GroverFigureOfMerit: An Agnostic Figure of Merit for Quantum Backend Characterization in the NISQ Era

Tiago Restucha[1], Marcos Guillermo Lammers[1], and Alejandro Fernández[1]

Universidad Nacional de La Plata, Facultad de Informática, Laboratorio de Investigación y Formación en Informática Avanzada (LIFIA), Argentina

Abstract. The Noisy Intermediate-Scale Quantum (NISQ) era poses a concrete challenge for developers: quantum hardware providers expose their capabilities through heterogeneous interfaces with proprietary metrics that vary widely across providers, hindering informed backend selection. The commonly available static characterization metrics—coherence times T_1/T_2, gate error rates, between others, exhibit specific limitations: they fail to capture dynamic variability across successive executions, overlook the impact of the transpilation process, and lack architectural comparability across physically distinct technologies. In this work, we propose a Figure of Merit (FoM) based on Grover's algorithm, designed to act as an algorithmic stress test that holistically evaluates the performance of quantum backends. The proposed metric combines the probability of success on target states with penalties for non-uniform amplification and probability leakage to non-marked states, yielding a unified and comparable score across different hardware architectures. We implement this FoM on top of the Qonscious framework, a conditional execution platform that readily provides the polymorphic adapters needed to execute the same metric in an agnostic manner on IBM and IonQ backends, as well as simulators. The main contributions are: (1) the proposal and validation of the GroverFigureOfMerit, a Grover-based metric that incorporates uniformity and leakage penalties (adapted from previous approaches such as GRADE), with emphasis on capturing the combined effect of noise, transpilation, and topological constraints; (2) a systematic analysis of heterogeneity across nine quantum providers, which motivates the need for agnostic metrics; and (3) the experimental demonstration of the proposed FoM through executions on ideal simulators and noise models derived from real processors, demonstrating its sensitivity to noise, topology, and transpilation overhead. The results confirm that the proposed metric enables distinguishing backend performance under a unified score, capturing intrinsic algorithmic limits. Validation on physical QPUs is identified as a natural next step.

Keywords: NISQ computing · quantum benchmarking · Grover's algorithm · GRADE · backend characterization · Qonscious framework

GroverFigureOfMerit: Una Figura de Mérito Agnóstica para la Caracterización de Backends Cuánticos en la Era NISQ

Resumen La era de la computación cuántica de escala intermedia ruidosa (NISQ) plantea un desafío concreto para los desarrolladores: los proveedores de hardware cuántico exponen sus capacidades a través de interfaces heterogéneas con métricas propietarias que varían ampliamente entre proveedores, dificultando una selección informada de backends. Las métricas de caracterización estática habituales, tiempos de coherencia T_1/T_2, tasas de error de compuerta, entre otras. Presentan limitaciones específicas: no capturan la variabilidad dinámica entre ejecuciones sucesivas, ignoran el impacto del proceso de transpilación y carecen de comparabilidad arquitectónica entre tecnologías físicas distintas. En este trabajo proponemos una Figura de Mérito (FoM) basada en el algoritmo de Grover, diseñada para actuar como un stress test algorítmico que evalúa de forma holística el rendimiento de backends cuánticos. La métrica propuesta combina la probabilidad de éxito de los estados objetivo con penalizaciones por desuniformidad en la amplificación y por fuga de probabilidad hacia estados no marcados, generando un score unificado y comparable entre distintas arquitecturas de hardware. Implementamos esta FoM sobre el framework Qonscious, una plataforma de ejecución condicional que proporciona los adaptadores polimórficos necesarios para ejecutar la misma métrica de forma agnóstica en backends de IBM, IonQ y simuladores. Las contribuciones principales son: (1) la propuesta y validación del GroverFigureOfMerit, una métrica basada en Grover que incorpora penalizaciones por desuniformidad y dispersión de probabilidad (adaptadas de enfoques previos como GRADE), con énfasis en capturar el efecto combinado de ruido, transpilación y restricciones topológicas; (2) un análisis sistemático de la heterogeneidad entre nueve proveedores cuánticos, que motiva la necesidad de métricas agnósticas; y (3) la demostración experimental de la FoM mediante ejecuciones en simuladores ideales y modelos de ruido derivados de procesadores reales. Los resultados confirman que la métrica permite distinguir el rendimiento entre backends bajo un score unificado, capturando los límites algorítmicos intrínsecos. La validación sobre QPUs físicas queda identificada como paso natural de confirmación.

Keywords: computación NISQ · benchmarking cuántico · algoritmo de Grover · GRADE · caracterización de backends · framework Qonscious

1 Introducción

La proliferación de diversas arquitecturas de computación cuántica exige una metodología robusta y estandarizada para evaluar el rendimiento de los *backends*. Las computadoras cuánticas son sistemas complejos cuyo comportamiento emergente no puede ser capturado ni predicho de manera confiable basándose únicamente en métricas de componentes de bajo nivel (Proctor et al., 2025). Los parámetros crudos de los dispositivos resultan insuficientes debido a su naturaleza libre de contexto, inestabilidad temporal y falta de comparabilidad arquitectónica entre proveedores como IBM, IonQ y Rigetti.

En consecuencia, este trabajo propone una Figura de Mérito (FoM) para la caracterización de backends cuánticos. Específicamente, diseñamos el GroverFigureOfMerit, una métrica que utiliza el algoritmo de Grover como *stress test* algorítmico para capturar el efecto combinado del ruido, la transpilación y las restricciones topológicas en un score unificado. Para abordar la heterogeneidad en la ejecución, implementamos nuestra FoM sobre el framework Qonscious (Lammers et al., 2025), incorporando la función

de scoring de enfoques recientes como GRADE (Manor et al., 2025) en su arquitectura extensible.

1.1 Métricas Estáticas vs. Rendimiento Algorítmico

Las métricas de caracterización específicas de cada proveedor son valiosas para comprender el hardware a nivel de componentes. Sin embargo, su limitado poder predictivo ha impulsado un enfoque creciente hacia benchmarks que resuman el desempeño holístico del sistema de manera directa y concisa (Lubinski et al., 2023). Desde nuestra perspectiva, esta evolución hacia benchmarks holísticos responde a que las métricas de bajo nivel, cuando se analizan de manera aislada, enfrentan las siguientes particularidades:

Poder predictivo limitado: Un backend con bajas tasas de error de compuerta puede fallar en amplificar estados objetivo en algoritmos como la búsqueda de Grover. Esto ocurre porque las métricas de bajo nivel no contemplan el impacto del proceso de transpilación, el cual genera una profundidad física que suele ser sustancialmente mayor a la profundidad original del circuito de alto nivel. Por ejemplo, las compuertas multi-controladas (como las compuertas Toffoli) deben descomponerse en múltiples compuertas nativas de uno y dos qubits para poder ejecutarse, introduciendo una profundidad de circuito excesiva que compromete el desempeño global (Lubinski et al., 2023).

Falta de relevancia contextual: Las métricas estáticas no logran encapsular el comportamiento de algoritmos particulares. Un conjunto de compuertas con alta fidelidad individual puede no traducirse en una ejecución exitosa de un algoritmo específico si la interacción entre compuertas, los tiempos de espera (*idle*) entre operaciones y la topología afectan de manera no trivial al resultado final.

Variabilidad temporal: Las propiedades del hardware cambian constantemente debido a factores ambientales y a la acumulación de perturbaciones aleatorias del entorno físico. Como se evidencia en la literatura, esta ”distribución de frecuencia impredecible y variabilidad temporal interfiere con la manipulación de los qubits, haciendo necesaria una recalibración frecuente” (Jiang et al., 2025). Estas continuas recalibraciones provocan que las instantáneas de caracterización estática pierdan validez rápidamente, ofreciendo una visión difusa del estado real del backend en el momento de la ejecución.

Estas características motivan un enfoque complementario: *FoMs basadas en circuitos* que miden el rendimiento de extremo a extremo ejecutando circuitos cuánticos y analizando los resultados. A diferencia de las métricas estáticas, las FoMs son:

- **Dinámicas:** Capturan el comportamiento de ejecución real incluyendo efectos de transpilación, acumulación de ruido y rendimiento específico del algoritmo.
- **Agnósticas al backend:** El mismo circuito de FoM puede ejecutarse en cualquier backend cuántico independientemente de las APIs específicas del proveedor.
- **Interpretables:** Las FoMs producen puntajes cuantitativos que se relacionan directamente con criterios de éxito algorítmico.

1.2 Contribución

Las contribuciones principales son el análisis de heterogeneidad entre nueve proveedores, la extensión del framework Qonscious con FoMs de circuitos agnósticas al backend, y la

validación experimental de la métrica propuesta sobre simuladores ideales y modelos de ruido realistas.

1.3 Trabajos Relacionados

El benchmarking en la era NISQ ha evolucionado desde métricas estáticas de bajo nivel hacia evaluaciones algorítmicas holísticas que capturan el rendimiento real de extremo a extremo (Lubinski et al., 2023; Proctor et al., 2025). Herramientas como Benchpress (Nation et al., 2025) ofrecen suites variadas de circuitos representativos, mientras que GRADE introduce un enfoque basado en Grover (Manor et al., 2025). Frameworks como QUARK y BenchQC enfatizan extensibilidad y portabilidad; en nuestro caso, enfatizamos la integración de las métricas directamente en tiempo de runtime para una ejecución consciente de recursos (Lammers et al., 2025).

2 Análisis de Heterogeneidad de Múltiples Backends

Para dimensionar la complejidad que enfrenta un desarrollador en la era NISQ, la Figura 1 presenta una taxonomía de proveedores agrupados por su arquitectura física. Cada proveedor impone su propio SDK, su propio conjunto de compuertas nativas (*gate set*) y sus propios mecanismos para exponer datos de calibración.

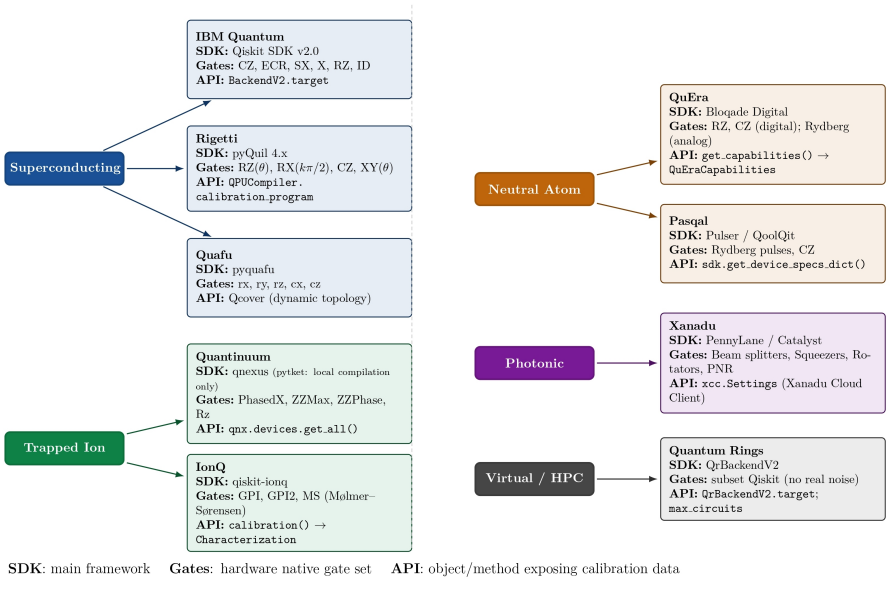


Fig. 1. Clasificación de proveedores de computación cuántica por arquitectura física.

Esta diferencia es lógica ya que ”diferentes realizaciones físicas de qubits dan lugar a distintas primitivas de circuito y compromisos computacionales” pero la realidad es

que ”este paradigma basado en circuitos es en gran medida agnóstico al hardware y se mapea naturalmente en la mayoría de las principales tecnologías de procesadores cuánticos, convirtiéndolo en la abstracción dominante para la computación cuántica en la era NISQ”(Gonski et al., 2026). Por lo tanto para la capa algorítmica debería ser prácticamente invisible. El análisis de las interfaces de calibración revela tres barreras que impiden predecir el éxito de un circuito basándose únicamente en métricas estáticas:

1. **Fragmentación de APIs y Compuertas Nativas:** Incluso dentro de una misma arquitectura física existen diferencias. En los sistemas superconductores, IBM Quantum (2026) y Rigetti Computing (2026) utilizan compuertas nativas distintas (ECR/SX frente a CZ/CPHASE) y exponen la topología mediante distintos tipos de objetos (JSON, diccionarios de datos, etc.). Esta misma heterogeneidad de interfaces se observa a lo largo de todo el ecosistema cuántico, desde plataformas de iones atrapados hasta sistemas fotónicos (véase por ejemplo, IonQ, 2026; Xanadu, 2026). Desde nuestra perspectiva, el diseño de compiladores portables se transforma en un problema tedioso.
2. **Semántica de las Métricas:** Los datos de calibración de bajo nivel generalmente no son comparables. IBM reporta errores de compuerta y duraciones por cada enlace físico específico. En contraste, las arquitecturas de iones de Quantinuum (2026) o los átomos neutros de QuEra Computing (2026) suelen reportar métricas globales o promedios, dado que sus qubits son idénticos o reconfigurables dinámicamente.
3. **Opacidad y Volatilidad:** El ruido cuántico no es estático, pero la disponibilidad de datos dinámicos es desigual. Algunos proveedores exponen historiales ricos de caracterización, mientras que otros ofrecen acceso limitado o fuertemente encapsulado, lo que dificulta cualquier flujo automatizado de selección de hardware.

Desde la perspectiva del desarrollador, las variables relevantes son las restricciones efectivas del grafo computacional (ancho, profundidad soportada y costo de conectividad real). En consecuencia, este trabajo adopta un enfoque top-down en favor de FoMs estandarizadas y basadas en la ejecución algorítmica. Esta aproximación permite:

- Abstraer las llamadas a APIs propietarias mediante adaptadores polimórficos, permitiendo ejecutar el mismo benchmark en cualquier backend.
- Evaluar el rendimiento real del hardware mediante FoMs dinámicas que capturan efectos de transpilación, ruido y topología.
- Caracterizar la idoneidad del procesador para tareas computacionales específicas mediante métricas centradas en algoritmos, traduciendo así la disparidad física a un puntaje (*score*) comparable y agnóstico.

3 El Framework Qonscious como Plataforma de Implementación

La FoM propuesta ha sido implementada sobre el framework Qonscious (Lammers et al., 2025), una plataforma de código abierto diseñada originalmente para la ejecución condicional consciente de recursos en entornos NISQ. Qonscious proporciona dos elementos que facilitan la implementación de FoMs agnósticas al backend:

– **Adaptadores polimórficos (BackendAdapters):** Abstraen las diferencias entre los SDKs y APIs de los distintos proveedores. Un mismo circuito lógico puede ser transpilado y ejecutado en cualquier backend soportado sin modificar el código de la FoM.
– **Flujo de ejecución condicional:** Permite parametrizar verificaciones (checks) cuyos resultados alimentan una función de decisión que puede desencadenar, pausar o cancelar la ejecución del circuito principal. Este flujo, ilustrado en la Figura 2, es el que utilizamos para ejecutar nuestra métrica como mecanismo de caracterización.

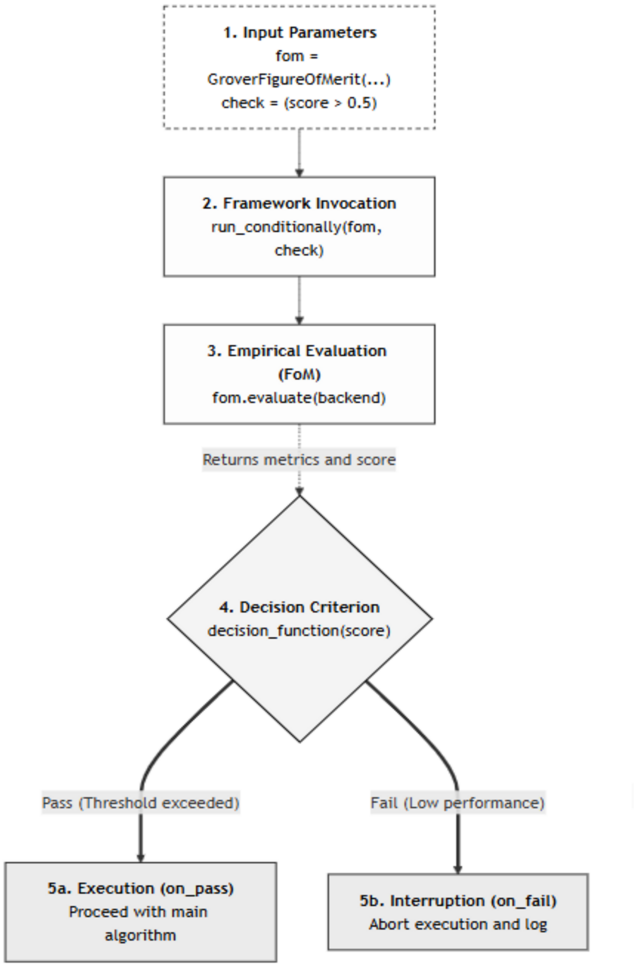


Fig. 2. Flujo de ejecución condicional de Qonscious. El resultado de la FoM (score) puede alimentar la función de decisión para optimizar el uso de recursos cuánticos.

En este trabajo nos centramos en la fase de caracterización, utilizando el flujo de Qonscious para ejecutar la FoM en múltiples backends y obtener scores comparables. La integración de estos scores como condiciones para decisiones runtime queda habilitada para trabajos futuros.

4 GroverFigureOfMerit: Diseño y Definición

Para demostrar la utilidad práctica del enfoque propuesto, implementamos una FoM basada en el algoritmo de Grover, adaptando la función de scoring de GRADE (Manor et al., 2025) a la arquitectura extensible de Qonscious.

4.1 Motivación y Diseño

El algoritmo de Grover (Grover, 1996) amplifica cuadráticamente la probabilidad de medir estados objetivo en un espacio de búsqueda no estructurado. En un backend ideal sin ruido, Grover alcanza probabilidad de éxito cercana a 1 para el número óptimo de iteraciones $R = \lfloor \frac{\pi}{4}\sqrt{N/M} \rfloor$, donde N es el tamaño del espacio de búsqueda y M es el número de estados objetivo. En dispositivos NISQ, el ruido degrada esta amplificación. La profundidad del circuito de Grover escala con R, y cada iteración incluye operaciones multi-controladas (*oracle*) y múltiples compuertas de 1 y 2 qubits (*difusión*). Esto hace a Grover un *stress test* algorítmico ideal para evaluar el hardware bajo condiciones reales de ejecución algorítmica.

4.2 Construcción del Circuito

El circuito de Grover se construye en tres etapas:

1. Inicialización: Aplicar compuertas Hadamard a los n qubits para crear una superposición uniforme:

$$|\psi\rangle = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} |x\rangle \tag{1}$$

2. Oráculo O_ω: Marca la solución aplicando una inversión de fase:

$$O_\omega|x\rangle = \begin{cases} -|x\rangle, & \text{si } x = \omega, \\ |x\rangle, & \text{en otro caso.} \end{cases} \tag{2}$$

3. Operador de difusión D: Invierte las amplitudes respecto del promedio:

$$D = 2|\psi\rangle\langle\psi| - I \tag{3}$$

4. Iteración: Repetir R veces, donde el número óptimo es aproximadamente:

$$R^* \approx \left\lfloor \frac{\pi}{4}\sqrt{\frac{N}{M}} \right\rfloor \tag{4}$$

siendo M el número de estados marcados.

4.3 Cálculo del Score y Parametrización

Tras ejecutar el circuito con $S = 2000$ repeticiones (*shots*), obtenemos una distribución de conteos $\{c_x\}$ para cada bitstring $x \in \{0,1\}^n$. Estimamos las frecuencias relativas de medición: $f(x) = c_x/S$, que aproximan la distribución de probabilidad del estado cuántico final. A efectos del cálculo del score, empleamos $P(x) \approx f(x)$ como estimador de la probabilidad de cada bitstring.

- Frecuencia relativa acumulada en objetivos, donde T es el conjunto de estados objetivo:
$$P_T = \sum_{x \in T} f(x) \tag{5}$$
- Probabilidad en no-objetivos:
$$P_N = 1 - P_T = \sum_{x \notin T} P(x) \tag{6}$$
- Desviación estándar entre objetivos:
$$\sigma_T = \sqrt{\frac{1}{M} \sum_{x \in T} (P(x) - \bar{P}_T)^2}, \quad \bar{P}_T = \frac{P_T}{M} \tag{7}$$
Esta métrica penaliza situaciones donde un objetivo se amplifica mucho más que otros (indicando un desequilibrio provocado por errores coherentes).
- Score final:
$$\text{Score} = \max(0, P_T - \lambda \cdot \sigma_T - \mu \cdot P_N) \tag{8}$$

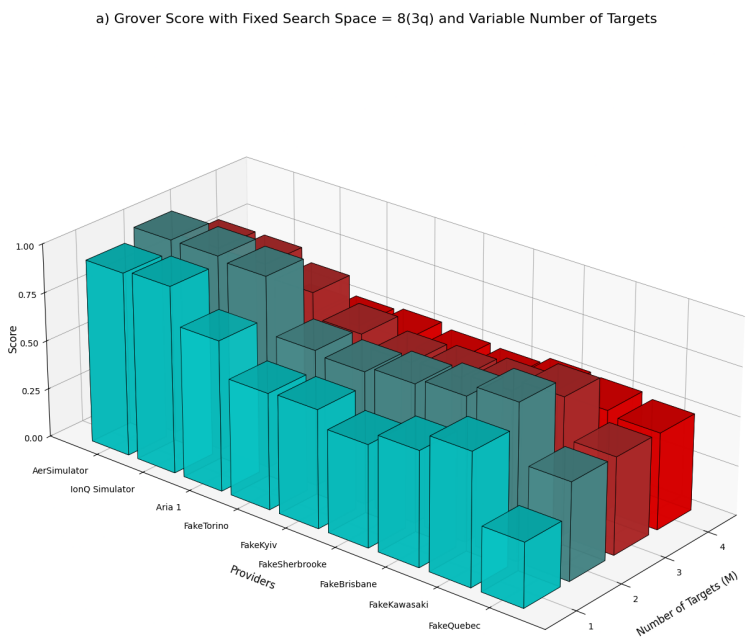
Para la construcción del circuito y su posterior evaluación, la implementación define la métrica a través de un conjunto de parámetros configurables. En primer lugar, se establece el número de estados a marcar mediante `num_targets` (M). El comportamiento analítico se ajusta con `lambda_factor` (λ) y `mu_factor` (μ), que introducen penalizaciones por el desbalance de probabilidad entre los propios objetivos y por la dispersión hacia estados no deseados, respectivamente. Además, el sistema acepta configuraciones opcionales como `num_qubits` (n), el cual fija el ancho del registro cuántico y se infiere automáticamente a partir de M si se omite, y `targets_int`, que permite asignar una lista explícita de enteros representando los estados objetivo en lugar de delegar su selección a un proceso aleatorio. Si $\mu \cdot P_N \geq P_T$, entonces Score = 0. Esto refleja situaciones donde los circuitos transpilados aumentan en profundidad debido a la descomposición de compuertas multicontroladas. Valores típicos de los parámetros de balance son $\lambda = 1$ y $\mu = 1$ para penalizar por igual la desuniformidad y la dispersión de probabilidad, aunque pueden ajustarse según el contexto de aplicación o las características del backend evaluado.

5 Experimentación y Resultados

Para validar empíricamente la sensibilidad del indicador propuesto frente a distintas restricciones de hardware, se diseñó un experimento de caracterización comparativo

utilizando los adaptadores nativos del *framework* Qonscious. El objetivo no es establecer un *benchmark* definitivo sobre el estado del arte comercial, sino demostrar que la métrica es sensible al ruido, la topología y la sobrecarga de transpilación de forma diferenciada entre backends. Para este fin, los modelos de ruido empleados —la familia *Fake Providers* de IBM y el modelo *Aria 1* de IonQ— constituyen una base de validación suficiente y controlada: son modelos derivados de caracterizaciones reales de hardware que preservan las propiedades topológicas y los perfiles de error de los procesadores físicos correspondientes, permitiendo reproducibilidad experimental sin depender de la disponibilidad de QPUs en la nube. La validación sobre procesadores físicos reales, que permitiría capturar fenómenos dinámicos adicionales como *crosstalk*[1] y fluctuaciones térmicas, queda identificada como paso natural de confirmación en trabajos futuros. En este experimento, se evaluó la respuesta del *score* al variar la densidad de soluciones en dos espacios de búsqueda fijos: $N = 8$ ($n = 3$ qubits) y $N = 32$ ($n = 5$ qubits). Para el primer caso, se incrementó secuencialmente el número de estados objetivo $M \in \{1,2,3,4\}$, mientras que para el segundo se evaluó $M \in \{1,2,3,4,8\}$. Se contrastó el rendimiento de simuladores ideales (`AerSimulator`, `IonQ Simulator`) frente a modelos de ruido basados en procesadores reales (la familia *Fake Providers* de IBM y el modelo `Aria 1` de IonQ). La Figura 3 ilustra los *scores* resultantes para ambos escenarios, evidenciando el impacto de la profundidad del circuito a medida que escalan los qubits.

[1] *Crosstalk* (diafonía en español) se refiere a interferencias no deseadas entre qubits adyacentes durante la ejecución de compuertas simultáneas, causando errores correlacionados que no son capturados por modelos de ruido independientes.



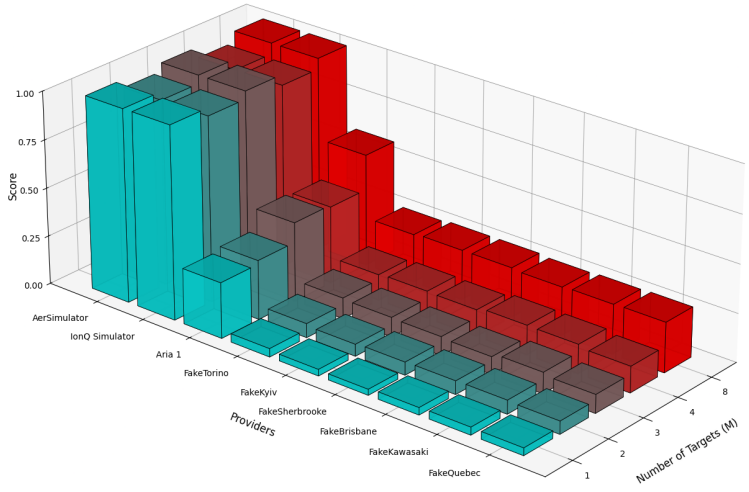


Fig. 3. Benchmarking comparativo del *Grover Score* con un número variable de *targets*. **a)** Para $N = 8$, se observa un pico de rendimiento en $M = 2$ coincidente con la rotación teórica óptima, y una degradación generalizada a ~ 0.5 en $M = 4$ debida al límite algorítmico de Grover ($M = N/2$). **b)** Para $N = 32$, los simuladores ideales mantienen una alta fidelidad, mientras que los *Fake Providers* sufren una severa descoherencia, relegando sus *scores* a la probabilidad de adivinanza aleatoria clásica (M/N). `Aria 1` demuestra una mayor resiliencia frente al ruido, superando ampliamente a los modelos de IBM en este espacio de búsqueda.

El análisis de los resultados revela un comportamiento no lineal predecible. Para $N = 8$, los simuladores ideales logran un *score* perfecto de 1.000 en $M = 2$ (correspondiente a la rotación de amplitud óptima inicial de $\theta = 30°$). Las arquitecturas de IBM sufren penalizaciones por ruido, variando entre 0.495 y 0.792, y al alcanzar el límite algorítmico de Grover en $M = 4$ ($M = N/2$), todos los *scores* en hardware ruidoso colapsan predeciblemente hacia ≈ 0.500.

Por otro lado, para $N = 32$ ocurre un comportamiento teórico similar en proporciones equivalentes (como el *score* perfecto en $M = 8$), pero se evidencia una degradación mucho más severa en la ejecución física: la mayor profundidad del circuito para 5 *qubits* provoca que los modelos de IBM colapsen totalmente por decoherencia, reduciendo sus resultados a probabilidades de adivinanza aleatoria clásica (≈ 0.25). En este *trade-off* general de escala y error capturado por la métrica, el modelo `Aria 1` de IonQ destaca por tener una resiliencia notablemente superior, siendo el único capaz de separarse del límite aleatorio y preservar la ventaja cuántica al escalar.

Para aislar el efecto de la topología física y evaluar la escalabilidad, el segundo experimento fijó la búsqueda en una única solución ($M = 1$) y escaló exponencialmente el tamaño del espacio de búsqueda $N \in \{8, 16, 32, 64, 128\}$, correspondiente a circuitos de 3 a 7 qubits. Se ejecutó la misma FoM a través del *framework* en distintos entornos.

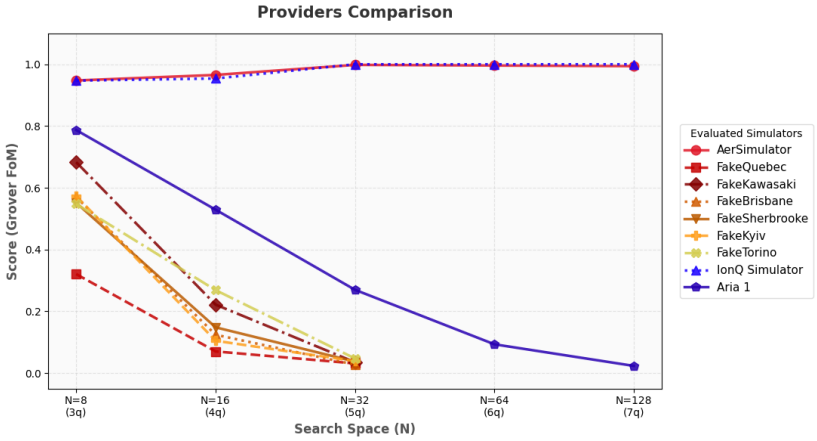


Fig. 4. Comparación de scores a través de distintos backends

En primer lugar, las líneas de los simuladores ideales confirman el límite teórico de Grover, convergiendo en valores prácticamente idénticos. Como se observa en la Figura 4, para $N = 8$, la simulación ideal arroja un *score* de aproximadamente 0.960. Esto no es un artefacto de la simulación, sino la manifestación del *overshoot* determinista de Grover: al estar obligados a aplicar un número entero de iteraciones ($R = 2$), el estado cuántico rota ligeramente más allá del eje ortogonal de las soluciones. De igual forma, en $N = 32$, $R = 4$ iteraciones caen casi milimétricamente sobre el eje vertical, alcanzando un valor

asintótico de 0.999. Cualquier valor por debajo de esta línea base aísla matemáticamente el impacto físico o de transpilación del hardware.

Finalmente, se observa una caída de los *scores* en los procesadores de IBM al escalar N, lo que expone empíricamente el costo de la topología para algoritmos altamente entrelazados. Al transicionar a $N = 32$ (5 qubits), el *score* colapsa a valores marginales (≈ 0.040). Esta degradación no se explica únicamente por las tasas de error estáticas de las compuertas, sino por el costo de descomposición originado por la inserción masiva de compuertas de 2 qubits necesarias para conectar qubits lejanos durante el cómputo de las operaciones `mcx`. Estos resultados demuestran que Qonscious permite a los desarrolladores cuantificar el rendimiento esperado de un procesador bajo modelos de ruido derivados de hardware físico frente a una carga de trabajo algorítmica, traduciendo diferencias topológicas complejas en un score unificado y comparable.

6 Conclusión

La implementación del GroverFigureOfMerit mediante adaptadores polimórficos demuestra ser una solución eficaz para abordar la heterogeneidad en la era NISQ. Al abstraer las particularidades de APIs y conjuntos de compuertas nativas, las FoMs habilitan comparaciones directas y estandarizadas entre proveedores con arquitecturas físicas distintas. Los resultados confirman que la métrica propuesta es sensible a las diferencias de ruido y topología entre backends, demostrando su capacidad de discriminar bajo condiciones de ruido realistas derivadas de procesadores físicos. Esto valida la viabilidad de evaluar procesadores heterogéneos bajo una métrica común y facilita la selección informada de *backends*. La validación en QPUs reales constituye el paso de confirmación natural de estos hallazgos. Más allá de la mera caracterización, las FoMs en Qonscious optimizan recursos al condicionar la ejecución en tiempo de ejecución, lo que puede reducir significativamente costos en entornos de computación cuántica en la nube. El código fuente y los experimentos reproducibles están disponibles en https://github.com/lifia-unlp/qonscious, invitando a la comunidad a adoptar y validar este enfoque en hardware real.

Es importante destacar que la elección del algoritmo de Grover obedece estrictamente a su utilidad como herramienta de caracterización (*stress test*) y no a la búsqueda de una ventaja algorítmica. Si bien el debate sobre la utilidad práctica y la ventaja de Grover a corto plazo en hardware ruidoso sigue abierto, su densa estructura de oráculos multi-controlados y operadores de difusión lo convierte en un instrumento útil para someter a prueba las capacidades de enrutamiento y la coherencia del hardware, independientemente de su viabilidad actual para resolver problemas de búsqueda en el mundo real. Sin embargo, la experimentación también expuso un cuello de botella en el ecosistema cuántico: el costo de la transpilación. Actualmente, delegar la descomposición de compuertas complejas (como `mcx`) a los compiladores por defecto de cada proveedor genera una sobrecarga de profundidad topológica severa, especialmente en arquitecturas de conectividad restringida. Si bien sería posible mitigar esto codificando manualmente el algoritmo utilizando las compuertas nativas óptimas para cada proveedor específico, este enfoque rompe el paradigma agnóstico, anula la escalabilidad del código y resulta altamente costoso en términos de tiempo de desarrollo. Esto subraya la necesidad de

abstracciones más inteligentes que medien entre el circuito lógico y las restricciones físicas.

7 Trabajos Futuros

A partir de estos hallazgos, se delinean las siguientes direcciones para trabajos futuros. En primer lugar, investigar el desarrollo o integración de una representación intermedia (IR) o un módulo de transpilación agnóstico permitiría una traducción optimizada a compuertas nativas, minimizando el costo topológico de forma transversal sin obligar al desarrollador a escribir código específico para cada proveedor. Por otro lado, la validación sobre procesadores físicos reales permitiría contrastar los resultados obtenidos bajo modelos de ruido derivados con el comportamiento dinámico real del hardware, incluyendo fenómenos como *crosstalk*, fluctuaciones térmicas y variabilidad entre ejecuciones que los modelos estáticos no capturan completamente. Finalmente, a partir de los resultados positivos obtenidos, expandir el catálogo de FoMs hacia algoritmos con patrones de entrelazamiento distintos a los de Grover resulta imperativo. Implementar FoMs basadas en algoritmos híbridos variacionales (VQE, QAOA) o en subrutinas de ventaja comprobada como la Transformada Cuántica de Fourier (QFT) permitiría evaluar si la agnosticidad del enfoque se sostiene frente a estructuras de circuito diversas, a la vez que se consolida una métrica más certera del potencial real de aceleración cuántica de cada arquitectura.

References

Gonski, J., Ott, J., Abbaszadeh, S., Addepalli, S., Cremonesi, M., Dickinson, J., Guglielmo, G. D., Ertorer, E. Y., Gray, L., Herbst, R., Herwig, C., Hong, T. M., Maier, B., Makou, M. B., Miller, D., Neubauer, M. S., Peña, C., Rankin, D., Seon-Hee, … Yu, F. (2026). Machine learning on heterogeneous, edge, and quantum hardware for particle physics (ml-hequpp). https://arxiv.org/abs/2602.22248

Grover, L. K. (1996). A fast quantum mechanical algorithm for database search. *Proceedings of the 28th Annual ACM Symposium on Theory of Computing (STOC '96)*, 212–219. https://doi.org/10.1145/237814.237866

IBM Quantum. (2026). *Qiskit api reference: Qiskit.providers.backendv2*. Retrieved March 13, 2026, from https://quantum.cloud.ibm.com/docs/api/qiskit/qiskit.providers.BackendV2

IonQ. (2026). *Using native gates with qiskit*. Retrieved March 13, 2026, from https://www.ionq.com/resources/using-native-gates-with-qiskit

Jiang, Y.-Y., Deng, C., Fan, H., Li, B.-Y., Sun, L., Tan, X.-S., Wang, W., Xue, G.-M., Yan, F., Yu, H.-F., Zhang, Y.-S., Zhang, Y.-R., & Zou, C.-L. (2025). Advancements in superconducting quantum computing. *National Science Review*, *12*(8), nwaf246. https://doi.org/10.1093/nsr/nwaf246

Lammers, M. G., Holik, F. H., & Fernández, A. (2025). Quantum resource management in the nisq era: Challenges, vision, and a runtime framework. *arXiv preprint arXiv:2508.19276*. https://doi.org/10.48550/arXiv.2508.19276

Lubinski, T., Johri, S., Varosy, P., Coleman, J., Zhao, L., Necaise, J., Baldwin, C. H., Mayer, K., & Proctor, T. (2023). Application-oriented performance benchmarks for quantum computing. *IEEE Transactions on Quantum Engineering*, *4*, 1–32. https://doi.org/10.1109/TQE.2023.3253761

Manor, S., Kumar, M., Behera, P., Khalid, A., & Zeng, O. (2025). Grade: Grover-based benchmarking toolkit for assessing quantum hardware. *arXiv preprint arXiv:2504.19387*. https://doi.org/10.48550/arXiv.2504.19387

Nation, P. D., Saki, A. A., Brandhofer, S., Bello, L., Garion, S., Treinish, M., & Javadi-Abhari, A. (2025). Benchmarking the performance of quantum computing software for quantum circuit creation, manipulation and compilation. *Nature Computational Science*, *5*(5), 427–435. https://doi.org/10.1038/s43588-025-00792-y

Proctor, T., Young, K., Baczewski, A. D., & Blume-Kohout, R. (2025). Benchmarking quantum computers. *Nature Reviews Physics*, *7*, 105–118. https://doi.org/10.1038/s42254-024-00796-z

Quantinuum. (2026). *Quantinuum nexus user guide: Backend snapshots*. Retrieved March 13, 2026, from https://docs.quantinuum.com/nexus/user_guide/concepts/backend_snapshots.html

QuEra Computing. (2026). *Bloqade reference: Hardware capabilities*. Retrieved March 13, 2026, from https://bloqade.quera.com/latest/analog/reference/hardware-capabilities/

Rigetti Computing. (2026). *Quil-t getting started: Get_calibration_program()*. Retrieved March 13, 2026, from https://pyquil-docs.rigetti.com/en/v4/quilt_getting_started.html

Xanadu. (2026). *Xanadu cloud client (xcc) documentation*. Retrieved March 13, 2026, from https://xanadu-cloud-client.readthedocs.io/