

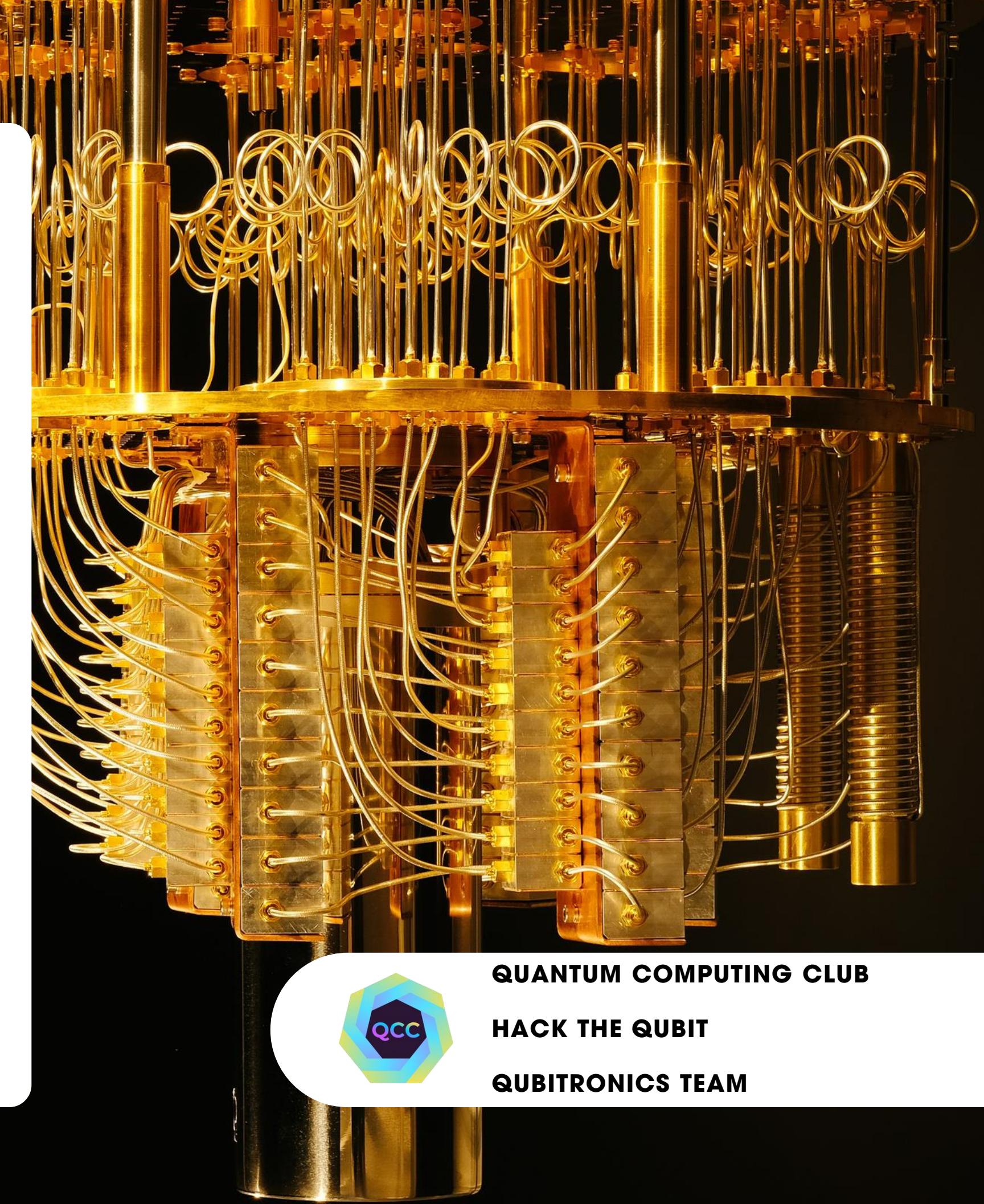
IMPLEMENTACIÓN DE UN ALGORITMO VARIACIONAL CUÁNTICO Y EVOLUCIÓN TEMPORAL TROTTERIZADA CON CORRECCIÓN DE ERRORES EN QISKIT PARA EL MODELO DE ISING

Por:

**José Gabriel Castillo
García
Gael Medina
Emilio Salcedo López**

Fecha:

22 de junio de 2025



QUANTUM COMPUTING CLUB

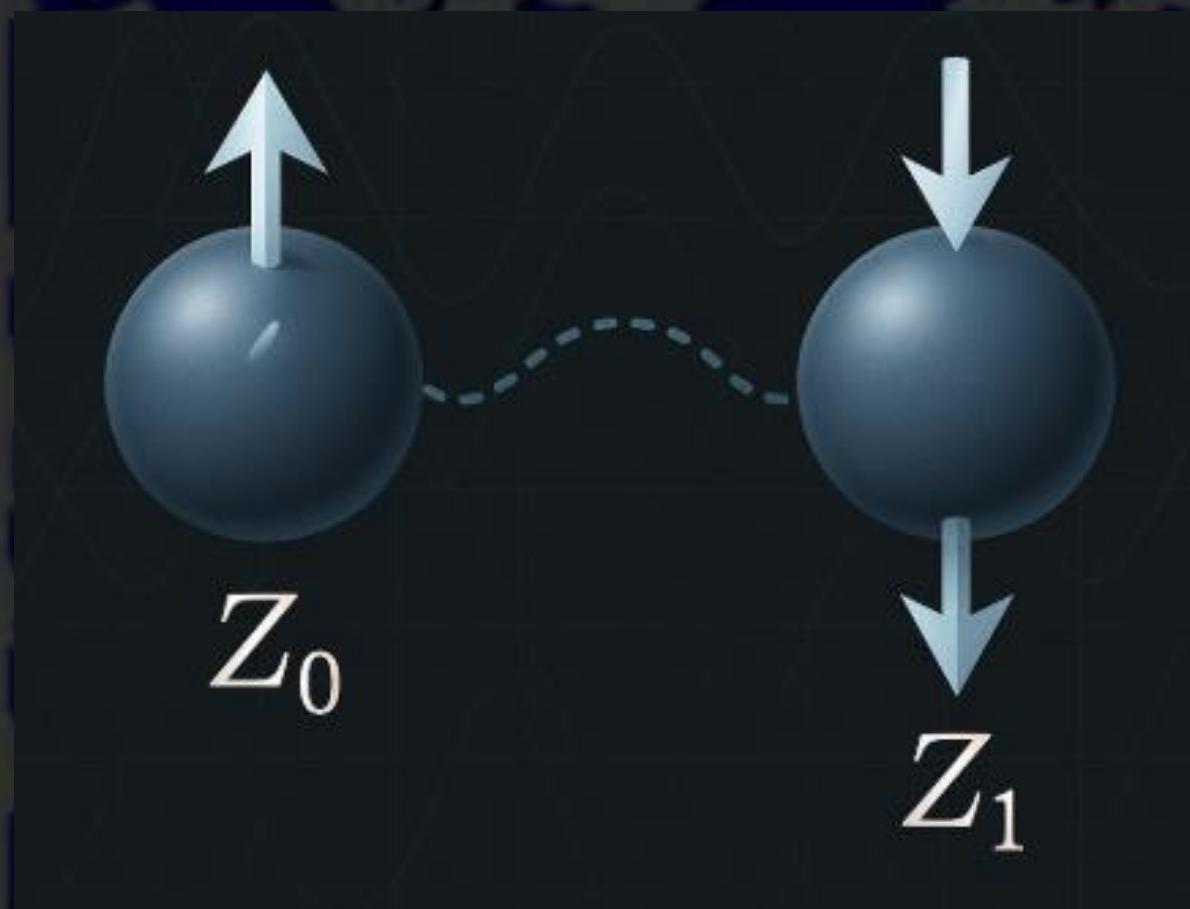
HACK THE QUBIT

QUBITRONICS TEAM

DESCRIPCIÓN DEL CIRCUITO

Simulamos el modelo de Ising en 1D con dos qubits y un campo transversal, utilizando Qiskit. El objetivo fue estimar la energía del estado fundamental mediante un algoritmo variacional (VQE), simular la evolución temporal mediante trotterización, y analizar los resultados obtenidos en simuladores y hardware real. Se utilizó el ansatz EfficientSU2 y se aplicó mitigación de errores de lectura (mthree.M3Mitigation).

ISING Es un modelo físico que describe dos partículas con espines cuánticos (qubits) que:
Interaccionan entre sí (por ejemplo, se alinean o anti-alinean como imanes)
Están sometidas a un campo transversal (que tiende a girar sus espines).





SOLUCIÓN ANALÍTICA ESPERADA

Hamiltoniano del Modelo de Ising

↑	↑	↓	↓	↑
↑	↑	↓	↓	↓
↑	↑	↑	↓	↑
↑	↓	↑	↑	↑
↑	↑	↑	↓	↓

$$H = -J \sum_i \sigma_i \sigma_{i+1} - h \sum_j \sigma_j$$

$$\hat{H} = -J \sum_i Z_i Z_{i+1} - h \sum_j X_j$$

Hamiltoniano del Modelo de Ising

$$\hat{H} = -JZ_0Z_1 - hX_0 - hX_1$$

Hamiltoniano del Modelo de Ising

$$Z \otimes Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \otimes \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

$$X \otimes I = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \otimes \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

$$I \otimes X = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \otimes \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

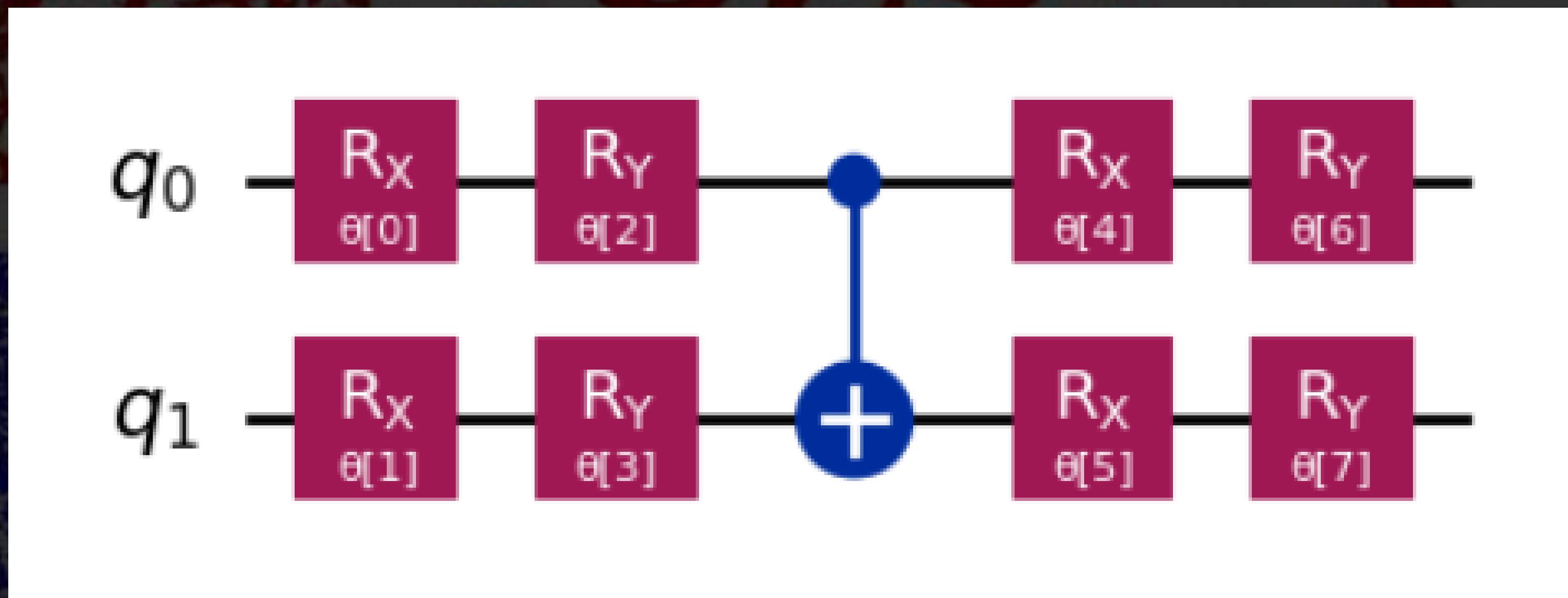
$$\det(\lambda I - H) = 0$$

$$\lambda_0 = -\sqrt{2}$$

$$J = 1 \quad h = 0.5$$

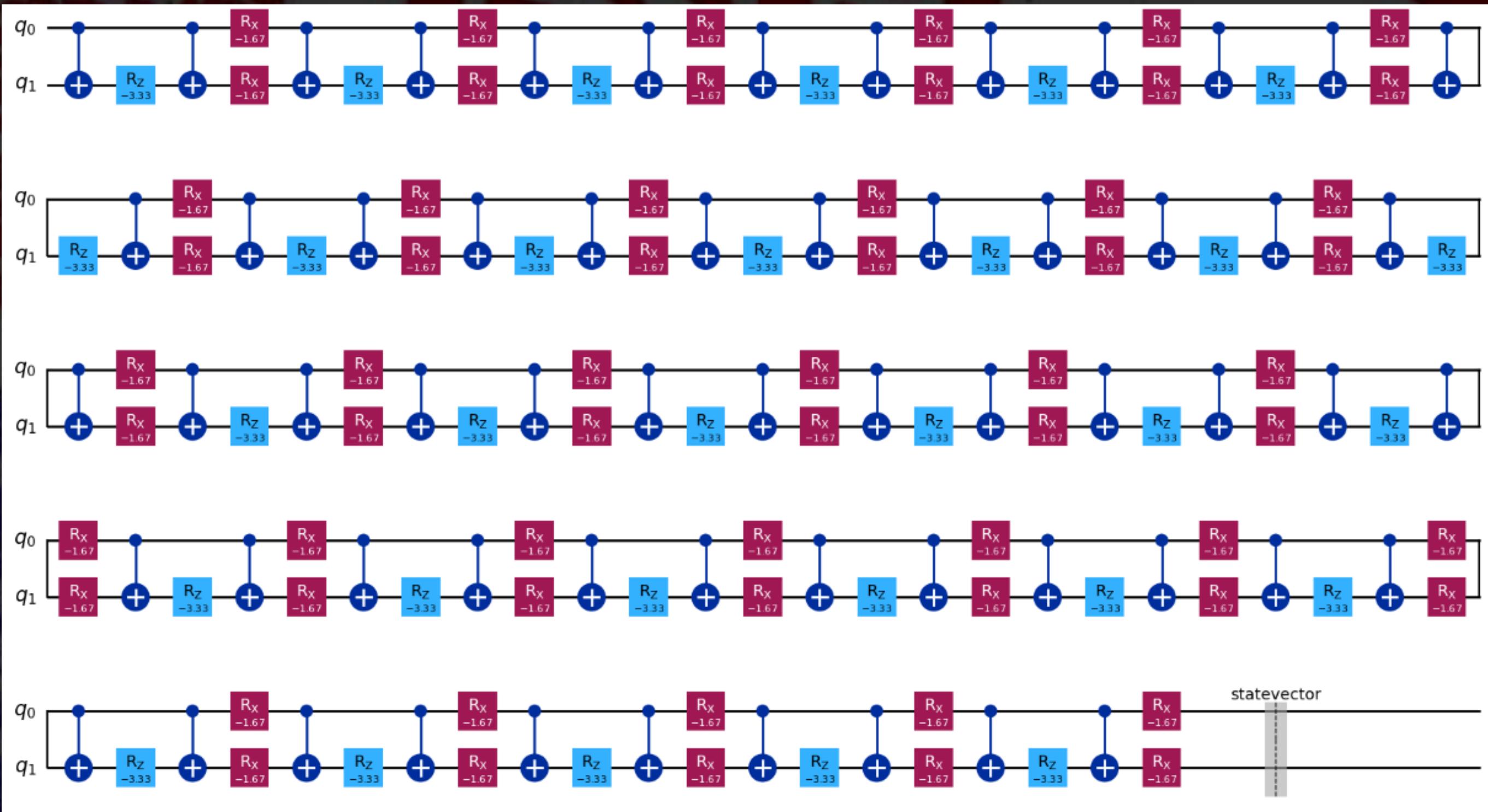
IMPLEMENTACIÓN DEL CIRCUITO CUÁNTICO

Representación VQA con ANZATS



Se utilizó un circuito variacional con el ansatz EfficientSU2 sobre 2 qubits, parametrizado por un ParameterVector. La optimización se realizó con COBYLA, minimizando el valor esperado del Hamiltoniano.

Evolución temporal del sistema desde $|00\rangle$



Se simula la evolución del sistema bajo el Hamiltoniano usando trotterización. La evolución es aproximada en pequeños pasos de tiempo (Trotter-Suzuki).

MODELO DE QEM PROPUESTO

MITIGADOR DE READ-OUT M3

```
import
mthree
mitigator =
mthree.M3Mit
igation(sim_n
oisy)
```

1

Objetivo

M3 busca corregir los errores de medición (readout errors) que ocurren cuando el sistema cuántico registra incorrectamente el estado final de los qubits.

2

Calibración

Se ejecutan circuitos especiales que preparan y miden todos los estados base posibles ($|00\rangle, |01\rangle, \dots |00\rangle, |01\rangle, \dots$) para construir una matriz de confusión, que muestra qué tan seguido se confunde un estado con otro al medir..

3

Aplicación

Cuando se ejecuta un circuito cuántico real, se obtienen conteos de medición (por ejemplo, cuántas veces salió "01").

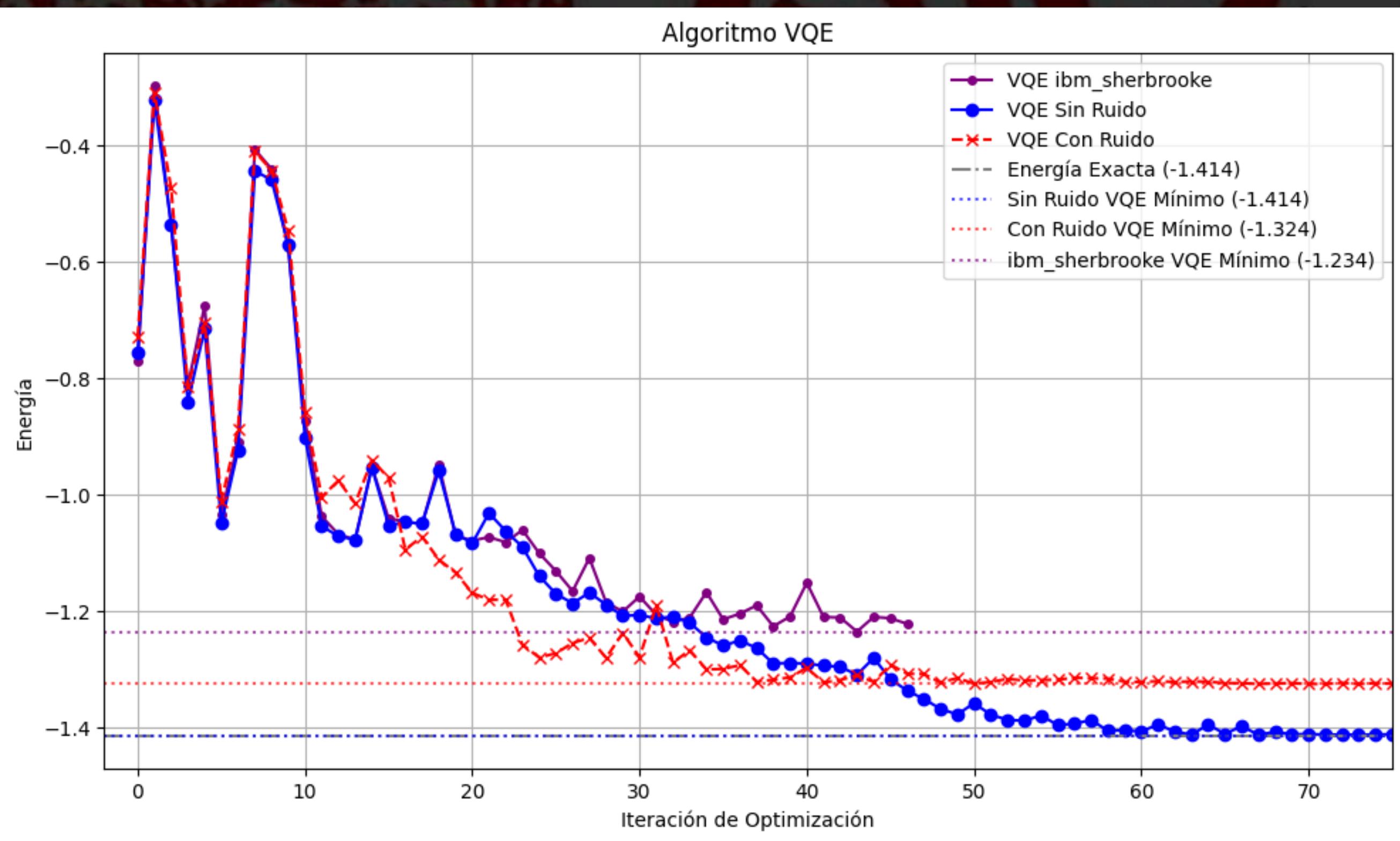
4

Corrección

M3 usa la matriz calibrada para ajustar estadísticamente esos conteos, aplicando una inversión de matriz para estimar la distribución real sin errores, obteniendo una distribución corregida de probabilidades, más fiel al estado cuántico verdadero

RESULTADOS Y ANALISIS DE FIDELIDAD

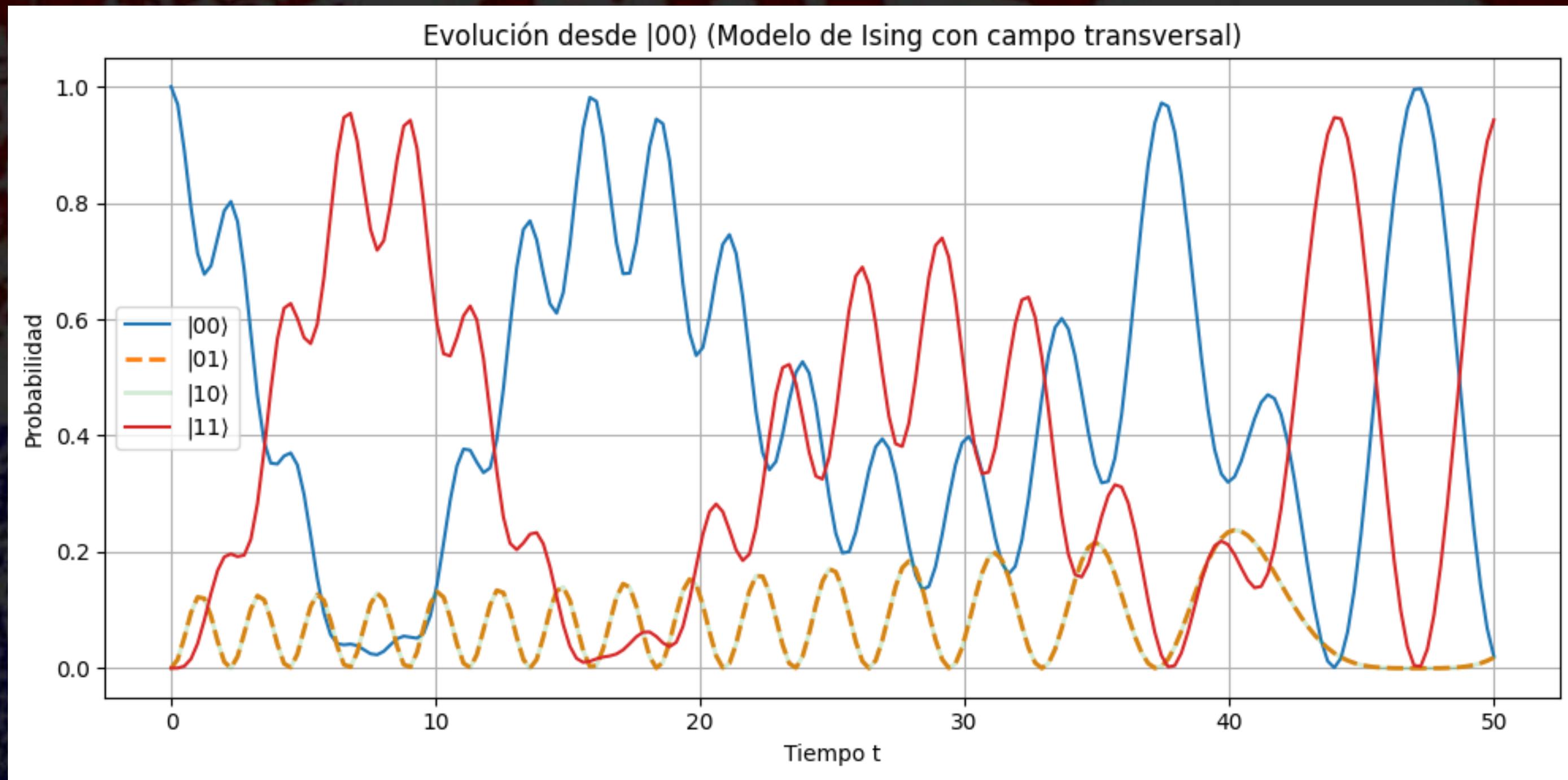
Ejecución en IBM Quantum:



Este gráfico muestra cómo evoluciona la función de costo (energía estimada) a lo largo de las iteraciones del optimizador clásico.

Se espera que esta curva baje progresivamente hasta estabilizarse, indicando que el VQA ha encontrado una buena aproximación del estado fundamental.

Ejecución en simulador local (sin ruido):



Aquí se muestra cómo cambian las probabilidades de observar cada estado base ($|00\rangle$, $|01\rangle$, $|10\rangle$, $|11\rangle$) durante la evolución temporal del sistema. Es útil para observar fenómenos de oscilación de población entre estados (como oscilaciones de Rabi, interferencia cuántica, etc.)

Ejecución en simulador local (con ruido, sin ruido y aplicando el QEM):

```
# Parámetros de error de puertas
```

```
p1, p2 = 0.01, 0.03
```

```
T1, T2, dt = 80e3, 60e3, 0.222e-6
```

```
depol1 = depolarizing_error(p1, 1)
```

```
therm1 = thermal_relaxation_error(T1, T2, dt)
```

```
err1q = depol1.compose(therm1)
```

```
depol2 = depolarizing_error(p2, 2)
```

```
# Error de lectura (readout) del 5%
```

```
ro_error = ReadoutError([[0.95, 0.05],  
[0.05, 0.95]])
```

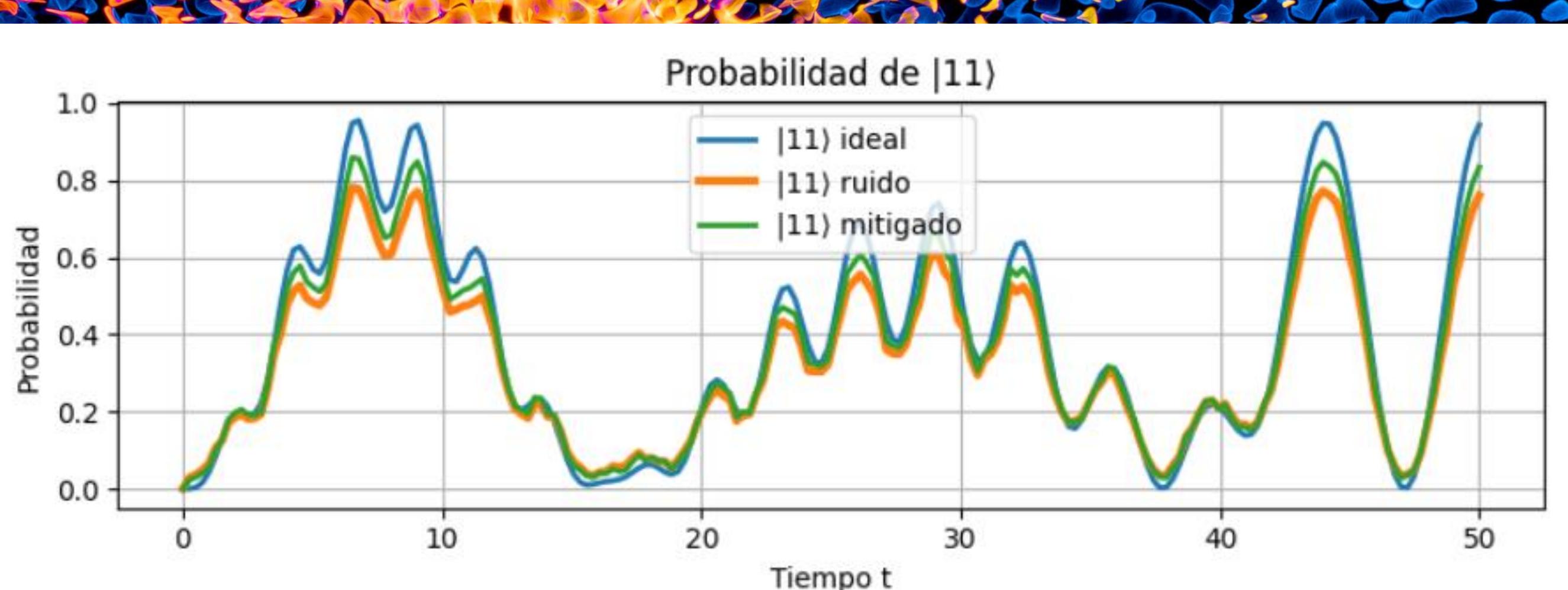
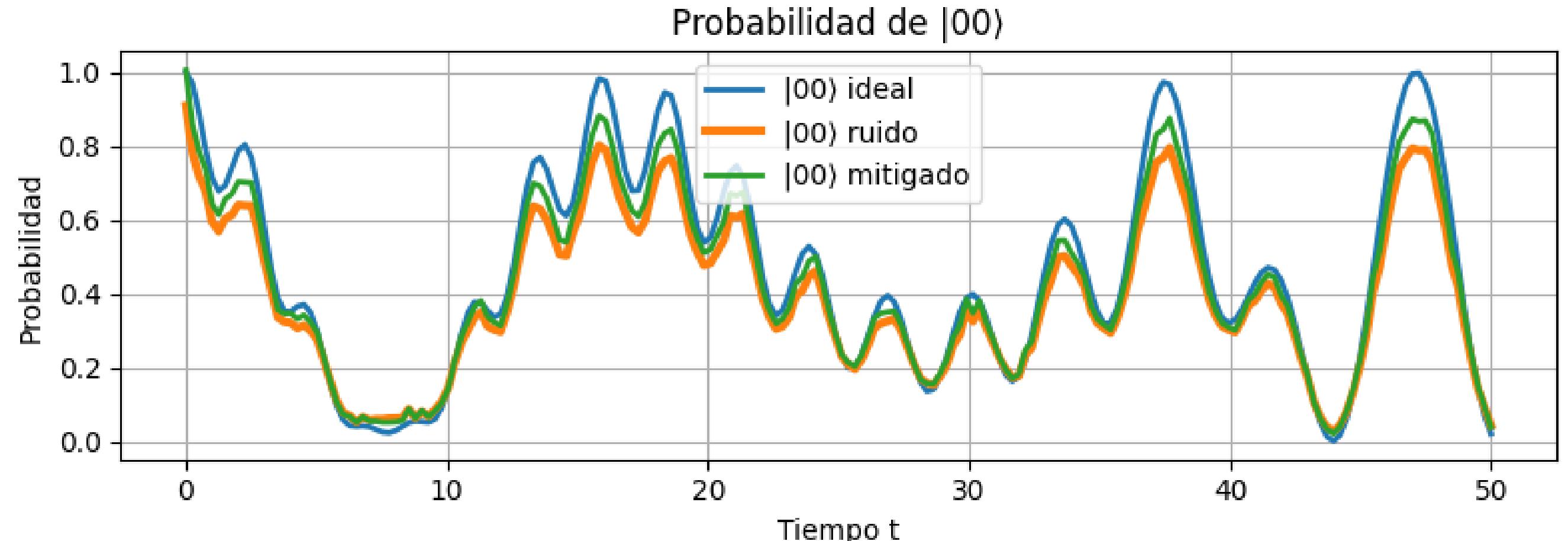
```
# Construcción del noise model
```

```
noise_model = NoiseModel()
```

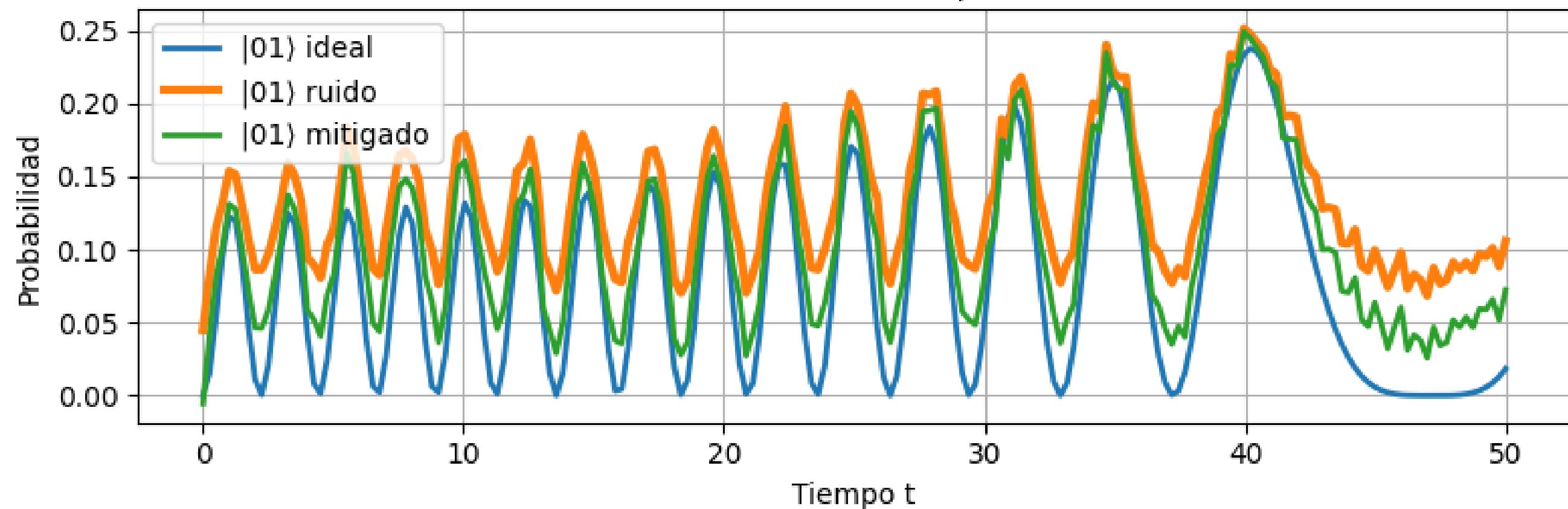
```
noise_model.add_all_qubit_quantum_error(err1q,  
['rx','ry','rz'])
```

```
noise_model.add_all_qubit_quantum_error(depol2,  
['cx'])
```

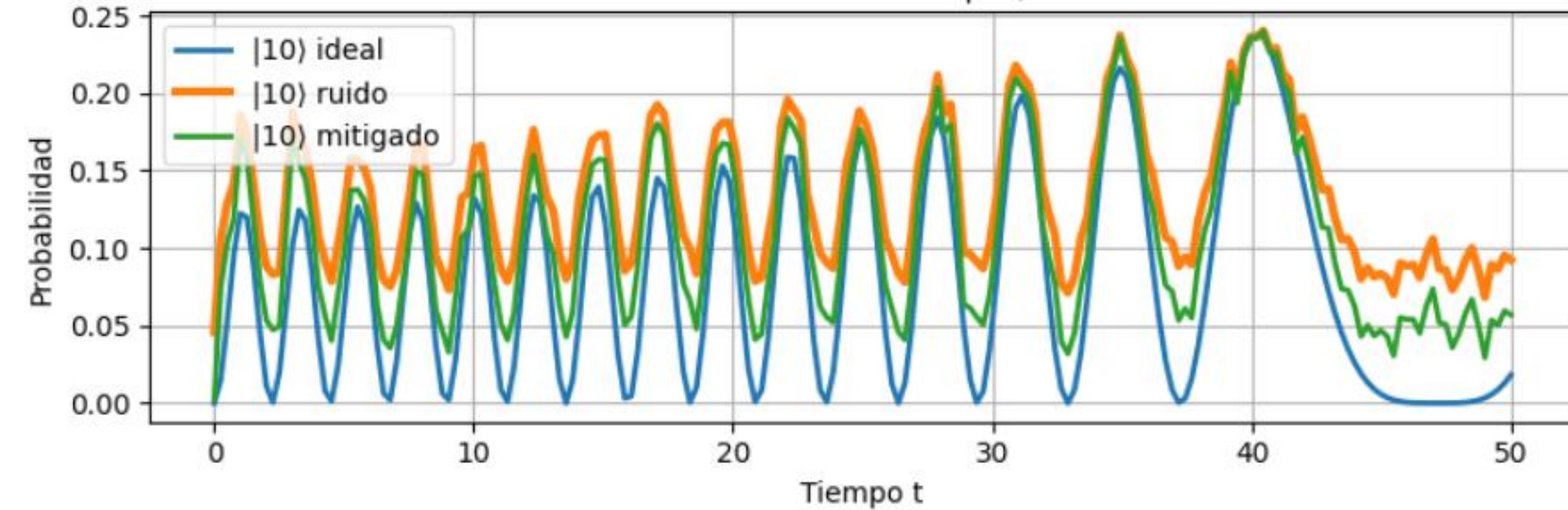
```
noise_model.add_all_qubit_readout_error(ro_error)
```

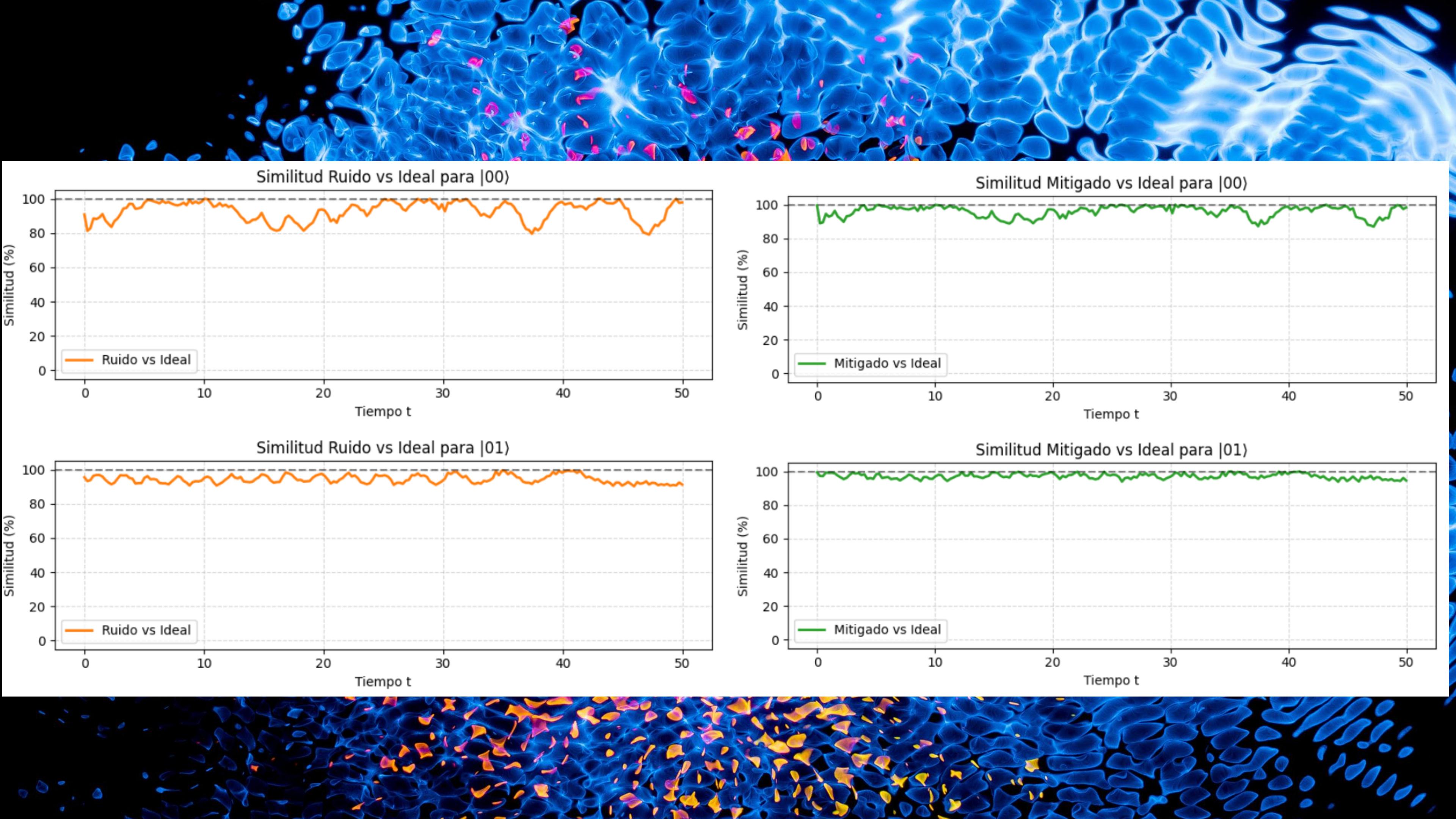


Probabilidad de $|01\rangle$



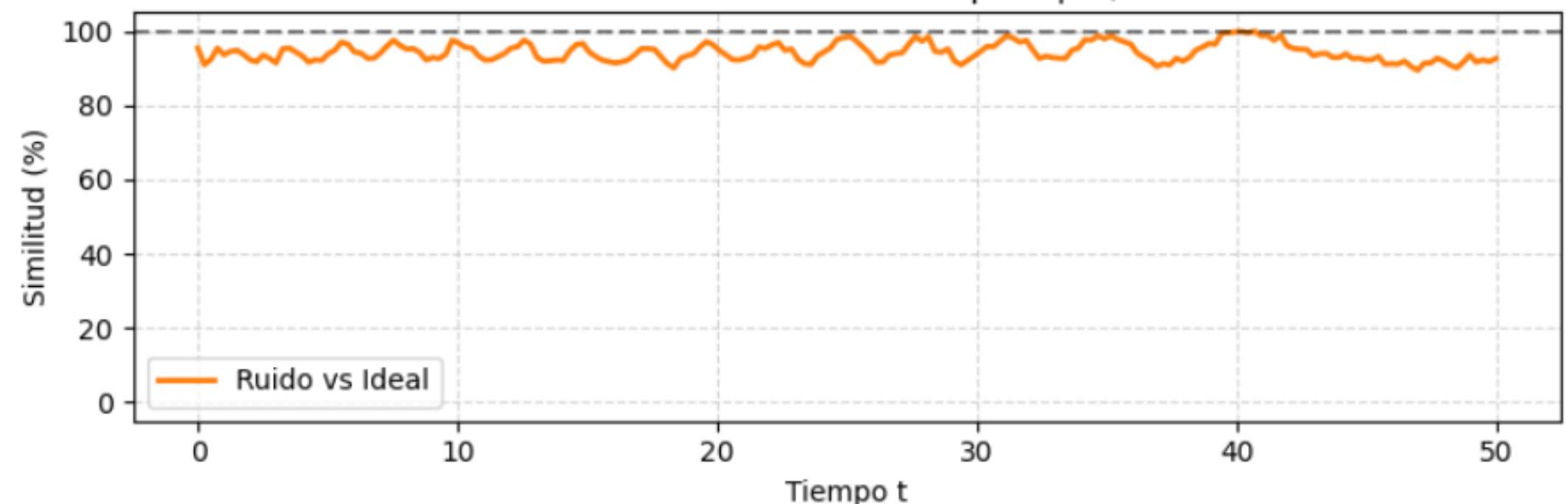
Probabilidad de $|10\rangle$



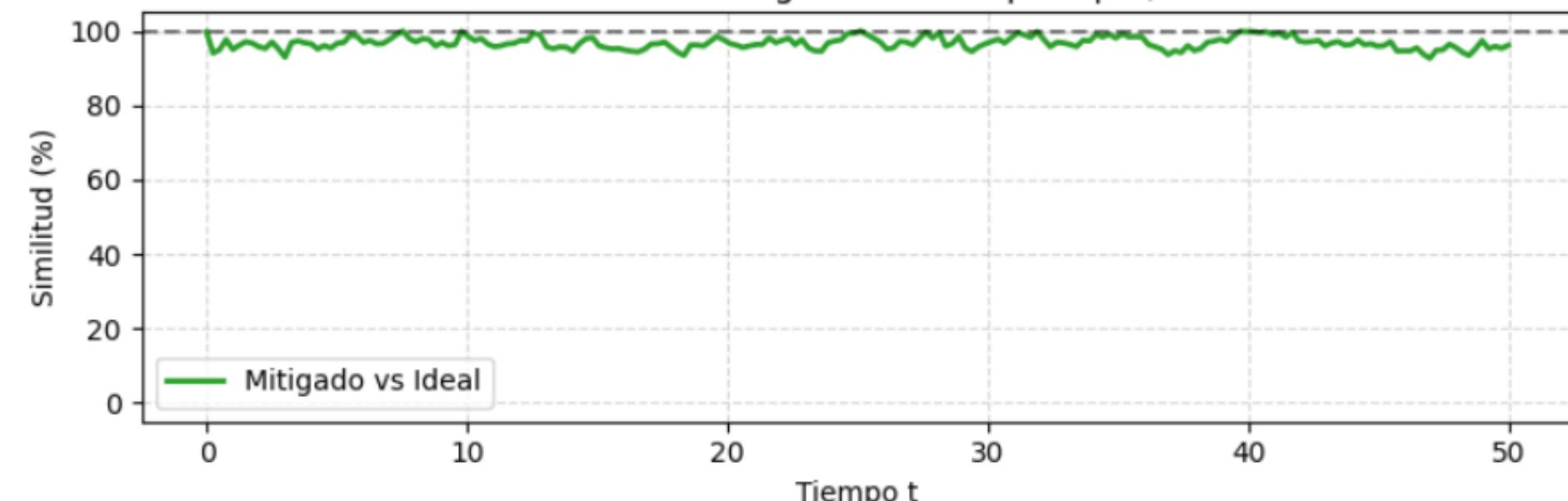




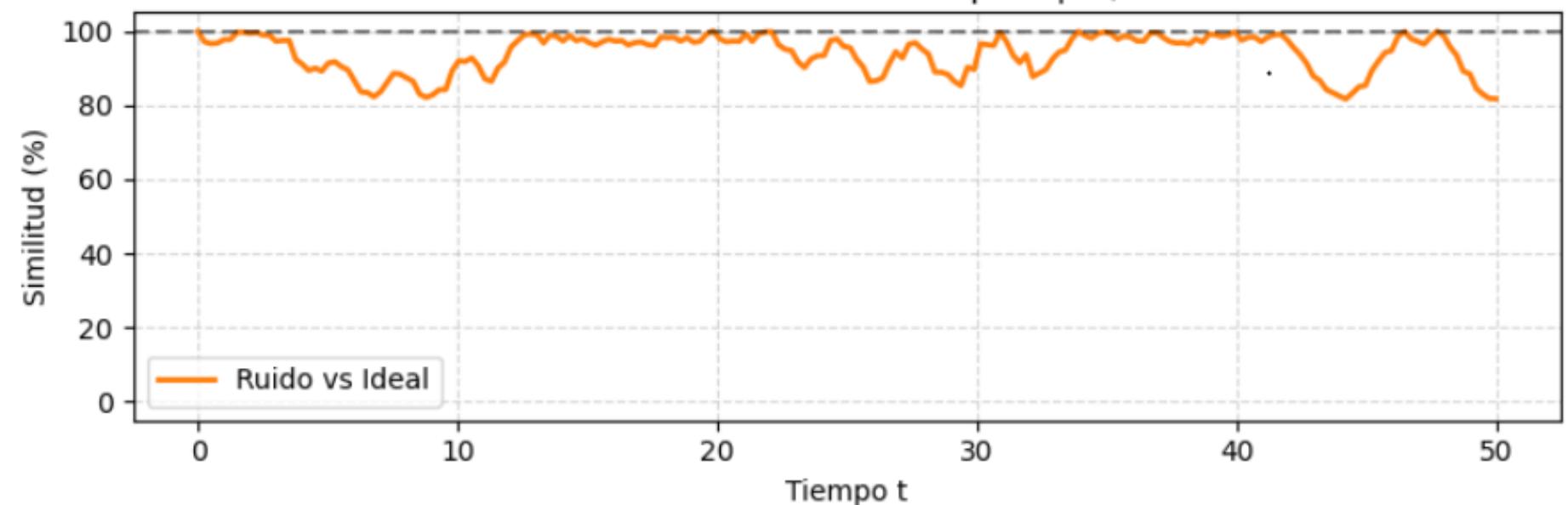
Similitud Ruido vs Ideal para |10)



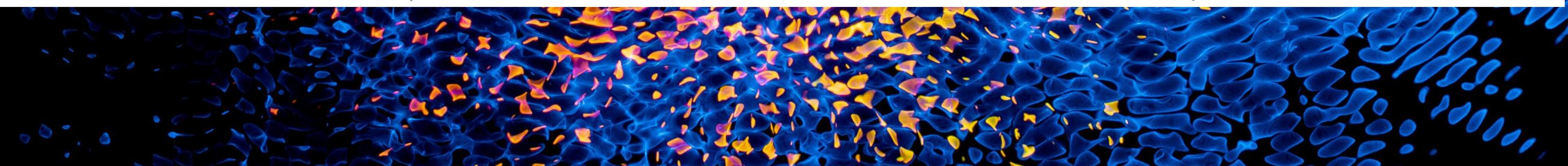
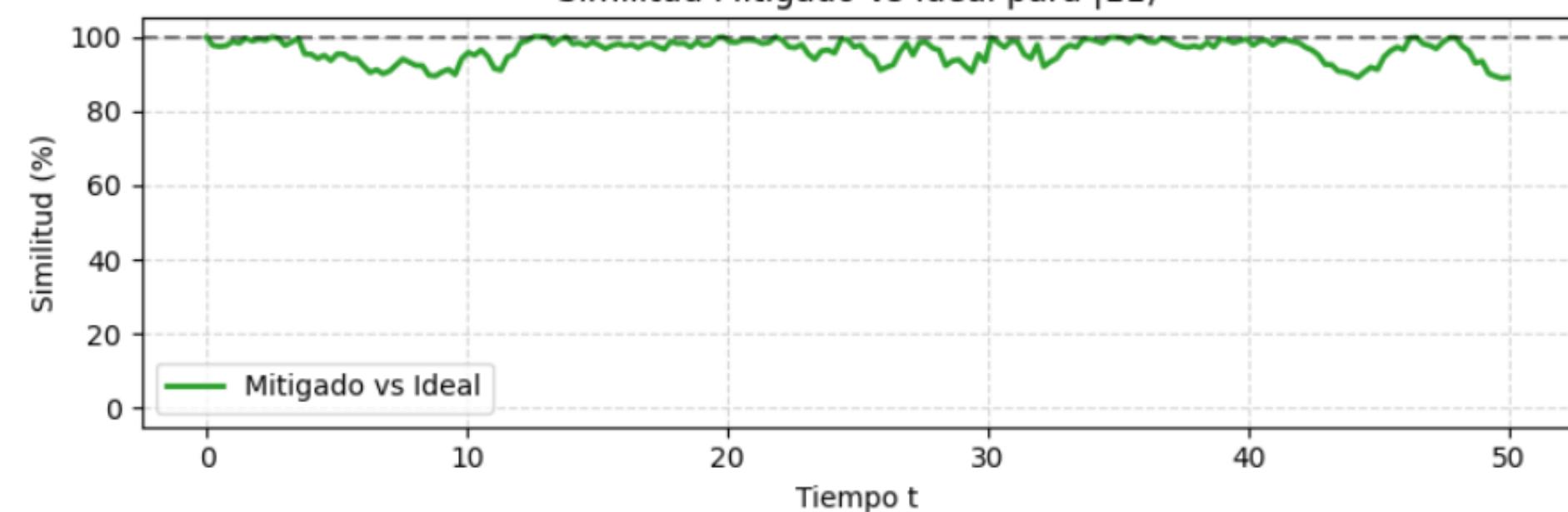
Similitud Mitigado vs Ideal para |10)



Similitud Ruido vs Ideal para |11)



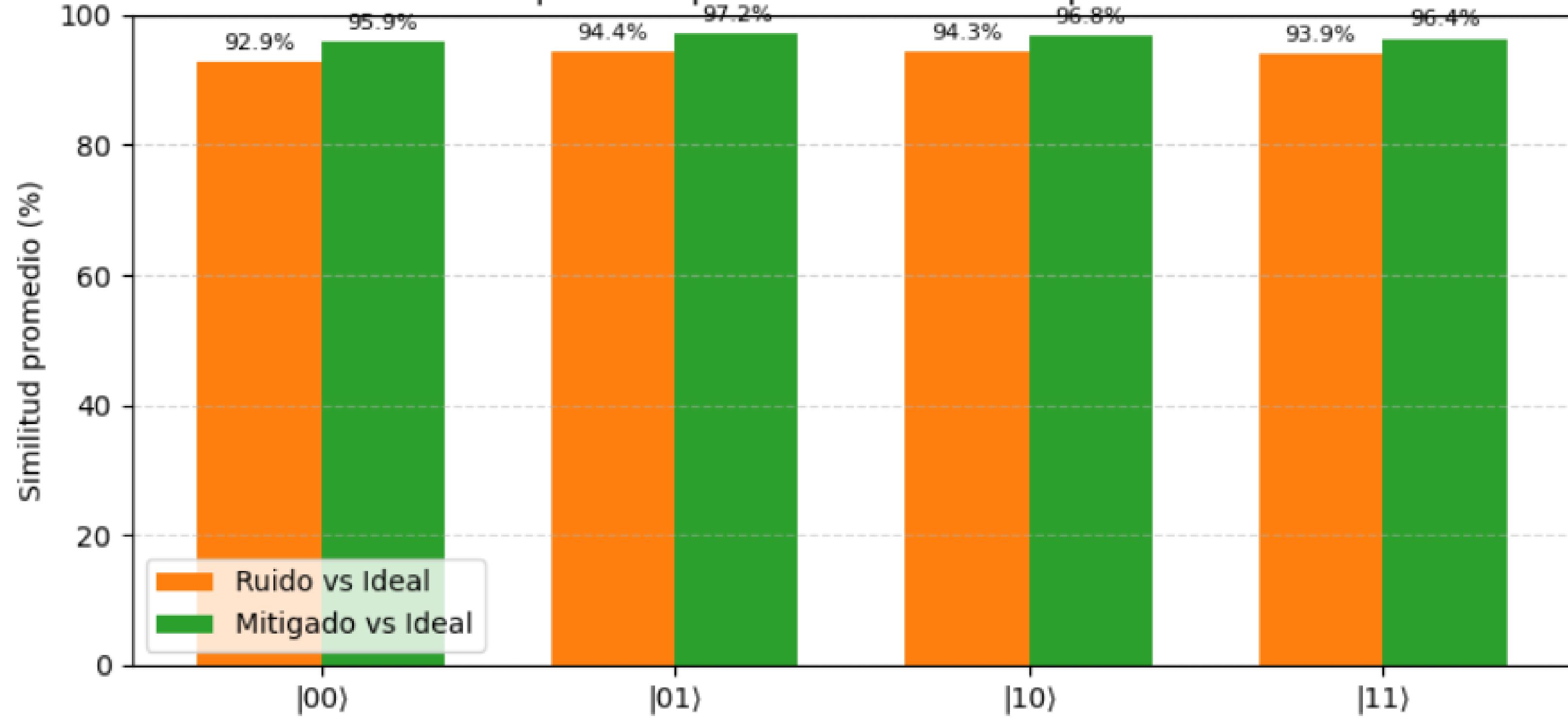
Similitud Mitigado vs Ideal para |11)



Promedio de similitud a lo largo de toda la evolución (0-100%):

Estado $ 00\rangle$: Ruido vs Ideal = 92.86%	Mitigado vs Ideal = 95.91%
Estado $ 01\rangle$: Ruido vs Ideal = 94.38%	Mitigado vs Ideal = 97.17%
Estado $ 10\rangle$: Ruido vs Ideal = 94.30%	Mitigado vs Ideal = 96.79%
Estado $ 11\rangle$: Ruido vs Ideal = 93.92%	Mitigado vs Ideal = 96.36%

Comparación promedio de similitud por estado



MEJORAS

1

**Comparar fidelidad de
ejecución en IBM
Quantum**

2

Robustecer el QEM

3

**Modulación de código
orientado a objetos**

REPOSITORIO GIT

<https://github.com/GaelMedC/Qubitronics.git>

[README](#)

Simulación Cuántica del Modelo de Ising con Campo Transversal

Este proyecto fue desarrollado como parte del Hack The QuBit 2025, organizado por el Quantum Computing Club y el Data Science Hub del Tecnológico de Monterrey. El reto consistía en simular un sistema físico cuántico, formular su Hamiltoniano, construir un circuito cuántico equivalente e implementar una estrategia de mitigación de errores.

Resumen del Proyecto

Simulamos el modelo de Ising en 1D con dos qubits y un campo transversal, utilizando Qiskit 2.1.0. El objetivo fue estimar la energía del estado fundamental mediante un algoritmo variacional (VQE), simular la evolución temporal mediante trotterización, y analizar los resultados obtenidos en simuladores y hardware real. Se utilizó el ansatz `EfficientsU2` y se aplicó mitigación de errores de lectura (`mthree.M3Mitigation`).

Objetivos del Hackathon

1. Formular el Hamiltoniano del sistema físico.
2. Resolverlo analíticamente (energías, evolución, observables).
3. Construir un circuito cuántico que lo modele.
4. Aplicar una técnica de mitigación o corrección de errores.
5. Ejecutar el circuito en simulador (ideal y ruidoso).
6. Ejecutar en hardware real de IBM Quantum.
7. Analizar resultados y fidelidad del sistema.

Tecnologías y Herramientas

- Python 3.10
- Qiskit 2.1.0
- `qiskit.primitives.StatevectorEstimator`
- `qiskit_aer.AerSimulator`
- `qiskit.circuit.library.EfficientsU2`
- `qiskit.quantum_info.SparsePauliOp`
- `matplotlib`, `numpy`, `scipy.optimize`

[README](#)

Formulación del Hamiltoniano

El Hamiltoniano simulado fue:

$$H = -J Z_0 Z_1 - g X_0 - g X_1$$

donde:

- (J) representa la interacción entre qubits,
- (g) es el campo transversal aplicado en la dirección X.

Implementado en Qiskit como `SparsePauliOp` con etiquetas de Pauli "zz", "xi", "ix".

Resolución Analítica y Evolución Temporal

Se compararon los valores de energía mínima encontrados por el algoritmo VQE contra los valores esperados de la diagonalización exacta del Hamiltoniano.

Además, se simuló la evolución temporal del sistema desde el estado $|00\rangle$ utilizando trotterización, una técnica que permite aproximar la evolución bajo un Hamiltoniano no trivial mediante productos de exponentiales de sus términos individuales. Esto permitió observar cómo cambian las poblaciones de los estados medidos en función del tiempo, revelando oscilaciones cuánticas coherentes.

Circuito Cuántico

Se utilizó un circuito variacional con el ansatz `EfficientsU2` sobre 2 qubits, parametrizado por un `ParameterVector`. La optimización se realizó con `COBYLA`, minimizando el valor esperado del Hamiltoniano.

Mitigación de Errores

Se aplicó Readout Error Mitigation utilizando la biblioteca `mthree`. El mitigador `M3Mitigation` se calibró en el backend real de IBM Quantum (`ibm_sherbrooke`) para los qubits medidos. Esta técnica permite reducir errores sistemáticos de lectura, mejorando la fidelidad sin alterar el circuito cuántico.

Resultados

- El VQE logró aproximar correctamente la energía del estado fundamental.
- La simulación de la evolución temporal mostró oscilaciones esperadas.
- Se observó una mejora significativa en fidelidad al aplicar mitigación de errores.
- El código está preparado para correr en backends reales de IBMQ, como `ibm_sherbrooke`.

Equipo

Participante principal:
José Gabriel Castillo García
Estudiante de Maestría en Inteligencia Artificial Aplicada
Tecnológico de Monterrey – Líderes del Mañana

Gael Medina Estudiante de Licenciatura en Física Universidad Autónoma de Nuevo León
Emilio Salcedo López Estudiante de Ingeniería en Electrónica Tecnológico de Monterrey

Licencia

Este repositorio está bajo la licencia MIT. Puedes usar, modificar y distribuir libremente este código, siempre que se mencione al autor original.

 *"La computación cuántica no solo consiste en simular sistemas ideales, sino en enfrentar el reto real: hacer ciencia sobre hardware imperfecto."*

BIBLIOGRAFÍA GENERAL

Simulation of quantum particles (2022)

<https://python.plainenglish.io/simulation-of-quantum-particles-with-python-and-qiskit-cfa656bb8773>

Introduction to Qiskit

<https://docs.quantum.ibm.com/guides>

Guo, Y., & Yang, S. (2022). **Quantum error mitigation via matrix product operators** [Preprint]. arXiv. <https://arxiv.org/pdf/2201.00752>

Phys466 Course Notes. (2016). **The Ising model.** University of Illinois. [Aregar algo de texto](#)

Wright, J. (2024). **Quantum coding theory: Lecture 1 – Introduction and the Shor 9-qubit code** [Notas de clase]. University of California, Berkeley. [Aregar algo de texto](#)

Variational Quantum Algorithms (2025)

Por Gilbereto Juárez Rangel

The Ising Model

<https://stanford.edu/~jeffjar/statmech/intro4.html>

IBM Quantum. (s.f.). **Setup your quantum channel.** IBM Quantum Documentation. [Aregar algo de texto](#)

Devitt, S. J., Munro, W. J., & Nemoto, K. (2024). **Quantum error correction for beginners.** National Institute of Informatics & NTT Basic Research Laboratories. Manuscrito presentado para su publicación.