Otimizador clássico recebe valor escalar

## Após Convergência (análise de resultados)

- ▶ Apenas Sampler com parâmetros ótimos
- ▶ **Shots**: n<sup>o</sup> de amostras por execução (ex.: 1024, 2048, 4096, 8192)
- ▶ Probabilidades = contagens/shots; mais shots  $\rightarrow$  menor variância
- ▶ Analisar bitstrings mais prováveis e custo clássico



# Configuração inicial (superposição no QAOA)

- Todas as soluções possíveis em **superposição** uniforme.
- Cada posição do grid é mapeada a um **qubit**.
- $n$  **qubits** podem representar  $2^n$  estados.
  - $3 \times 3$  ( $n = 9$ ):  $2^9 = 512$  estados
  - $4 \times 4$  ( $n = 16$ ):  
 $2^{16} = 65\,536$  estados
- Qubit de cada posição  $i$ :

$$|+\rangle = \frac{1}{\sqrt{2}} |0\rangle + \frac{1}{\sqrt{2}} |1\rangle$$

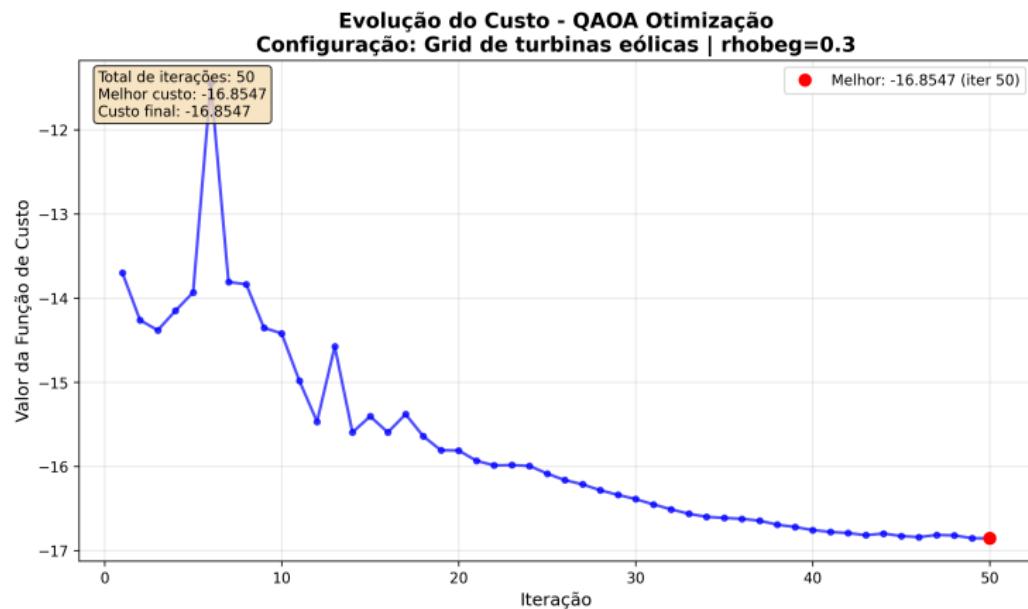
Estado inicial do sistema:

$$\begin{aligned} |\psi_0\rangle &= |+\rangle^{\otimes n} \\ &= \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle \end{aligned}$$



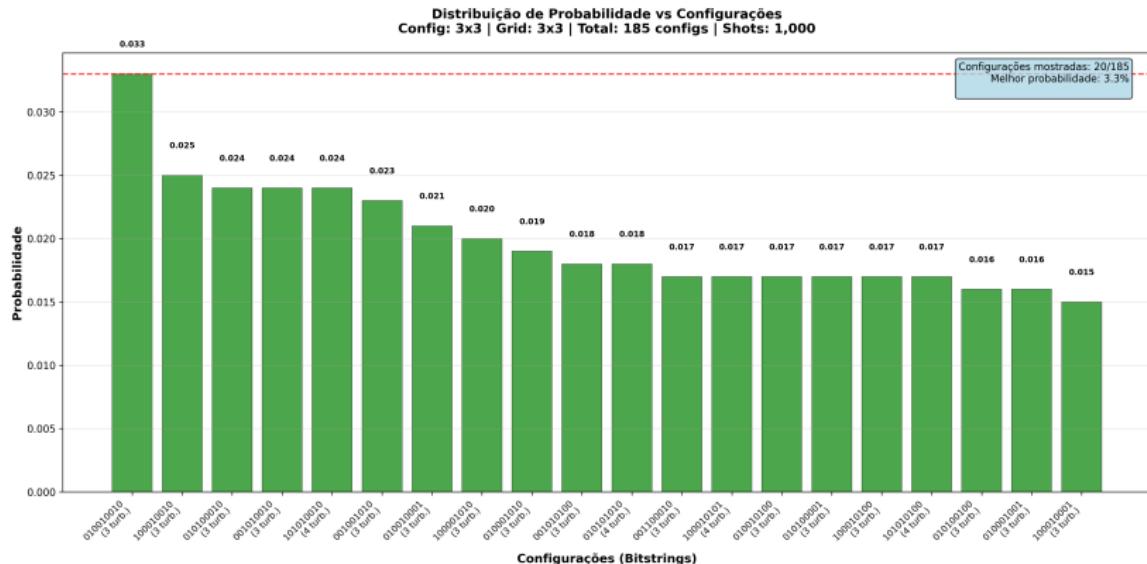
Estado inicial em superposição (4x4).

# Evolução do custo nas iterações

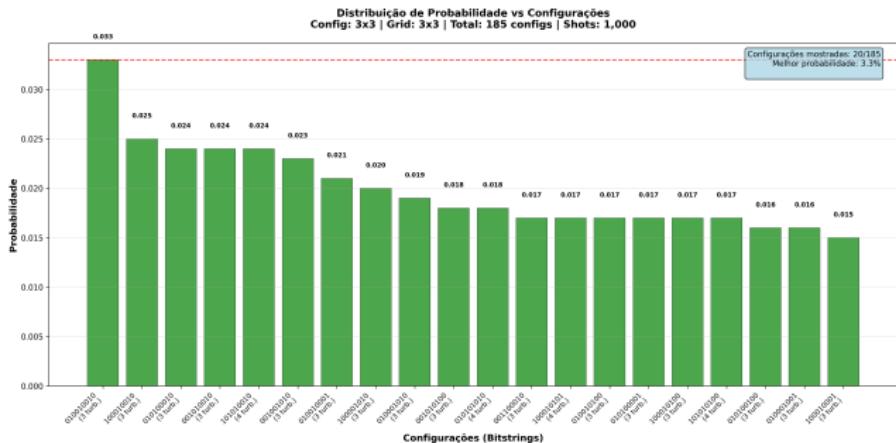


Exemplo no **grid 3x3**: trajetória do custo ao longo das iterações do otimizador.

# Distribuição de probabilidades (3x3)



# Distribuição de probabilidades (3x3)



3,3%



$|0100010010\rangle$

2,5%



$|100010010\rangle$

2,4%



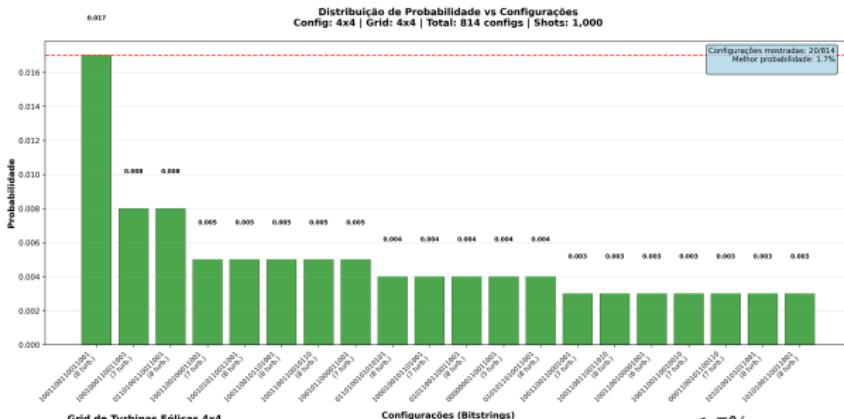
$|010100010\rangle$

2,4%



$|001010010\rangle$

## Distribuição de probabilidades (4x4) e layouts



**Grid de Turbinas Eólicas 4x4**  
Direção do Vento: Oeste → Leste | Probabilidade: 1.7%  
Score Total: 40.0 - Penalidade: 0.0 = Score Líquido: 40.0

			<b>VENTO</b>	
Row 0	0	1	2	
Row 1	4	5	6	
Row 2	8	9	10	
Row 3	12	13	14	
	Col 0	Col 1	Col 2	
	Turbinas Instaladas	●	Posição Vazia	○



## Layout otimizado (4x4) — execução

$$|1001100110011001\rangle$$

# Layout otimizado (4x4) e custos



$|1001100110011001\rangle$

# Resultados QAOA — Simulação em CPU (Grid 2x3, 6 qubits)

Camadas	rhobeg (COBYLA)	Prob. Média (%)	Prob. Máx (%)	Score Médio	Score Máximo	Tempo (s)
1	0.3	32.7	71.0	18.3	20.0	0.62
1	0.5	15.7	21.0	18.0	18.0	0.62
1	0.7	11.0	12.0	12.0	17.0	0.61
2	0.3	68.3	70.0	20.0	20.0	0.72
2	0.5	39.0	96.0	19.7	20.0	0.82
2	0.7	39.7	67.0	19.3	20.0	0.74
3	0.3	56.3	96.0	19.7	20.0	0.87
3	0.5	86.7	93.0	20.0	20.0	0.85
3	0.7	52.3	65.0	20.0	20.0	0.81

Análise por camadas e parâmetro rhobeg (COBYLA).

# Resultados QAOA — Simulação em CPU (Grid 3x3, 9 qubits)

Camadas	rhobeg (COBYLA)	Prob. Média (%)	Prob. Máx (%)	Score Médio	Score Máximo	Tempo (s)
1	0.3	4.3	5.0	10.0	11.0	0.70
1	0.5	4.3	5.0	10.0	11.0	0.71
1	0.7	4.3	5.0	11.7	15.0	0.67
2	0.3	13.3	18.0	13.0	15.0	0.85
2	0.5	10.0	18.0	13.0	15.0	0.85
2	0.7	4.3	5.0	12.7	15.0	0.85
3	0.3	12.0	13.0	13.0	15.0	0.95
3	0.5	13.0	14.0	13.0	15.0	0.95
3	0.7	5.0	7.0	13.7	15.0	0.97

Análise por camadas e parâmetro rhobeg (COBYLA).

# Resultados QAOA — Simulação em CPU (Grid 4x4, 16 qubits)

Camadas	rhobeg (COBYLA)	Prob. Média (%)	Prob. Máx (%)	Score Médio	Score Máximo	Tempo (s)
1	0.3	2.0	2.0	20.7	30.0	3.28
1	0.5	1.7	2.0	16.0	20.0	1.74
1	0.7	1.7	2.0	16.7	17.0	1.48
2	0.3	2.0	2.0	30.0	36.0	7.35
2	0.5	1.3	2.0	6.3	13.0	14.29
2	0.7	2.7	4.0	21.3	31.0	4.40
3	0.3	3.7	7.0	35.3	40.0	14.82
3	0.5	32.3	92.0	34.7	40.0	12.88
3	0.7	2.3	3.0	26.3	40.0	13.35

Resultados por **camadas** e **rhobeg** (COBYLA). Tempos em Intel Core i3-10100.

## Comparação de performance (Tabela IV — Artigo)

Grade	Qubits	Score Máximo	Camadas Ótimas	rhobeg Ótimo	Prob. Máx (%)	Turbinas Médias	Tempo Médio (s)
2x3	6	20.0	1	0.3	71.0	4.7	0.62
3x3	9	15.0	1	0.7	5.0	4.0	0.67
4x4	16	40.0	3	0.3	7.0	7.3	14.82

Melhores resultados por grade conforme a Tabela IV do artigo.

# Execução na IBM Quantum Platform

Marcos Candido Henriques

## IBM Quantum Platform

Help us improve Qiskit and IBM Quantum Platform! Share your thoughts in our [annual feedback survey](#) ↗

**API key** [Create +](#) | [View all ↗](#)

**My recent workloads** [View all](#)

ID	Status	Instance	Created	QPU	Usage
d2ua3g9utc7s73...	Pending	meu_primeiro_computador_quan...	Sep 6, 2025	ibm_torino	-
d2ua39rok8rs73...	Completed	meu_primeiro_computador_quan...	Sep 6, 2025	ibm_torino	11s
d2ua33s7sg0c73...	Completed	meu_primeiro_computador_quan...	Sep 6, 2025	ibm_torino	11s
d2ua2ts7se0c73...	Completed	meu_primeiro_computador_quan...	Sep 6, 2025	ibm_torino	11s

# Dificuldades em grids 5x5

## ► Topologia do hardware da IBM

- ▶ Cada qubit conecta-se diretamente a no máximo 3 vizinhos.
- ▶ Ligações adicionais exigem operações *SWAP*, que aumentam erros quânticos.

## ► Limitações na simulação clássica

- ▶ Sistemas com 25 qubits já demandam alto poder computacional.
- ▶ O custo de memória e processamento cresce de forma exponencial.

## ► Complexidade do circuito

- ▶ Quanto maior a profundidade do circuito, mais portas quânticas são aplicadas.
- ▶ O acúmulo de erros de porta e decoerência reduz a fidelidade dos resultados.

# Conclusões

- QAOA viabiliza formulação híbrida para o layout de turbinas com esteiras simplificadas.

# Trabalhos futuros

- ▶ Escalar para ***grids* maiores**.
- ▶ Refinar **modelo de esteiras** e calibração de penalidades.
- ▶ Definir **score heterogêneo** de produção por local.
- ▶ Testar outros otimizadores além do **COBYLA** (ex.: **SPSA**, **Nelder–Mead**, **L-BFGS-B**, **SLSQP**, **ADAM**).
- ▶ Investigar, por meio de **extensos benchmarkings**:
  - ▶ Efeito de diferentes *ranges* dos parâmetros  $\gamma$  e  $\beta$  na busca pela solução.
  - ▶ Buscar número de camadas adequado para cada configuração de problema.
- ▶ Pós-processamento para “**limpeza**” de soluções imperfeitas.
- ▶ Estudo de **ruído** e execução em **hardware quântico**.
- ▶ Testar hamiltonianos de mistura mais complexos que  $H_M = \sum_i X_i$ .
- ▶ Técnicas de *warm-start* e *layerwise* (otimização camada por camada).

# Referências

-  R. J. Barthelmie et al., "Modelling and measuring flow and wind turbine wakes in large wind farms offshore," *Wind Energy*, 12(5):431–444, 2009.
-  H. Kagemoto, "Possible application of quantum computing in the field of ocean engineering: optimization of an offshore wind farm layout with the Ising model," *Journal of Ocean Engineering and Marine Energy*, 10:773–782, 2024.
-  G. Kochenberger et al., "The unconstrained binary quadratic programming problem: a survey," *Journal of Combinatorial Optimization*, 28(1):58–81, 2014.
-  F. Glover, G. Kochenberger, Y. Du, "Quantum bridge analytics I: a tutorial on formulating and using QUBO models," *4OR*, 17(4):335–371, 2019.
-  E. Farhi, J. Goldstone, S. Gutmann, "*A Quantum Approximate Optimization Algorithm*," arXiv:1411.4028, 2014.
-  Qiskit Community, "Qiskit: An Open-Source Framework for Quantum Computing," 2017. DOI: 10.5281/zenodo.2562110. Disponível em: <https://github.com/Qiskit/qiskit> (acesso em agosto de 2025).
-  J. Preskill, "Quantum computing in the NISQ era and beyond," *Quantum*, 2:79, 2018.

# Agradecimentos, contato e repositório

## Agradecimentos

- ▶ Apoio institucional (UFERSA).
- ▶ Comunidade Qiskit e contribuidores do projeto.
- ▶ Organizadores do VIII WECIQ/WCQ 2025.

## Contato e repositório

- ▶ Re却tório: [https://github.com/vinirn/qaoa\\_wind](https://github.com/vinirn/qaoa_wind)
- ▶ Contato:  
[viniciuscandido@ufersa.edu.br](mailto:viniciuscandido@ufersa.edu.br)

Obrigado!