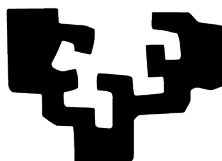

Digital-analog
quantum computing
and
algorithms

Ana Martín Fernández

Supervised by

Mikel Sanz

eman ta zabal zazu



Departamento de Química Física
Facultad de Ciencia y Tecnología
Universidad del País Vasco

April 2023

This document is a PhD thesis developed during the period from October 2018 to March 2023 at the University of the Basque Country.

©2023 by Ana Martín Fernández. All rights reserved.

An electronic version of this thesis can be found at www.nquirephysics.com

Bilbao, April 2023

This document was generated with the 2015 L^AT_EX distribution.

The L^AT_EX template is adapted from a [template](#) by Iagoba Apellaniz.

*A mi familia, la que toco y la que pienso;
a mis amigos, distribuidos en diferentes lugares y momentos;*

y a Hanse, silenciosa pero siempre presente

*To my family, tangible and intangible;
to my friends, spread over the space and time*

... and to my cat, best roommate I can ask for

If we are satisfied with things as they are, nothing will ever be discovered.

Seneca
Paraphrase of a passage from "Moral Letters to Lucilius"

Contents

Abstract	i
Resumen	iii
Acknowledgements	ix
List of Publications	xi
Abbreviations and conventions	xiii
1 Introduction	1
1.1 What you will find in this thesis	6
2 Digital–analog quantum computation	9
2.1 Analog, digital, and digital–analog quantum computing. Formal definitions.	10
2.2 Digital–analog quantum computation	11
2.2.1 Enhancing the connectivity of quantum hardware using DAQC . .	19
2.3 Stepwise and banged DAQC	20
2.3.1 Error estimation	21
2.4 Performance of the DAQC compared to other paradigms	23
2.4.1 Theoretical description of the effect of noise sources in quantum systems.	23
2.4.2 Error mitigation in DAQC	27

3	Digital-analog quantum Fourier transform and phase estimation	29
3.1	Quantum Fourier transform	30
3.1.1	Description of the QFT algorithm and its DAQC implementation	30
3.1.2	Realistic implementation with experimental errors	33
3.1.3	Error mitigation in DA-QFT	35
3.2	Quantum phase estimation	38
3.2.1	Description of the QPE algorithm and its DAQC implementation	38
3.2.2	Realistic implementation with experimental errors	39
4	Digital-analog algorithm for solving linear systems of equations	43
4.1	Quantum algorithm for solving linear systems of equations	44
4.2	Proposal for digital implementation: the ancilla quantum encoding	46
4.3	Digital-analog implementation	49
4.4	Co-designed processor architecture for the HHL algorithm	51
5	Digital-analog quantum approximate optimization algorithm	53
5.1	The Quantum Approximate Optimization Algorithm	54
5.2	Digital-Analog Quantum Approximate Optimization Algorithm	56
5.3	Performance of the QAOA under different computational paradigms . . .	58
5.3.1	Performance of the bDA-QAOA	58
5.3.2	Variational resilience of DA-QAOA to digital-analog errors . . .	59
5.3.3	Sensitivity to other errors	61
6	Digital-analog simulations using the cross-resonance effect	63
6.1	Deriving the effective cross-resonance Hamiltonian	64
6.1.1	Two qubits	65
6.1.2	N qubits	65
6.2	Digital-analog dynamics from the CR Hamiltonian	66
6.2.1	Synthesis Error	66
6.2.2	Hamiltonian toggling	68
6.2.3	Two-dimensional generalization	68
6.3	Many-body compilation	70
6.3.1	Ising model	70
6.3.2	XY model	73
6.3.3	Heisenberg model	78
6.4	Practical Implementation	81
6.5	Discussions	82
7	Conclusions and perspectives	85
	Appendices	89

A	Additional details and proofs regarding the fundamentals of DAQC	91
A.1	Universality	91
A.2	Negative times in the Ising model	92
A.3	Independent coupling parameters in XZ model	93
A.4	M-body Hamiltonians	94
B	Detailed calculations regarding the cross-resonance effect	97
B.1	CR Hamiltonian	97
	B.1.1 Two-qubit case	97
	B.1.2 N-qubit case	100
B.2	Synthesis errors	101
B.3	Unitary transformation to the Quad Frame	102
B.4	2D XY model	104
	Bibliography	107

Abstract

The field of quantum computing has experienced significant growth in recent years, driven by breakthroughs in the development of new algorithms, the discovery of novel quantum phenomena, and the construction of increasingly larger and more stable quantum systems. Despite these advances, the implementation of quantum algorithms on real hardware remains a significant challenge, as quantum systems are prone to errors and decoherence that can limit their computational power.

In this context, the digital-analog quantum computational paradigm has emerged as a promising approach to building large-scale quantum computers and implementing quantum algorithms for practical applications. The digital-analog paradigm combines the strengths of both digital and analog quantum computing, allowing for efficient implementation of quantum algorithms with improved accuracy and error mitigation.

The digital-analog paradigm is particularly well-suited for implementing quantum algorithms that require a large number of quantum gates and measurements or that involve complex entanglement structures. By leveraging the robustness of analog quantum simulation and the flexibility of digital quantum computation, the digital-analog paradigm provides a powerful framework for designing and implementing quantum algorithms on real hardware.

In this thesis, we provide a detailed analysis of the digital-analog quantum computational paradigm and its application to the implementation of quantum algorithms. We first present a comprehensive overview of the digital-analog paradigm, including a toolkit for Hamiltonian simulation that enables the implementation of quantum algorithms with improved efficiency and accuracy. In addition, we present a noise model

to compare the performance of the digital and digital-analog quantum computation frameworks.

Next, we focus on the implementation of four relevant quantum algorithms using both the digital and digital-analog quantum computational frameworks. These algorithms include the quantum Fourier transform, the quantum phase estimation, the Harrow-Hassidim-Lloyd algorithm for solving linear systems of equations, and the quantum approximate optimization algorithm. By comparing the performance of these algorithms using both frameworks, we demonstrate that the digital-analog paradigm offers significant advantages over the digital approach, particularly in terms of reduced error rates and improved scalability.

Finally, we explore how the cross-resonance effect present in superconducting circuits can be used for digital analog quantum simulations. Our results show that this effect can be leveraged to implement Hamiltonian simulations with high accuracy and efficiency, further demonstrating the potential of the digital-analog paradigm for practical quantum computing applications.

In conclusion, our thesis demonstrates the promise of the digital-analog quantum computational paradigm for implementing quantum algorithms on real hardware. The digital-analog paradigm offers significant advantages over the purely digital approach, including improved scalability, reduced error rates, and greater flexibility in circuit design. We believe that this work will contribute to ongoing efforts to develop practical quantum computing technologies and pave the way for future research in the field. As the field of quantum computing continues to evolve, the exploration of new methods for improving the accuracy and efficiency of quantum computations will be crucial for advancing our understanding of quantum computing and identifying new approaches for implementing quantum algorithms.

Resumen

En las últimas décadas, la computación cuántica ha evolucionado de un concepto abstracto a una tecnología con el potencial de revolucionar la computación tal como la conocemos. Los avances tecnológicos y el desarrollo de la teoría han permitido el asentamiento este campo en distintas áreas de la ciencia. En esta tesis, exploramos el paradigma computacional cuántico digital-analógico y su potencial para implementar algoritmos cuánticos en los dispositivos actualmente disponibles. Para ello, partimos de la extensa investigación realizada previamente en el campo y contribuimos a ella proponiendo enfoques novedosos para resolver problemas computacionales utilizando la computación cuántica. Nuestro objetivo es contribuir al campo y avanzar en nuestra comprensión del potencial de la computación cuántica.

Fue en la década de los 80 cuando, Beniof, Manin y Feynman, entre otros, señalaron que existían dificultades esenciales a la hora de simular sistemas cuánticos debido al crecimiento exponencial de sus grados de libertad. Sugirieron entonces la posibilidad de construir ordenadores que estuvieran basados en los principios de la mecánica cuántica para así solventar esas limitaciones. Más concretamente, propusieron emplear sistemas cuánticos completamente controlables para codificar la dinámica de sistemas cuánticos de muchos cuerpos, ya que la complejidad de este tipo de sistemas crece exponencialmente con la dimensión del mismo.

Una década más tarde, en 1996, Lloyd demostró que si los sistemas cuánticos evolucionan a través de interacciones locales, entonces un ordenador cuántico puede simularlos eficientemente. Argumentó que la aplicación de operaciones que preservan la coherencia –tanto como sea posible–, puede considerarse como una activación y desactivación de Hamiltonianos. Cada operación experimental mueve el sistema

a lo largo de cualquier evolución temporal, representada por la operación unitaria $U = e^{iAt}$. Ese resultado implica que existe una variedad de sistemas cuánticos, a los que se les puede aplicar un conjunto de operaciones experimentales simples, pudiendo así simular otros sistemas cuánticos.

Con estos resultados e inspirados por el desarrollo de la computación clásica, se propuso la idea de la computación digital cuántica basada en puertas. Para ello, resultó fundamental el concepto de *conjunto de puertas universales*, que no es más que un set de puertas a las que puede reducirse, con precisión arbitraria, cualquier operación posible en un ordenador cuántico. Esto, junto con el desarrollo de la corrección cuántica de errores, condujo a la computación cuántica basada en puertas tal y como la conocemos hoy en día.

Las puertas utilizadas en los ordenadores cuánticos son pulsos digitales que pueden rotar un solo qubit (puerta de un solo qubit) o activar la interacción entre dos qubits (puerta de dos qubits), según su aplicación. Al aplicar estas puertas de forma secuencial, se pueden ejecutar diferentes algoritmos cuánticos, según la secuencia escogida. La primera demostración concreta de que un ordenador cuántico podría superar a uno clásico fue realizada por Deutsch. A esto le siguió la extensión realizada por Deutsch y Jozsa, que mostró por primera vez una brecha similar para el escalado del tiempo requerido para resolver un problema. Poco después de eso, se propusieron otros algoritmos que ofrecieron una mayor velocidad que sus análogos clásicos, incluido el algoritmo de Shor para factorizar números primos y el algoritmo de Grover para buscar en bases de datos no estructuradas. Desde entonces, se han desarrollado muchos algoritmos en busca de ventajas cuánticas, ampliando enormemente el repertorio de algoritmos cuánticos.

Ha habido varias propuestas de plataformas físicas para realizar computación cuántica. El primer esquema de implementación fue presentado en 1995 por J. I. Cirac y P. Zoller, quienes propusieron puertas lógicas cuánticas utilizando iones atrapados. Posteriormente, se propusieron otros métodos físicos para lograr la computación cuántica, como la resonancia magnética nuclear (RMN), los qubits de espín y los qubits superconductores. Desde entonces, ha habido un progreso experimental significativo en el campo.

La computación cuántica puede llevarse a cabo en estas diferentes plataformas físicas, siempre que éstas cumplan una serie de criterios conocidos como criterios DiVincenzo. Estos criterios establecen que la plataforma utilizada debe: ser un sistema físico escalable con qubits bien caracterizados; tener la capacidad de inicializar el estado de los qubits a un estado fiduciario simple; tener tiempos de decoherencia suficientemente largos; tener un conjunto universal de puertas cuánticas y una capacidad de medición específica de qubit.

El primer resultado experimental de un algoritmo cuántico se logró en un ordenador cuántico de RMN de 2 qubits donde J. A. Jones y M. Mosca resolvieron el problema de Deutsch. Poco después, I. L. Chuang y sus colaboradores realizaron la primera ejecución del algoritmo de Grover también en un ordenador cuántico de RMN. Un ordenador cuántico de RMN utiliza los estados de espín del núcleo de una molécula como qubits. Estas moléculas podrían estar en una muestra líquida –RMN

de estado líquido– o en una muestra sólida –RMN de estado sólido–, por ejemplo, red de diamante de vacancia de nitrógeno, siendo esta última una mejor aproximación mediante computación cuántica. Desde 1988, los logros experimentales de los ordenadores cuánticos de RMN crecieron rápidamente, con resultados interesantes en la implementación de algoritmos cuánticos y la creación de ordenadores cuánticos RMN híbridos. A pesar de este ritmo de progreso, se reconoció desde el principio que dichos dispositivos RMN se enfrentaban a un pobre escalado de la relación señal/ruido. Además, en 2002, N. C. Menicucci y C. M. Caves demostraron que todos los experimentos en RMN de muchos cuerpos en estado líquido realizados hasta ese momento presentaban poco o ningún entrelazamiento, lo que llevó a afirmar que dichos experimentos podrían haber sido meras simulaciones clásicas de un sistema cuántico. Esta limitación se reduce a la RMN de estado líquido y todavía es posible que la RMN de estado sólido se pueda utilizar para la computación cuántica. Estos resultados llevaron a la investigación en computación cuántica a reenfocarse más en otras tecnologías, como iones atrapados, puntos cuánticos, fotónica y circuitos superconductores, y la idea de computación cuántica de RMN fue relegada progresivamente.

Estos avances abrieron la puerta a las *simulaciones analógicas*, que se basan en sistemas cuánticos controlables cuyas dinámicas son similares a las de los sistemas que se simulan. Para lograr este objetivo, se desarrollaron dispositivos cuyo único propósito es estudiar las dinámicas de estos sistemas, conocidos como simuladores cuánticos analógicos. Estos dispositivos han permitido obtener resultados relevantes en una amplia gama de campos, incluido el modelo cuántico de Rabi, la física de Casimir, dinámica de fluidos, química cuántica y física de la materia condensada, entre muchos otros.

Desde entonces y durante la última década, se han logrado avances significativos, llegando incluso a alcanzar uno de los mayores hitos de la computación cuántica: la resolución de un problema que resulta fuera del alcance de los ordenadores clásicos. Este avance, conocido como *ventaja cuántica*, fue logrado por primera vez por Google en 2019. En su estudio, F. Arute *et al.* utilizó un muestreo de circuito aleatorio en su procesador sycamore de 53-qubit. Posteriormente, ha habido numerosos experimentos que demuestran la ventaja cuántica en una variedad de sistemas, incluidos los qubits superconductores y los qubits fotónicos. En esta tesis nos hemos centrado en los circuitos superconductores, pero es importante tener en cuenta que lo que presentamos en este sentido puede aplicarse a otras plataformas, como las mencionadas anteriormente.

La idea detrás de la *computación cuántica digital-analógica* (DAQC por sus siglas en inglés). es explotar la interacción natural presente en el procesador cuántico y usarla junto con puertas de un solo qubit para controlar el sistema y simular una dinámica deseada. Este nuevo paradigma computacional surgió como una alternativa al conocido como *computación cuántica digital* (DQC por su traducción al inglés). Una de las principales diferencias entre los paradigmas DAQC y DQC es que, en este último, la dinámica que se quiere simular se descompone en puertas de uno y dos qubits, en lugar de en términos de la interacción hamiltoniana del sistema. Las puertas de dos qubits requieren que la interacción entre los qubits del sistema que

no están involucrados en dicha operación, se atenúe lo más posible, idealmente debe ser casi nula. Esta atenuación parcial de la interacción inherente del sistema es una operación “antinatural”, que genera una fuente de ruido relevante. No obstante, cabe destacar que el enfoque digital en la computación cuántica ha logrado resultados relevantes en varias áreas, como el aprendizaje automático cuántico, finanzas, sistemas cuánticos abiertos, química cuántica, o teoría cuántica de campos. Sin embargo, dado que estamos en la era de la computación cuántica de escala casi intermedia (NISQ por su siglas en inglés), el paradigma de la computación cuántica digital se ve afectado por las limitaciones experimentales de los dispositivos NISQ. Los dispositivos NISQ son lo más cercano que estamos hoy a los ordenadores cuánticos. Se dice que son “ruidosos” porque estos dispositivos no tienen corrección de errores y el ruido limita su poder de cómputo. Aún así, representan una tecnología puntera que nos permite explorar los sistemas cuánticos de muchas partículas en un régimen que nunca antes había sido accesible experimentalmente.

Para superar esta situación e inspirado en el trabajo de Dodd *et al.*, en 2020 se propone por parte de Parra *et al.* avanzar en este tipo de dispositivos mediante el uso de DAQC. Este enfoque combina la robustez de la simulación cuántica analógica con la flexibilidad de la computación cuántica digital. En el primer artículo, los autores demuestran de manera elegante que es posible generar cualquier Hamiltoniano entrelazado de dos cuerpos usando cualquier otro Hamiltoniano entrelazado arbitrario como generador. En su trabajo, Parra *et al.* van un paso más allá y proponen que, a cambio de fijar el hamiltoniano de origen, es posible simular cualquier hamiltoniano entrelazado arbitrario de múltiples cuerpos. Además, proveen instrucciones explícitas sobre cómo usar este paradigma para simular diferentes familias de hamiltonianos. En base a este trabajo, a lo largo de estos cuatro años, hemos desarrollado propuestas para la implementación de algoritmos y subrutinas cuánticas ubicuas utilizando el paradigma DAQC antes mencionado.

La estrategia que seguimos para desarrollar nuestro trabajo fue proponer algoritmos en los que pudiéramos apoyarnos para construir los siguientes, como una matryoska. Por lo tanto, comenzamos con la transformada cuántica de Fourier debido a su relevancia como paso clave en una multitud de algoritmos. El siguiente paso lógico fue trabajar en la estimación cuántica de fase. Finalmente, uniendo estas dos subrutinas e incluyendo una adicional, pudimos completar una propuesta para implementar el algoritmo Harrow-Hassidim-Lloyd para resolver sistemas de ecuaciones lineales. Además, como muchos algoritmos variacionales encajan con el paradigma DAQC, propusimos una implementación para uno de los algoritmos variacionales que han despertado interés más allá de las fronteras de la ciencia, permeando en el campo de la industria debido a sus aplicaciones en logística, estamos hablando del algoritmo de optimización cuántica aproximada (QAOA por sus siglas en inglés). Finalmente, inspirados por las limitaciones físicas de los dispositivos NISQ, consideramos procesadores cuánticos de circuitos superconductores y extendimos el efecto de resonancia cruzada de una interacción de dos qubits a un Hamiltoniano analógico por sí solo. De esa manera, demostramos que es posible generar una amplia gama de hamiltonianos al aumentar la dinámica hamiltoniana analógica con puertas de un solo qubit.

Nuestro trabajo ha tenido un impacto significativo en varios grupos de diferentes universidades, quienes han adoptado algunas de nuestras propuestas para sus investigaciones en el campo de la computación cuántica. En 2020, Babukhin *et al.* utilizaron el efecto de la resonancia cruzada en procesadores cuánticos basados en circuitos superconductores para realizar simulaciones DAQC. Sus resultados demuestran que un enfoque DA híbrido permite una simulación eficiente de la dinámica de un modelo Ising de campo transversal sin la necesidad de puertas estándar de dos qubits, que presentan desafíos importantes en la computación cuántica. Al aprovechar el efecto de resonancia cruzada en los procesadores cuánticos superconductores de IBM, simularon con éxito la dinámica Trotterizada de los clústeres de espín y compararon los resultados con los obtenidos mediante la computación digital convencional basada en puertas de dos qubits. Sus hallazgos indican que el enfoque DA supera al enfoque digital estándar, lo que ofrece perspectivas prometedoras para las computadoras NISQ a corto plazo. Este trabajo destaca el potencial de mejora adicional con procesadores más especializados y sugiere la viabilidad de una estrategia de digital a analógico para la computación cuántica.

Otro trabajo destacable es el desarrollado por M. Gong *et al.* en el que incluyen una red neuronal cuántica digital-analógica cuya tarea es distinguir la naturaleza de un conjunto de estados cuánticos. Aquí, los autores presentan una técnica novedosa para categorizar estados cuánticos de muchos cuerpos, llamada detección neuronal cuántica. Usando un procesador cuántico superconductor de 61 qubits, demuestran la efectividad de su método para clasificar dos fases distintas de la materia: ergódica y localizada. El corazón de su enfoque de detección neuronal cuántica es una red neuronal cuántica analógica digital (QNN), diseñada para distinguir entre estos estados cuánticos. El QNN toma como entrada un circuito cuántico variacional analógico digital de doble capa, y se muestra que es eficiente en términos de hardware y universalmente capaz de computación cuántica, como lo demostraron previamente Parra *et al.*. Este nuevo método representa un paso prometedor hacia la comprensión y caracterización de las propiedades complejas de los sistemas cuánticos de muchos cuerpos.

También ha habido interesantes propuestas a nivel teórico, una de ellas relacionada con el campo de los algoritmos cuánticos variacionales. En su trabajo, A. Michel *et al.* proponen usar DAQC para estimar la energía del estado fundamental de varias moléculas usando un algoritmo de solución propia cuántica variacional analógica digital (VQE). El estudio se centra en implementar el algoritmo VQE en un simulador cuántico de Rydberg para moléculas H_2 , LiH y BeH_2 . Los autores consideran las limitaciones de la plataforma, como la restricción de la acción local a la preparación y medición del estado inicial y la implementación de la evolución temporal hamiltoniana global. A pesar de esto, destacan que la interacción digital-analógica se puede lograr mediante el uso de una caja de herramientas, como se demostró en un estudio anterior llevado a cabo por S. Notarnicola *et al.*, que enfatiza el potencial de los dispositivos de átomos neutros de próxima generación.

Adicionalmente, se ha propuesto una arquitectura de circuitos superconductores para realizar DAQC, lo que nos hace vislumbrar un futuro donde DAQC juega un

papel importante en el desarrollo del campo de la computación cuántica. En este estudio, los autores se enfocan en el avance de DAQC como un enfoque prometedor hacia el desarrollo de computadoras cuánticas de próxima generación. Los autores proponen un diseño de circuito superconductor para DAQC, que consta de una cadena de qubits de carga acoplados a través de dispositivos superconductores de interferencia cuántica (SQUID por sus siglas en inglés) conectados a tierra. Este diseño permite la manipulación de interacciones individuales entre qubits vecinos más cercanos, produciendo interacciones diferentes e independientes, como un término de intercambio o doble excitación/desexcitación. Mediante el uso de este diseño, los autores pretenden expandir los mapeos algorítmicos de DAQC y la variedad de geometrías y topologías de computadoras cuánticas accesibles. La flexibilidad de este diseño proporciona una gran familia de hamiltonianos multicuerpo analógicos, adecuados para implementaciones eficientes de protocolos DAQC. Este trabajo es un paso significativo hacia el objetivo de alcanzar computadoras cuánticas tolerantes a fallas y sistemas complejos de computación en plataformas cuánticas. Nos acerca al objetivo de lograr una ventaja cuántica con menos recursos algorítmicos y de hardware, contribuyendo a los esfuerzos continuos para avanzar en el paradigma DAQC.

Relacionado con la simulación de sistemas cuánticos de muchos cuerpos, ha habido una interesante propuesta de la mano de L. C. Céleri *et al.* En su trabajo, los autores presentan un algoritmo cuántico DA que puede simular una amplia gama de Hamiltonianos fermiónicos, incluido el modelo de Fermi-Hubbard. La simulación de sistemas cuánticos de muchos cuerpos es una tarea desafiante que requiere una cantidad significativa de recursos, que aumentan exponencialmente con el tamaño del sistema. Esto se vuelve aún más difícil con los sistemas fermiónicos debido a las interacciones no locales causadas por la naturaleza antisimétrica de la función de onda fermiónica. Sin embargo, los autores demuestran cómo los métodos DA permiten que los algoritmos cuánticos se ejecuten de manera más eficiente que sus contrapartes digitales al hacer un mejor uso del tiempo de coherencia. Además, en su trabajo, los autores proporcionan una implementación práctica de estas técnicas utilizando una arquitectura de baja conexión para modelos fermiónicos específicos.

Después de cuatro años de intensa investigación, hemos desarrollado un catálogo completo de algoritmos diseñados específicamente para DAQC. Nuestro trabajo ha sentado una base sólida para que futuros investigadores amplíen y exploren más este apasionante campo. Creemos que los algoritmos que hemos desarrollado serán útiles para aquellos que deseen realizar investigaciones en DAQC, y somos optimistas de que esta tesis servirá como un recurso valioso para futuros estudios. Confiamos en haber contribuido al avance del campo y esperamos ver qué progresos adicionales se pueden lograr.

Acknowledgements

“Cuanto más te acercas a lo esencial, menos puedes nombrarlo”

- Rosa Montero, La ridícula idea de no volver a verte

Nothing comes from nothing and I know that I never walked alone on the path that has brought me here. The completion of this thesis has been a journey filled with challenges, triumphs, and unforgettable experiences. It is with immense gratitude that I acknowledge the individuals and institutions that have supported me throughout this process. Without their guidance, encouragement, and unwavering belief in my abilities, this accomplishment would not have been possible. Their support has been invaluable and has contributed greatly to my personal and professional growth.

I want to start by thanking those who trusted me when I was just an unknown master’s student who had barely been in Bilbao for a few months. Kike Solano invited me to test this research, Mikel Sanz helped me shape it and Iñigo Egusquiza gave me time whenever I needed it. I am enormously grateful to Mikel Sanz for the trust he has always placed in me, as well as for everything he has taught me over the years.

I would like to extend my sincere gratitude to the following individuals and institutions for their support throughout my Ph.D. research. Firstly, I would like to thank Shanghai University for hosting me for the first two months at the beginning of my research journey. I am also grateful to Andreas Wallraff, and Christopher Eichler

for hosting me during my visit to their group in ETH Zurich. In particular, I would like to thank them for their hospitality and kindness. During my research, I had the privilege of collaborating with the group of Frank Wilhelm-Mauch at Saarland University. This collaboration also included David Headley and Thorge Müller, and their support and insights were invaluable to the success of my work. I would also like to express my gratitude to Pavel Lougovski, and Eugene F. Dumitrescu, from the Oak Ridge National Laboratory for their work and support and for hosting us at Berkeley Labs. I will always cherish the sundae birthday dessert Eugene arranged for me. Finally, I would like to thank Göran Johansson and Laura García Álvarez, from Chalmers University, for their generosity and hospitality in hosting me and my colleagues during our stay in Göteborg. I met many wonderful people during my stay, including Pontus, David, and Tom, who helped me learn a great deal about physics and different ways of working.

Special attention goes to the people that have surrounded me every single day: my colleagues from both QUTIS and NQUIRE groups. To my colleagues in my own research group, thank you for your support and for being a constant source of inspiration and motivation. I have a special mention to G. Gatti, who, unexpectedly, has been the more consistent travel partner I ever had. We have been traveling together since we both started our Ph.D. journey and I am glad that we can share all of those scientific and personal memories. This thing about mentioning one by one the people that have made an impact on my journey can be easily converted to torture, so please, accept my extended gratitude to all of you.

On a personal level, I am deeply grateful for the unwavering support and encouragement of my friends and family. They have been my pillars of strength, cheering me on from the sidelines and always believing in me, even when I doubted myself. I will always be grateful for their love and support, and I hope to make them proud of my achievements. I wish all of my family and friends could read these lines, but I am aware that interdimensional travel is not available for now, so I content myself with being aware of the great impact that they all have had on the researcher that I am today.

List of Publications

This thesis is based on the following publications and preprints:

1. [A. Martin](#), L. Lamata, E. Solano, and M. Sanz,
Digital-analog quantum algorithm for the quantum Fourier transform,
[Physical Review Research](#) **2**, 013012 (2020).
2. D. Headley, T. Müller, [A. Martin](#), E. Solano, M. Sanz, and F. Wilhelm,
Approximating the Quantum Approximate Optimisation Algorithm,
[Physical Review A](#), **106**, 042446 (2022).
3. T. Gonzalez-Raya, R. Asensio-Perea, [A. Martin](#), L. C. Céleri, M. Sanz, P. Lougovski, and E. F. Dumitrescu,
Digital-Analog Quantum Simulations Using The Cross-Resonance Effect,
[PRX Quantum](#) **2**, 020328 (2021).
4. P. García-Molina, [A. Martin](#), M. Garcia de Andoin, and M. Sanz,
Noise in Digital and Digital-Analog Quantum Computation,
[arxiv: 2107.12969](#) (2021).
5. [A. Martin](#), R. Ibarondo, and M. Sanz
Digital-analog co-design of the Harrow-Hassidim-Lloyd algorithm,
[arxiv: 2207.13528](#) (2022).

Other publications not included in this thesis:

6. [A. Martin](#), B. Candelas, Á. Rodríguez-Rozas, J. D. Martín-Guerrero, X. Chen, L. Lamata, R. Orús, E. Solano, and M. Sanz, *Towards Pricing Financial Derivatives with an IBM Quantum Computer*, [Physical Review Research](#) **3**, 013167 (2021).

Abbreviations and conventions

We use the following abbreviations throughout the thesis

ATA All-to-all

bDAQC Banged digital-analog quantum computing

CR Cross-resonance

DA Digital-analog

DAQC Digital-analog quantum computing

DFT Discrete Fourier transform

DQC Digital quantum computing

HHL Harrow–Hassidim–Lloyd

RWA Rotating-wave approximation

NISQ Near intermediate-scale quantum computing

NMR Nuclear Magnetic Resonance

NN Nearest-neighbor

- QAOA** Quantum approximate optimization algorithm
- QFT** Quantum Fourier transform
- QPE** Quantum phase estimation
- sDAQC** Step-wise digital-analog quantum computing
- SQG** Single-qubit gate
- SQR** Single-qubit rotation
- TQG** Two-qubit gate

We set the reduced Planck constant $\hbar = 1$ throughout the thesis.

1

Introduction

OVER the past few decades, quantum computing has evolved from an abstract concept to a technology with the potential to revolutionize computing as we know it. In this thesis, we explore the digital-analog quantum computational paradigm and its potential to implement quantum algorithms on currently available devices. We build upon the extensive research conducted in this field and contribute to it by proposing novel approaches to solving computational problems using quantum computing. By doing so, we aim to make valuable contributions to the field and advance our understanding of the potential of quantum computing.

In the 1980s, Benioff [1], Manin [2], and Feynman [3] recognized the need to use a quantum platform for simulating quantum systems, since that task was unattainable for classical computers due to the exponential growth of degrees of freedom. This was triggered by the aim of finding a computation model that could efficiently simulate any other model of computation. To address this challenge, they proposed to use fully controllable quantum systems to encode the dynamics of many-body quantum systems. The complexity of these systems also grows exponentially with the dimension of the system, making them difficult to simulate classically. By employing a quantum platform, it is possible to achieve a more efficient and accurate simulation of quantum systems, which could lead to a better understanding of their behavior and potential applications.

A decade later, Lloyd showed that if quantum systems evolve through local in-

teractions, then a quantum computer can efficiently simulate them [4]. He argued that the application of operations that –approximately– preserve coherence, can be thought of as turning on and off Hamiltonians. Each experimental operation drives the system along any time evolution, represented by the unitary operation $U = e^{iAt}$. That result implied that a variety of quantum systems, to which a set of simple experimental operations can be applied, can simulate other quantum systems. With these results and inspired by the development of classical computing, the idea of gate-based quantum digital computing was proposed. Fundamental to this was the concept of the universal gate set, a set of gates to which any possible operation in a quantum computer can be reduced with arbitrary precision. This, along with the development of quantum error correction led to gate-based quantum computing.

The gates used in quantum computers are digital pulses that can rotate a single qubit (single-qubit gate) or activate the interaction between two qubits (two-qubit gate), depending on their application. By sequencing these gates in different ways, various quantum algorithms can be executed. The first concrete demonstration that a quantum computer could outperform a classical one was shown by Deutsch [5]. This was followed by the extension made by Deutsch and Jozsa [6], which showed for the first time a similar gap for the scaling of the time required to solve a problem. Shortly after that, other algorithms were proposed that offered speed-up over their classical counterparts, including Shor’s algorithm for prime number factorization [7] and Grover’s algorithm for searching unstructured databases [8]. Since then, many algorithms have been developed in the search for quantum advantage, greatly expanding the repertoire of quantum algorithms [9].

There have been several proposals for physical platforms to perform quantum computing. The first implementation scheme was introduced in 1995 by J. I. Cirac and P. Zoller, who proposed quantum logic gates using trapped ions [10]. Subsequently, other physical methods for achieving quantum computing, such as nuclear magnetic resonance (NMR) [11, 12, 13], spin qubits [14, 15], and superconducting qubits [16], were proposed. Since then, there has been significant experimental progress in the field.

Quantum computing can be carried out on these different physical platforms, as long as they meet a series of criteria known as the DiVincenzo criteria [17]. These criteria establish that the platform used must be a scalable physical system with well-characterized qubits; have the ability to initialize the state of the qubits to a simple fiducial state; have sufficiently long decoherence times; have a universal set of quantum gates and a qubit-specific measurement capability.

The first experimental result of a quantum algorithm was achieved in a 2-qubit NMR quantum computer where J. A. Jones and M. Mosca solved Deutsch’s problem [18]. Shortly after, I. L. Chuang and collaborators performed the first execution of Grover’s algorithm also on an NMR quantum computer [19]. An NMR quantum computer uses the spin states of a molecule nucleus as qubits. These molecules could be in a liquid sample – liquid state NMR – or in a solid sample – solid-state NMR –, being the latter a better approach by means of quantum computing. Since 1988, the experimental achievements of NMR quantum computers grew rapidly, with

interesting results in implementing quantum algorithms and creating hybrid NMR quantum computers [14]. Despite this pace of progress, it was recognized from the beginning that NMR quantum computers face a poorly scaled signal-to-noise ratio. Moreover, in 2002, N. C. Menicucci and C. M. Caves [20] showed that all experiments in liquid state bulk ensemble NMR carried out until then, presented little or no entanglement, which led to the affirmation that liquid-state NMR experiments could have been mere classical simulations of a quantum system. This limitation reduces to liquid-state NMR and it is still possible that solid-state NMR could be used for quantum computing. These results led research in quantum computing to refocus more on other technologies, such as trapped ions, quantum dots, photonics, and superconducting circuits, and the idea of NMR quantum computing was progressively relegated.

The advances in quantum computing have paved the way for *analog quantum simulations* [21, 22], which rely on controllable quantum systems whose dynamics are similar to those of the systems being simulated. Single-purpose devices known as analog quantum simulators have been developed to achieve this goal. These simulators have been used to obtain numerous results in a wide range of fields, including the quantum Rabi Model [23, 24, 25, 26, 27], Casimir physics [28, 29, 30], fluid dynamics [31], quantum chemistry [32], and condensed matter physics [33], among many others.

Since then, significant progress has been made in the last decade, culminating in one of the biggest milestones for quantum computing: solving a problem that is beyond the reach of classical computers. This breakthrough, known as *quantum advantage*, was first achieved by Google in 2019 [34]. In their study, F. Arute *et al.* used a randomized circuit sampling on their 53-qubit sycamore processor. Subsequently, there have been numerous experiments demonstrating quantum advantage in a variety of systems, including superconducting qubits [35, 36] and photonic qubits [37, 38, 39]. In this thesis, we have focused on superconducting circuits, but it is important to keep in mind that what we present along these lines can be applied to other platforms, such as the ones previously mentioned.

The idea behind *digital-analog quantum computation* (DAQC) is to exploit the natural interaction present in the quantum processor and use it together with single-qubit gates to control the system and simulate a desired dynamic. This computational paradigm emerged as an alternative of the widespread digital quantum computational (DQC) one. One of the main differences between DAQC and DQC is that, in the latter, the dynamic that one wants to simulate is decomposed into single- and two-qubit gates, rather than in terms of the interaction Hamiltonian of the system. The two-qubit gates require that the interaction among the qubits of the system that are not involved in such operation, has to be attenuated as much as possible, ideally, it has to be almost null. This partial attenuation of the system’s inherent interaction is a rather “anti-natural” operation, resulting in a relevant noise source. Nevertheless, the digital approach in quantum computing has achieved relevant results in several areas such as quantum machine learning [40, 41], finance [42, 43, 44], open quantum systems [45], quantum chemistry [46], or quantum field theories [47]. However, since we are in

the near intermediate scale quantum computing (NISQ) era [48], the DQC paradigm is affected by the experimental limitations of NISQ devices. NISQ devices are the closest we are today to quantum computers. They are said to be “noisy” because these devices are not error corrected, and the noise limits their computational power. Still, they represent an exciting technology that allows us to explore the complex many-particle quantum systems in a regime that has never been experimentally accessible before [49].

In order to overcome this situation and inspired by the work of Dodd *et al.* [50], it has been proposed by Parra *et al.* in 2020, to get advantage of such devices by using DAQC [51]. This approach merges the robustness of analog quantum simulation with the flexibility of digital quantum computation. In [50], the authors elegantly prove that it is possible to generate any two-body entangling Hamiltonian using any arbitrary entangling Hamiltonian as the generator. In their work, Parra *et al.* go a step further and propose that, in exchange for fixing the origin Hamiltonian, it is possible to simulate any arbitrary multi-body entangling Hamiltonian. Furthermore, they give explicit instructions on how to use this paradigm to simulate different families of Hamiltonians. Based on this work, throughout these four years, we have developed proposals for the implementation of algorithms and ubiquitous quantum subroutines using the aforementioned DAQC paradigm.

The strategy that we followed to develop our work was to propose algorithms on which we could rely to build the following ones, like a matryoshka doll. Thus, we start with the quantum Fourier transform [52] due to its relevance as a key step in a multitude of algorithms. The next logical step was to work on the quantum phase estimation [53]. Finally, joining these two subroutines and including an additional one, we were able to complete a proposal to implement the Harrow-Hassidim-Lloyd (HHL) algorithm to solve linear systems of equations [54]. In addition, as many variational algorithms fit with the DAQC paradigm, we proposed an implementation for the quantum approximate optimization algorithm (QAOA) [55]. Finally, inspired by the physical limitations of NISQ devices, we considered superconducting circuits quantum processors and extend the cross-resonance effect from a two-qubit interaction to an analog Hamiltonian on its own. That way, we showed that it is possible to generate a wide range of Hamiltonians by augmenting the analog Hamiltonian dynamics with single-qubit gates [56].

Our work has had a significant impact on various groups from different universities, who have adopted some of our proposals for their research in the field of quantum computing. In 2020, Babukhin *et al.* [57] showcase the potential of using superconducting circuit quantum processors for DAQC by utilizing the cross-resonance effect in transmon qubits. Their results demonstrate that a hybrid digital-analog (DA) approach allows for efficient simulation of the dynamics of a transverse-field Ising model without the need for standard two-qubit gates, which present significant challenges in quantum computing. By leveraging qubit-qubit crosstalks in IBM superconducting quantum processors, they successfully simulate the Trotterized dynamics of spin clusters and compare the results with those obtained from conventional digital computation based on two-qubit gates. Their findings indicate that the DA approach

outperforms the standard digital approach despite the relatively small crosstalks in IBM quantum processors, offering promising prospects for near-term NISQ computers. This work highlights the potential for further improvement with more specialized processors and suggests the feasibility of a digital-to-analog strategy for quantum computing.

Another remarkable work is the one developed by M. Gong *et al.* [58] in which they include a DA quantum neural network whose task was to distinguish the nature of a set of quantum states. Here, the authors introduce a novel technique for categorizing many-body quantum states, called quantum neuronal sensing. Using a 61-qubit superconducting quantum processor, they demonstrate their method’s effectiveness in classifying two distinct phases of matter: ergodic and localized. The heart of their quantum neuronal sensing approach is a DA quantum neural network (QNN), designed to distinguish between these quantum states. The QNN takes as input a double-layered DA variational quantum circuit, and is shown to be hardware efficient and universally capable of quantum computing, as previously demonstrated by Parra *et al.* in Ref. [51]. This new method represents a promising step towards understanding and characterizing the complex properties of many-body quantum systems.

There have been also interesting proposals at the theoretical level, one of them related to the field of variational quantum algorithms [59]. In their work, A. Michel *et al.* propose to use DAQC to estimate the ground-state energy of several molecules using a DA variational quantum eigensolver (VQE) algorithm. The study focuses on implementing the VQE algorithm on a Rydberg quantum simulator for H_2 , LiH and BeH_2 molecules. The authors consider the platform’s constraints, such as the restriction of the local action to initial state preparation and measurement and the implementation of global Hamiltonian time evolution. Despite this, they highlight that the DA interplay is attainable by using a toolbox, as demonstrated in a prior study [60], emphasizing the potential for next-generation neutral atom devices.

Additionally, it has been proposed a superconducting circuit architecture for performing DAQC [61], which makes us foresee a future where DAQC plays an important role in the development of the quantum computing field. In this study, the authors focus on the advancement of DAQC as a promising approach toward the development of next-generation quantum computers. The authors propose a superconducting circuit design for DAQC, consisting of a chain of charge qubits coupled through grounded superconducting quantum interference devices (SQUIDs). This design allows for the manipulation of individual interactions between nearest-neighbor (NN) qubits, producing different and independent interactions such as an exchange or double excitation/de-excitation term. By using this design, the authors aim to expand the DAQC algorithmic mappings and the variety of accessible quantum computer geometries and topologies. The flexibility of this design provides a large family of analog multibody Hamiltonians, suitable for efficient implementations of DAQC protocols. This work is a significant step towards the goal of reaching fault-tolerant quantum computers and computing complex systems on quantum platforms. It brings us closer to the goal of achieving a quantum advantage with fewer algorithmic and hardware

resources, contributing to the ongoing efforts to advance the DAQC paradigm.

Related to the simulation of quantum many-body systems, there has been an interesting proposal by L. C. Céleri *et al.*. In [62], the authors present a DA quantum algorithm that can simulate a wide range of fermionic Hamiltonians, including the Fermi-Hubbard model. Simulating quantum many-body systems is a challenging task that requires a significant amount of resources, which increase exponentially with the system's size. This becomes even more difficult with fermionic systems due to the nonlocal interactions caused by the fermionic wave function's antisymmetric nature. However, the authors demonstrate how DA methods enable quantum algorithms to run more efficiently than their digital counterparts by making better use of coherence time. Additionally, they provide a practical implementation of their techniques using a low-connected architecture for specific fermionic models.

After four years of intensive research, we have developed a comprehensive catalog of algorithms specifically designed for DAQC. Our work has laid a solid foundation for future researchers to expand upon and explore this exciting field further. We believe that the algorithms we have developed will be useful for those who wish to pursue research in DAQC, and we are optimistic that this thesis will serve as a valuable resource for future studies. We are confident to have contributed to the advancement of the field and look forward to seeing what further progress can be made.

1.1 What you will find in this thesis

In this thesis, we make an in-depth description of the DAQC paradigm and present several implementations of quantum algorithms, tailored for this computational paradigm. The performance of these proposals is compared with the one of the usual digital approach. We discuss its resilience against noise sources, especially those coming from the experimental constraints of NISQ devices. This thesis is structured into seven chapters, including this introduction and a concluding chapter, and it is devoted to the description of several ubiquitous quantum algorithms, using the DAQC approach.

As we mention, in Chapter 2 we present the fundamentals and main techniques for DAQC. Although these techniques may be extended to any resource, we propose to use propagators generated by the ubiquitous Ising Hamiltonian for the analog entangling blocks and provide proof of its universal character. We construct explicit DAQC protocols for efficient simulations of arbitrary inhomogeneous Ising, two-body, and M-body spin Hamiltonian dynamics by means of single-qubit gates and a fixed homogeneous Ising Hamiltonian. Additionally, we compare a sequential approach where the interactions are switched on and off (step-wise DAQC) with an always-on analog interaction interspersed by fast single-qubit pulses (banged DAQC). We

also present a DAQC algorithm to enhance the physical connectivity among qubits coupled by an arbitrary inhomogeneous NN Ising Hamiltonian and generate an arbitrary all-to-all (ATA) Ising Hamiltonian by employing single-qubit rotations. Finally, we introduce the theoretical description of some of the most relevant noise sources present in current superconducting quantum computers, in order to be able to compare the resilience under those noises of both the DQC and the DAQC methods in the following chapters.

In Chapter 3 we introduce an efficient DAQC algorithm to compute the quantum Fourier transform, a subroutine widely employed in several relevant quantum algorithms, such as the quantum phase estimation, which is also presented in this chapter. We show that, under reasonable assumptions about noise models, the fidelity of the quantum Fourier transformation improves considerably using this approach when the number of qubits involved grows. This suggests that, in the NISQ era, hybrid protocols combining digital and analog quantum computing could be a sensible approach to reaching useful quantum supremacy.

In Chapter 4 we use the quantum subroutines that we have described in the previous chapter to propose a DA implementation of the Harrow-Hassidim-Lloyd algorithm to solve linear systems of equations. Additionally, a co-designed architecture is proposed, which allows for a reduction of the number of SWAP gates required for its implementation. Merging a co-design quantum processor architecture with a DA implementation contributes to the reduction of noise sources during the experimental realization of the algorithm.

In Chapter 5 it is shown that the DAQC paradigm is suited to the QAOA due to the algorithm's variational resilience against the coherent errors introduced by the scheme. By performing large-scale simulations and providing analytic bounds for its performance in devices with finite single-qubit operation time, we observe regimes of single-qubit operation speed in which the considered variational algorithm provides a significant improvement over nonvariational counterparts in the DAQC scheme.

In Chapter 6, we examine the feasibility of using superconducting circuit quantum processors, specifically transmon qubits, for DAQC by exploiting the cross-resonance effect. Our aim is to assess the potential of this approach as a solution for implementing DAQC in superconducting circuit quantum processors. To achieve it, we consider superconducting architectures and extend the cross-resonance effect, up to first order in perturbation theory, from a two-qubit interaction to an analog Hamiltonian acting on 1D chains and 2D square lattices which, in an appropriate reference frame, results in a purely two-local Hamiltonian. By augmenting the analog Hamiltonian dynamics with single-qubit gates we show how one may generate a larger variety of distinct analog Hamiltonians. We then synthesize unitary sequences, in which we toggle between the various analog Hamiltonians as needed, simulating the dynamics of Ising, XY , and Heisenberg spin models. Our dynamics simulations are Trotter error-free for the Ising and XY models in 1D. We also show that the Trotter errors for 2D XY and 1D Heisenberg chains are reduced, with respect to a digital decomposition, by a constant factor. In order to realize these important near-term speedups, we discuss the practical considerations needed to accurately characterize

and calibrate our analog Hamiltonians for use in quantum simulations. We conclude with a discussion of how the Hamiltonian toggling techniques could be extended to derive new analog Hamiltonians which may be of use in more complex DA quantum simulations for various models of interacting spins.

Finally, we dedicate a chapter to discuss the overall conclusions of this thesis, and we provide an appendix section with additional material that complements the discussions held during the text. A complete bibliography can be found at the end of this document.

The main focus of this thesis is the DAQC, which is why the most relevant section is the second chapter. There, the protocol bases are established and useful tools are provided so the following chapters are easy to follow. Chapter 2 closely follows sections of the dedicated work carried out by Parra *et al.* in 2020 [51] since this constitutes a foundational piece of the DAQC.

2

Digital–analog
quantum computation

IN this chapter, we present an in-depth description of the DAQC paradigm to simulate any arbitrary Hamiltonian. In the following chapters, these techniques will be extended to implement quantum algorithms. The DAQC paradigm requires the implementation of entangling multipartite evolutions and fast single-qubit gates (SQGs), which universality has already been proven [50, 63, 64, 65, 52]. In this approach, we make use of a sequence of entangling time slices, called analog blocks, and fast single-qubit rotations (SQRs), which comprise our class of digital steps. In this chapter, we also introduce protocols to simulate arbitrary inhomogeneous two-body and M -body spin Hamiltonians. Then, we present the *bang DAQC* (bDAQC) approach, to mitigate the errors that derive from the *stepwise DAQC* (sDAQC) version. Afterwards, we show that it is possible to enhance the connectivity of quantum hardware on the software level with digital-analog control. Finally, we present a theoretical description and models of some of the most relevant noise sources present in current superconducting quantum computers, since they will be employed to compare the performance of the DQC and DAQC paradigms when implementing quantum algorithms.

As we mention in the introduction, we want to follow significantly the work carried out by Parra et al. in 2020 [51]. In their work, the authors established formal

definitions for the main pieces of the DAQC paradigms, as well as smart systematic algorithms to simulate relevant families of Hamiltonians. The rest of the chapter include part of the work that we developed in [52] and [53].

2.1 Analog, digital, and digital-analog quantum computing. Formal definitions.

The terms “digital” and “analog” are commonly utilized in the context of quantum computing, however, it is important to provide precise definitions for these concepts in a formal manner. In the following, we will present clear definitions for these terms as they relate to quantum gates, digital, and analog blocks in N -qubit systems. These definitions are extracted from Ref. [51].

Definition (Quantum gate). *A quantum gate is a fixed unitary evolution U_n , $U_n \in \mathcal{B}((\mathbb{C}^2)^{\otimes n})$.*

Definition (Digital block). *A k -parametric continuous family of unitary operators $U_n(\vec{\phi})$, with $\phi_l \in \mathcal{I}_l(\mathbb{R})$ and $1 \leq l \leq k$, comprises a digital block if it is equivalent to a fixed unitary evolution U_n up to a set of local rotations $W_i(\vec{\phi})$, i.e., $U_n(\vec{\phi}) = \left(\bigotimes_i^n W_i(\vec{\phi}) \right) U_n$.*

It is important to note that digital blocks, which include both parameter-fixed entangling quantum gates and single qubit rotations with arbitrary angles, generate an entanglement that remains the same for all values of the parameters. On the other hand, analog blocks generate entanglement that varies with the values of the parameters.

Definition (Analog block). *We call analog block a k -parameter-dependent entangling unitary evolutions $V_n(\vec{\phi})$ with a semigroup structure $V_n(\vec{\phi}) = V_n(\vec{\phi}_1)V_n(\vec{\phi}_2)$; $\vec{\phi} = \vec{\phi}_1 + \vec{\phi}_2$. For $k = 0$, it obviously reduces to a quantum gate.*

Under these definitions, for instance, $U_n = e^{i\frac{\pi}{4}\sigma_z^1\sigma_z^2}$ is a quantum gate, both $U_n(\phi) = (e^{i\phi_1\sigma_z^1} \otimes e^{i\phi_2\sigma_z^2})e^{i\frac{\pi}{4}\sigma_z^1\sigma_z^2}$ and $W_i(\phi) = e^{i\phi\sigma_z^1}$ are digital blocks, while $V_n(\phi) = e^{i\phi\sigma_z^1\sigma_z^2}$ is an analog block.

When a quantum protocol only employs digital blocks, it is referred to as a digital quantum algorithm. However, if the protocol consists of a single analog block for different values of the parameters, it is referred to as an analog quantum algorithm. Additionally, a protocol that contains both digital and analog blocks is referred to as a digital-analog protocol.

In this paper, we have limited the digital blocks to arbitrary single qubit rotations, which allows us to write the total evolution of the system as $\prod_j \left(U_j(\vec{\phi}_j) \left[\otimes_i W_i^{(j)}(\vec{\alpha}_{ji}) \right] \right)$.

With these concepts formally defined, we can go deeper into the description of the DAQC paradigm.

2.2 Digital-analog quantum computation

In this section, our focus will be on two quintessential models found in quantum computing systems, serving as examples to highlight the DAQC paradigm, though it can be easily extended to other scenarios. Our focus will be on utilizing either a homogeneous nearest-neighbor (NN) or a homogeneous all-to-all (ATA) two-body Ising Hamiltonian as analog blocks. By combining these evolutions with SQRs, we will demonstrate their universality, meaning that any unitary can be simulated with arbitrary accuracy using these resources. We show the protocols, sometimes optimal, to generate increasingly complex families of Hamiltonians.

First, we will detail how to generate arbitrary inhomogeneous two-body Ising Hamiltonians. The subsequent step will be to create an inhomogeneous two-body Hamiltonian, where the terms do not commute, requiring the application of Trotterization to the algorithm. Finally, we will show how to generate an arbitrary M -body Hamiltonian with a polynomial number of resources in terms of the number of spins, for a fixed value of M .

The foundation of the DAQC algorithms presented in this section lies in three key components:

1. The ability to accurately evolve for a specific time t with a Hamiltonian H , while incorporating local rotations to create an evolved Hamiltonian $H' = RHR^\dagger$. The impact of limited precision during the physical implementation of engineered times will also be considered.
2. The utilization of a Trotter decomposition, $e^{-itH} = (e^{-iH_1t/n_T} e^{-iH_2t/n_T})^{n_T} + O(t^2/n_T)$, which is necessary for simulating Hamiltonians with non-commuting terms, such as $H = H_1 + H_2$, where $[H_1, H_2] \neq 0$. The error introduced through this approximation is of second order in the time error Δt . Advanced decompositions, like the symmetrized Trotter decomposition, are also considered and can reduce the error to $O(\Delta t)^3$.
3. The ability to evolve with Hamiltonian λH , where $\lambda > 0$ is a continuous positive parameter. This provides the flexibility to adjust the strength of the Hamiltonian evolution to meet the needs of the simulation.

Ising model

Let us illustrate the problem of implementing the inhomogeneous all-to-all (ATA) two-body Ising model, described by the Hamiltonian $H_{ZZ} = \sum_{j < k}^N g_{ij} \sigma_z^{(j)} \sigma_z^{(k)}$. The matrices $\sigma_{x,y,z}^{(n)}$ denote the application of the Pauli matrices x , y , or z on the n -th qubit. The Hamiltonian describes the interaction between qubits through the coupling parameters g_{ij} . The goal is to simulate the unitary evolution $U_{ZZ}(t) = e^{iH_{ZZ}t}$ of this Hamiltonian using a combination of digital and analog blocks.

For this example, the analog block that will be used is the unitary evolution, $U_{zz}(t)$, of the homogeneous ATA two-body Ising model whose Hamiltonian reads

$$H_{zz} = g \sum_{j < k}^N \sigma_z^{(j)} \sigma_z^{(k)}, \quad (2.1)$$

where g is a fixed coupling strength and the independent time parameter t is used in the calculation of the unitary evolution. We set $\hbar = 1$ in the whole chapter (and thesis).

The digital blocks used will be single qubit rotations around the x -axis, which can be performed with a continuous range of phases $\theta \in [0, 2\pi]$ and are defined as

$$R_x(\theta) = e^{-i\theta\sigma_x/2}. \quad (2.2)$$

only digital blocks employed are single qubit rotations around the x -axis with the continuous range of phases $\theta \in [0, 2\pi]$. The protocol can be easily modified to incorporate a fixed inhomogeneous evolution as an analog block, with Hamiltonian

$$H_{ZZ} = \sum_{j < k}^N g_{jk} \sigma_z^{(j)} \sigma_z^{(k)}. \quad (2.3)$$

Such Hamiltonians, which feature scaling of coupling parameters of the form $\bar{g}_{jk} \propto 1/|j-k|^\alpha$ with $0 < \alpha < 3$, commonly arise in various quantum platforms, such as ion-trap setups [66]. In effective Hamiltonian models, where qubits are coupled through linear multi-mode systems [67, 68], even more complex coupling distributions may arise and can be tailored or designed [69, 70]. A different decomposition into most entangling Hamiltonians and single qubit gates has been shown to be a universal quantum machine by Dodd et al. in Ref. [50]. Under the digital-analog paradigm, the Ising Hamiltonian serves as an example of a universal Hamiltonian, as it enables the construction of a universal ZZ gate [71] between any two arbitrary qubits. This is detailed in Appendix A.1.

The target Hamiltonian evolves according to the unitary

$$U_{ZZ} = e^{it_F H_{ZZ}}, \text{ where } H_{ZZ} = \sum_{j < k} g_{jk} \sigma_z^j \sigma_z^k, \quad (2.4)$$

where t_F is the final time. The objective is to determine a suitable mapping between the expressions $g_{jk}t_F$ and gt_{nm} through the decomposition of the homogeneous time

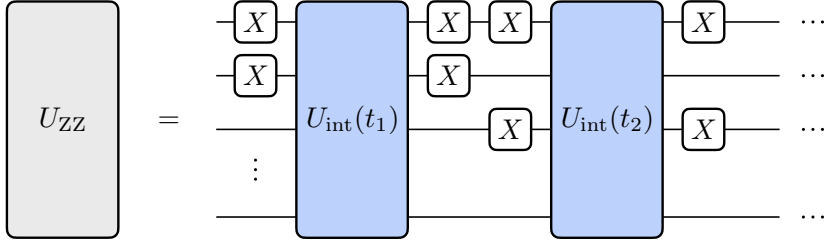


Figure 2.1: **Algorithm to simulate the general inhomogeneous Ising model from a fixed one.** Each analog block consisting on a time step evolution of time t_α is sandwiched by a pair of single qubit gates, X , which refers to π rotation around x -axis, $R_x(\pi)$ applied to qubits (n, m) , with $\alpha(n, m)$. It is possible to use optimal sequences of SQRs to simplify the number of pulses.

evolution into a maximum of $N(N-1)/2$ analog blocks with varying durations t_{nm} , which are bracketed by local rotations $\sigma_x^n \sigma_x^m$,

$$\begin{aligned} H_{ZZ} &= \sum_{j < k}^N g_{jk} \sigma_z^j \sigma_z^k = \frac{g}{t_F} \sum_{j < k}^N \sum_{n < m}^N t_{nm} \sigma_x^{(n)} \sigma_x^{(m)} \sigma_z^{(j)} \sigma_z^{(k)} \sigma_x^{(n)} \sigma_x^{(m)} \\ &= \frac{g}{t_F} \sum_{j < k}^N \sum_{n < m}^N t_{nm} (-1)^{\delta_{nj} + \delta_{nk} + \delta_{mj} + \delta_{mk}} \sigma_z^j \sigma_z^k, \end{aligned} \quad (2.5)$$

as depicted in Fig. 2.1. As $\sigma_x^{(n)} \sigma_z^{(k)} \sigma_x^{(n)}$ is equal to $-\sigma_z^{(n)}$ if $n = k$, and it is equal to $\sigma_z^{(k)}$ otherwise, then,

$$H_{ZZ} = \frac{g}{t_F} \sum_{j < k}^N \sum_{n < m}^N t_{nm} (-1)^{\delta_{nj} + \delta_{nk} + \delta_{mj} + \delta_{mk}} \sigma_z^{(j)} \sigma_z^{(k)}. \quad (2.6)$$

Thus, the problem of finding the value of each time t_{nm} is a matrix-inversion problem,

$$g_\beta = t_\alpha M_{\alpha\beta} \frac{g}{t_F} \quad \rightarrow \quad t_\alpha = M_{\alpha\beta}^{-1} g_\beta \frac{t_F}{g}, \quad (2.7)$$

where α and β are introduced to vectorize each pair of indices (n, m) and (j, k) as

$$\alpha = N(n-1) - \frac{n(n+1)}{2} + m, \quad \beta = N(j-1) - \frac{j(j+1)}{2} + k, \quad (2.8)$$

and M is a sign matrix built up by the elements,

$$M_{\alpha\beta} = (-1)^{\delta_{nj} + \delta_{nk} + \delta_{mj} + \delta_{mk}}, \quad (2.9)$$

where the inverted relations for the indices n and j are

$$j = 1 + \left\lfloor \frac{\beta}{N} \right\rfloor, \quad (2.10)$$

$$k = \beta + \frac{1}{2} \left(1 + \left\lfloor \frac{\beta}{N} \right\rfloor \right) \left(2 + \left\lfloor \frac{\beta}{N} \right\rfloor \right) - N \left\lfloor \frac{\beta}{N} \right\rfloor, \quad (2.11)$$

$$n = 1 + \left\lfloor \frac{\alpha}{N} \right\rfloor, \quad (2.12)$$

$$m = \alpha + \frac{1}{2} \left(1 + \left\lfloor \frac{\alpha}{N} \right\rfloor \right) \left(2 + \left\lfloor \frac{\alpha}{N} \right\rfloor \right) - N \left\lfloor \frac{\alpha}{N} \right\rfloor. \quad (2.13)$$

The matrix $\mathbf{M}_{\alpha\beta}$ is a non-singular matrix $\forall N \in \mathbb{Z} - \{4\}$ since it has three degenerate eigenvalues, namely, $\lambda_1 = N(N-9)/2+8$, $\lambda_2 = 2(4-N)$ and $\lambda_3 = 4$ with degeneracies 1, $N-1$, and $N(N-1)/2-N$, respectively. The corner case $N=4$ requires the use of a slightly modified set of SQRs, e.g. single σ_x -rotations per site is sufficient for a NN Hamiltonian. The total unitary evolution is $U_{ZZ}(t_F) = e^{it_F \sum_{\beta} H_{ZZ}^{\beta}}$, where

$$H_{ZZ}^{\beta} = g_{\beta} \sigma_z^{j(\beta)} \sigma_z^{k(\beta)}, \quad (2.14)$$

and $g_{\beta} = t_{\alpha} \mathbf{M}_{\alpha\beta}(g/t_F)$.

By solving the linear problem, we determine the durations $t_{\alpha} = \mathbf{M}_{\alpha\beta}^{-1} g_{\beta}(t_F/g)$ for each analog block, interspersed with pairs of SQRs, as depicted in Figure 2.1. The uniqueness of the solution is guaranteed by the Rouché-Frobenius theorem, as the matrix $\mathbf{M}_{\alpha\beta}$ is invertible. It is noteworthy that some of the t_{α} may be negative, indicating the need for inverted coupling signs during the corresponding analog evolutions. However, this issue can be easily resolved by evolving with modified durations $\tilde{t}_{\alpha} = t_{\alpha} + |t_{\min}|$. In Appendix A.2 we provide further information.

In conclusion, this section presents an optimized DAQC protocol for the construction of an arbitrary inhomogeneous ATA two-body Ising Hamiltonian utilizing a homogeneous ATA two-body Ising Hamiltonian and x -rotations as resources. The protocol, which is quadratic in the total number of qubits, is optimal for a generic Hamiltonian as it utilizes the same number of resources as degrees of freedom are introduced in the Hamiltonian. In the case of NN Hamiltonian, the protocol is even simpler and requires only $N-1$ SQRs, with one SQR per site. These protocols demonstrate the universality of the Ising model, as it can be utilized to generate a ZZ gate between any two arbitrary qubits.

XZ model

As the Ising model is a universal Hamiltonian within the DAQC paradigm, it can be used to simulate the evolution of any other Hamiltonian. To accomplish this in a systematic manner, we will utilize a Trotter decomposition in conjunction with the previous algorithm. As an example, we will demonstrate the construction of an inhomogeneous general two-body XZ Hamiltonian, as depicted in Figure 2.2.

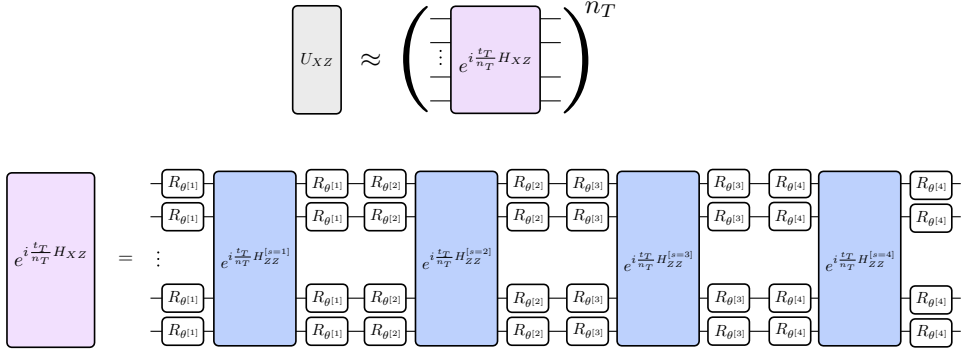


Figure 2.2: **Algorithm to simulate the inhomogeneous XZ model from an Ising model.** The combination of four different general Ising evolutions and rotations handles the minimum degrees of freedom for each term $g_{jk}^{\mu\nu}$. R_{θ} rotations can be combined with inner $\sigma_x^n \sigma_x^m$ rotations required to implement the inhomogeneous ZZ Ising model into unique blocks of SQRs.

We want to simulate the unitary evolution

$$U_{XZ} = e^{it_F H_{XZ}}, \quad \text{with } H_{XZ} = \sum_{j < k} \sum_{\mu, \nu = \{x, z\}} g_{jk}^{\mu\nu} \sigma_{\mu}^{(j)} \sigma_{\nu}^{(k)}. \quad (2.15)$$

To do so, we start performing a Trotter decomposition $U_{XZ} \approx (e^{i \frac{t_F}{n_T} H_{XZ}})^{n_T}$, with n_T the number of Trotter steps,

$$\begin{aligned} H_{XZ} &= \sum_{j < k} \sum_{\mu, \nu} g_{jk}^{\mu\nu} \sigma_{\mu}^{(j)} \sigma_{\nu}^{(k)} = \sum_{j < k} \sum_{\mu, \nu, s} g_{jk}^{[s]} \alpha_j^{(\mu, s)} \alpha_k^{(\nu, s)} \sigma_{\mu}^{(j)} \sigma_{\nu}^{(k)} \\ &= \sum_{j < k} \sum_{s=1}^4 g_{jk}^{[s]} \left(\cos(\theta_j^{[s]}) \sigma_z^{(j)} + \sin(\theta_j^{[s]}) \sigma_x^{(j)} \right) \times \\ &\quad \left(\cos(\theta_k^{[s]}) \sigma_z^{(k)} + \sin(\theta_k^{[s]}) \sigma_x^{(k)} \right), \end{aligned} \quad (2.16)$$

with s running from $\{1, \dots, 4\}$, which are the number of combinations of the two types of couplings (for instance, for the XYZ Hamiltonian, it would run from 1 to 9). The parameters of the above formula are defined as

$$\alpha_j^{(x, s)} = \sin(\theta_j^{[s]}) \quad \text{and} \quad \alpha_j^{(z, s)} = \cos(\theta_j^{[s]}). \quad (2.17)$$

The pairs of operators are decomposed with their coefficients in homogeneous $\sigma_z^{(j)} \sigma_z^{(k)}$ operators with local rotations

$$R_{\theta_j^{[s]}} = \left(\cos(\theta_j^{[s]}/2) \sigma_z^{(j)} + \sin(\theta_j^{[s]}/2) \sigma_x^{(j)} \right), \quad (2.18)$$

thus

$$R_{\theta_j^{[s]}} \sigma_z^{(j)} R_{\theta_j^{[s]}} = \left(\cos(\theta_j^{[s]}) \sigma_z^{(j)} + \sin(\theta_j^{[s]}) \sigma_x^{(j)} \right), \quad (2.19)$$

for all pairs of qubits. This rotation is produced by the Hamiltonian

$$H_{\theta_j^{[s]}} = (\pi/2) \left(-\mathbb{1} + \cos(\theta_j^{[s]}) \sigma_z^{(j)} + \sin(\theta_j^{[s]}) \sigma_x^{(j)} \right). \quad (2.20)$$

The SQRs have to be performed in all qubits $R_{\theta^{[s]}} = \otimes_w^N R_{\theta_w^{[s]}}$. The total Hamiltonian is reconstructed as

$$H_{XZ} = \sum_{s=1}^4 R_{\theta^{[s]}} \left(g_{jk}^{[s]} \sigma_z^{(j)} \sigma_z^{(k)} \right) R_{\theta^{[s]}}^\dagger. \quad (2.21)$$

This equality is only valid in the Hamiltonian and not in the total unitary evolution. This means that, in each Trotter time step $t_T = (t_F/n_T)$, we must evolve according to $H_{ZZ}^{[s]} = g_{jk}^{[s]} \sigma_z^{(j)} \sigma_z^{(k)}$ between a pair of sets of SQRs $R_{\theta^{[s]}}$

$$U_{XZ} \approx \left(\prod_{s=1}^4 R_{\theta^{[s]}} e^{i \frac{t_T}{n_T} (H_{ZZ})^{[s]}} R_{\theta^{[s]}} \right)^{n_T}. \quad (2.22)$$

The objective then, is to find a set of phases $\theta_w^{[s]}$ that satisfy the equation $g_{jk}^{\mu\nu} = g_{jk}^{[s]} \alpha_j^{(\mu,s)} \alpha_k^{(\nu,s)}$. A suitable set of phases is given by $\theta_w^{[s]} = \frac{s\pi w}{2(w+1)}$, with a polynomially-scaled distance between nearest-neighbor qubit phases of $d(s, w) = \frac{s\pi}{2(w^2+3w+2)}$. Further details can be found in Appendix A.3.

Thus, we have presented a protocol for constructing an arbitrary two-body Hamiltonian using the freedom in the phases of SQRs and Trotterization. This protocol requires $4N(N-1)$ ($4(N-1)$) analog blocks per Trotter step for the ATA (NN) XZ model, making it optimal in terms of the number of free coefficients. Additionally, it can be shown that extending the protocol to the general two-body XYZ model would require even more angles per rotation, summing up to $9N(N-1)$ ($9(N-1)$) blocks per Trotter step for the ATA (NN) model.

M-body Hamiltonian

By utilizing similar methods, one can systematically build the evolution of any completely general Hamiltonian with interactions of up to M bodies. The procedure can be visualized in the diagram shown in Fig. 2.3. For the sake of clarity, we present a clear outline of the steps necessary to simulate an arbitrary 4-body nearest-neighbor Hamiltonian and provide a detailed explanation on how to extend the protocol to Hamiltonians with up to M bodies in the appendix (see Appendix A.4 for further details). The 4-body Hamiltonian would be defined as

$$\begin{aligned} H_{4b} = & \sum_{j\chi\eta} g_{(2,j)}^{\chi\eta} \sigma_\chi^{(j)} \sigma_\eta^{(j+1)} + \sum_{j\chi\eta\gamma} g_{(3,j)}^{\chi\eta\gamma} \sigma_\chi^{(j)} \sigma_\eta^{(j+1)} \sigma_\gamma^{(j+2)} \\ & + \sum_{j\chi\eta\gamma\rho} g_{(4,j)}^{\chi\eta\gamma\rho} \sigma_\chi^{(j)} \sigma_\eta^{(j+1)} \sigma_\gamma^{(j+2)} \sigma_\rho^{(j+3)}, \end{aligned} \quad (2.23)$$

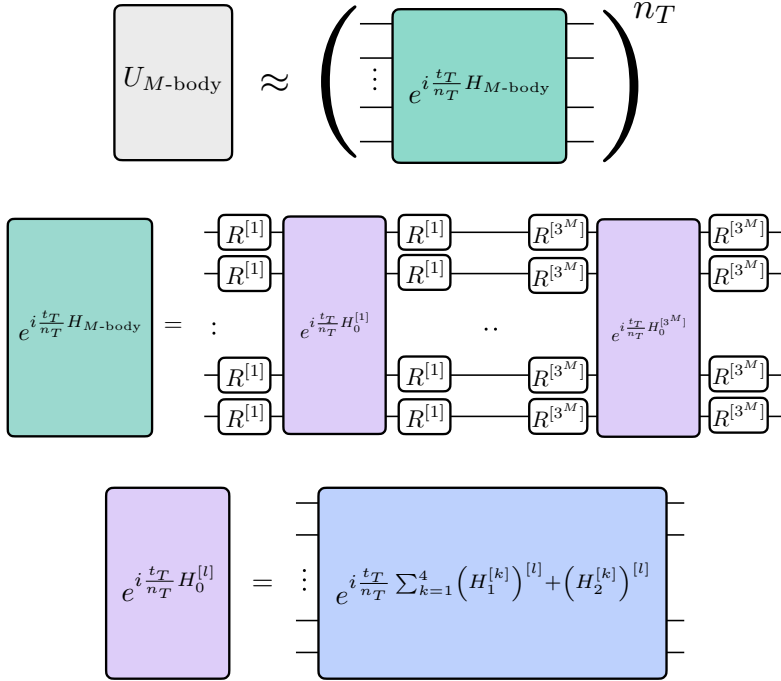


Figure 2.3: **Algorithm to implement the M -body evolution.** The time-evolution of $H_0^{[l]} = \sum_{k=1}^4 (H_1^{[k]})^{[l]} + (H_2^{[k]})^{[l]}$ is sandwiched by the generalized rotations $R^{[l]} = \otimes_j^N r_j^{[l]} \sigma_x^{(j)} + s_j^{[l]} \sigma_y^{(j)} + t_j^{[l]} \sigma_z^{(j)}$. The Hamiltonians $H_1^{[k]}$ and $H_2^{[k]}$ are defined in Eqs. (2.24) and (2.25). For an example with $M = 4$, we refer the reader to Appendix A.4.

where the indices run over $\{\chi, \eta, \gamma, \rho\} \in \{x, y, z\}$ and $j \in \{1, \dots, N\}$. First, an inhomogeneous ZZ-Ising Hamiltonian is sandwiched between rotated XX-Ising evolutions and their conjugate transposes,

$$H_1^{[k]} = e^{-iO_{XX}^{1[k]}} H_{ZZ}^{1[k]} e^{iO_{XX}^{1[k]}}, \quad (2.24)$$

$$H_2^{[k]} = e^{-iO_{XX}^{2[k]}} H_{ZZ}^{2[k]} e^{iO_{XX}^{2[k]}}, \quad (2.25)$$

with

$$O_{XX}^{1[k]} = \Phi_1^{[k]} \sigma_x^{(1)} \sigma_x^{(2)} + \Phi_3^{[k]} \sigma_x^{(3)} \sigma_x^{(4)} + \Phi_5^{[k]} \sigma_x^{(5)} \sigma_x^{(6)} + \dots \quad (2.26)$$

$$O_{XX}^{2[k]} = \Phi_2^{[k]} \sigma_x^{(2)} \sigma_x^{(3)} + \Phi_4^{[k]} \sigma_x^{(4)} \sigma_x^{(5)} + \Phi_6^{[k]} \sigma_x^{(6)} \sigma_x^{(7)} + \dots \quad (2.27)$$

built from evolutions of ZZ-Ising models rotated with the single qubit gates R in all qubits, defined as

$$R = \otimes_j (\cos(\pi/4) \sigma_z^j + \sin(\pi/4) \sigma_x^j). \quad (2.28)$$

It is important to note that the operators $O_{XX}^{1[k]}$ and $O_{XX}^{2[k]}$ contain interactions that are separated by an interaction length $L = M/2 = 2$. For example, $O_{XX}^{1[k]}$ includes a term $\sigma_x^{(1)}\sigma_x^{(2)}$, but not $\sigma_x^{(2)}\sigma_x^{(3)}$. To simulate a three-body (five-body) Hamiltonian, a different decomposition with 3 (5) translationally invariant sets of blocks would be required.

In other words, $H_1^{[k]}$ and $H_2^{[k]}$ together form a complete representation of the arbitrary Hamiltonian with up to four-body interactions. To further extend this idea to simulate Hamiltonians with M -body interactions, it is necessary to decompose the Hamiltonian into $M/2$ translationally invariant sets of blocks, similar to the procedure described for the 4-body Hamiltonian. The detailed explanation on how to achieve this generalization can be found in the appendix section A.4.

The coefficients of the $h_{ij\dots}$ operators are interrelated. To simulate an arbitrary four-body nearest-neighbor Hamiltonian, it is sufficient to sum four copies each of the blocks $H_0 = \sum_{k=1}^4 H_1^{[k]} + H_2^{[k]}$, in order to separate the parameters and generate at least one term that operates on each support. To obtain all XYZ operators, it is necessary to concatenate $3^M = 81$ H_0 blocks, interleaved by SQRs described by

$$R^{[l]} = \otimes_j^N (r_j^{[l]}\sigma_x^{(j)} + s_j^{[l]}\sigma_y^{(j)} + t_j^{[l]}\sigma_z^{(j)}). \quad (2.29)$$

In other words,

$$H_{4b} = \sum_l R^{[l]} H_0^{[l]} R^{[l]}, \quad (2.30)$$

with $(r_j^{[l]}, s_j^{[l]}, t_j^{[l]})$ fulfilling the constraint $|r_j^{[l]}|^2 + |s_j^{[l]}|^2 + |t_j^{[l]}|^2 = 1$. It is important to note that the above construction is applied to a single Trotter step, meaning that it must be repeated n_T times to approximate the evolution of the Hamiltonian, which can be expressed as $U_{4b} = e^{-iH_{4b}t} \approx (e^{-iH_{4b}t/n_T})^{n_T}$. The total number of analog blocks, or engineered time slices, required for the simulation of an arbitrary Hamiltonian with up to M -body interactions and N qubits is given by $a(M)N + b(M)$ with,

$$a(M) = \frac{9}{4} (3^{M-1} - 3), \quad (2.31)$$

$$b(M) = \frac{3^{M-1}}{2} \left(\frac{3}{2} - M \right). \quad (2.32)$$

Then, it can be stated that the computational cost of simulating NN Hamiltonians grows linearly with the number of qubits and exponentially with the number of bodies. For a specific example of a four-body system, the total number of required analog blocks is $117N - 306$. For a more in-depth understanding of the simulation algorithm for general M -body interactions, readers can refer to the appendix section (Appendix A.4).

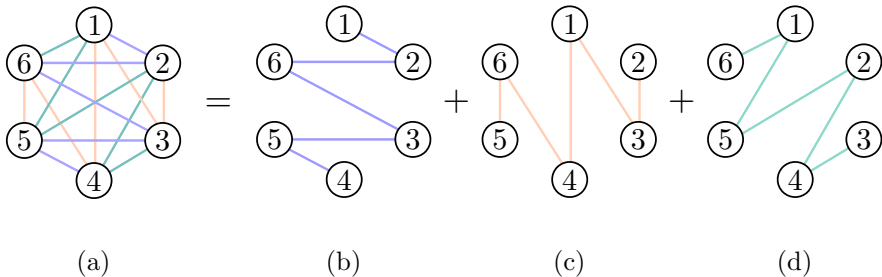


Figure 2.4: **Decomposition of a 6-qubit system with ATA connectivity into 3 NN 6-qubit system.** (a) Complete graph that one wants to simulate. (b - d) Hamiltonian paths that are used as the resource. Figure (b) is built by starting in the first node, going forward to the next node, then 2 nodes backward, 3 forward... The other two Hamiltonian paths are obtained rotating the first one.

2.2.1 Enhancing the connectivity of quantum hardware using DAQC

Throughout the entire section, we are assuming that the quantum hardware that we are employing presents an ATA connectivity. But it does not necessarily has to be like this, in fact, a realistic quantum processor is expected to present lower connectivity. This is given by the fact that ATA connections are extremely demanding in the amount of wiring among qubits.

In Ref. [72], the authors design an algorithm that simulated an arbitrary ATA Ising Hamiltonian employing as a resource a given inhomogeneous NN Ising model and SQRs, in an optimal manner. They do so using $\mathcal{O}(5N^2)$ analog block, where N stands for the number of qubits of the system. The algorithm is based in two main ideas:

1. The Ising Hamiltonian for N qubits can be represented as a weighted graph with N vertices, where the weight of the edge connecting vertex i to vertex j is the coupling constant between qubits i and j , denoted as g_{ij} . If two qubits are not connected, the vertices i and j representing them are also not connected, resulting in a null coupling constant ($g_{ij} = 0$).
2. In this representation, an ATA Ising Hamiltonian for N qubits can be expressed as a complete graph K_N , with edges between every possible vertex without repetition. On the other hand, the NN Ising Hamiltonian is represented as a Hamiltonian path that visits all possible vertices only once.

Figure 2.4 shows an example of an ATA 6-qubit system represented as a complete K_6 graph and its path decomposition into 3 NN 6-qubit systems.

The algorithm first splits the complete graph, representing the ATA Hamiltonian, into a set of Hamiltonian paths that can be efficiently decomposed into NN resources. Next, the algorithm obtains each of the Hamiltonian path connections using a SWAP-like gate U , which changes the connections between the qubits of the system. The

SWAP gate is defined as

$$U_{\text{SWAP}} = e^{i\frac{\pi}{4}(\sigma_x^i\sigma_x^j + \sigma_y^i\sigma_y^j + \sigma_z^i\sigma_z^j)}, \quad (2.33)$$

where the superscripts refer to the qubit on which the gate acts.

Partitioning a complete graph into a set of paths, allow to simulate an ATA Hamiltonian by using only NN-connected resources. This result extends the DAQC paradigm to a wide range of quantum platforms that require low connectivity. For a more detailed explanation of the algorithm, we refer the reader to the main article.

2.3 Stepwise and banged DAQC

The method we have described to simulate different families of Hamiltonians using the DAQC paradigm is the sDAQC. Under ideal circumstances, i. e. without taking into account noise sources, would lead to the same result as the DQC method. But this noiseless situation is just an illusion. The reality is that turning on and off multi-qubit quantum gates introduce an important source of errors in realistic quantum algorithms. In the interest of reducing its effect, we introduce the concept of bDAQC as a different way to perform quantum algorithms.

The term ‘‘bang bang’’ is a well-established concept in classical control theory, as seen in references [73, 74]. It was later introduced into the field of quantum physics to serve as a means of dynamically decoupling controlled quantum systems from their environment [75, 76]. In this work, we use the term ‘‘banged’’ to describe the situation in which the analog evolution is not interrupted while fast SQRs pulses are being applied. Compared to sDAQC, the bDAQC introduces an additional digital error, which could be of the third order in the duration of the pulse, at best. However, we argue that this error is in competition with the experimental error produced by the switch of multi-qubit gates and may, in certain cases, prove to be less harmful to the overall algorithm.

Stepwise DAQC

In a sDAQC scenario, a total unitary evolution is constructed by alternating an evolution under a fixed entangling Hamiltonian H_{Int} with sequences of SQRs

$$U_T = \dots U_{R_2} e^{-iH_{\text{Int}}t_2} U_{R_1} e^{-iH_{\text{Int}}t_1} U_{R_0}, \quad (2.34)$$

with U_{R_n} a general digital rotation operator acting on any subset of the Hilbert space –in this case, two qubits. In Fig. 2.4(a), we show a schematic representation of the procedure. According to the sudden approximation, we can implement such

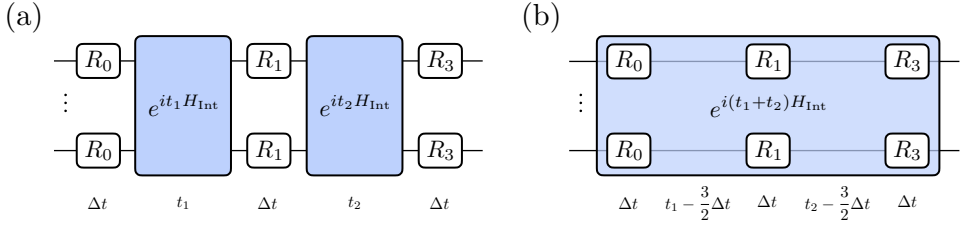


Figure 2.5: **Schematic representation of the sDAQC and the bDAQC protocol.** (a) sDAQC: digital evolutions with Hamiltonian H_{R_k} , and analog blocks ones evolving with Hamiltonians H_I are well separated in time under the sudden approximation. (b) bDAQC: the adiabatic evolution H_I is on during the whole time and fast rotations are added ($H_{R_k} + H_I$). The sudden approximation is employed.

evolution $U(t) \approx e^{-it_F H(t)}$ with a time-dependent Hamiltonian,

$$H(t) = \sum_{n=0} H_{R_n} \Pi_{\Delta t}(t - [T_n + n\Delta t]) + H_I \Pi_{T_{n+1}}(t - [T_n + (n+1)\Delta t]), \quad (2.35)$$

where H_{R_n} gives rise to the digital unitary evolution $U_{R_n} = e^{-iH_{R_n}\Delta t}$. We have defined $T_n = \sum_{r=0}^n t_r$, assuming $t_0 = 0$, and the rectangular window function

$$\Pi_T(t - t_s) = \theta(t - t_s) - \theta(t - (t_s + T)), \quad (2.36)$$

with $\theta(t) = 1$ for $t \geq 0$ and 0 otherwise.

Banged DAQC

The bDAQC approach involves continuously evolving the system with the entangling Hamiltonian H_I , while applying short, intense pulses to implement the SQRs H_{R_n} . In contrast, the stepwise protocol involves switching the Hamiltonians H_I and H_{R_n} on and off, an unrealistic assumption due to the finite speed of operation in physical systems.

By utilizing the symmetrized exponential decomposition for all the SQRs blocks, the ideal evolution of the system described in Eq. (2.34) can be accomplished without having to turn off the entangling Hamiltonian

$$H(t) = H_I + \sum_n^{N-1} H_{R_n} \Pi_{\Delta t}(t - [T_n - \Delta t/2]) + H_{R_N} \Pi_{\Delta t}(t - [T_N - \Delta t]), \quad (2.37)$$

where $T_n = \sum_{r=0}^n t_r$. In Fig. 2.4(b), we show an schematic representation of the bDAQC approach.

2.3.1 Error estimation

The error-per-step introduced can be estimated using the Schatten norm by calculating the difference between the evolution of the system using the sDAQC protocol

and the bDAQC. The error would then be defined as

$$\begin{aligned}
 e_n &= \left\| 1 - e^{-iH_I \Delta t/2} e^{-iH_{R_n} \Delta t} e^{-iH_I \Delta t/2} e^{i(H_I + H_{R_n}) \Delta t} \right\| \\
 &= \frac{(\Delta t)^3}{4} \left\| [[H_I, H_{R_n}], H_I + 2H_{R_n}] \right\| + O((\Delta t)^4), \tag{2.38}
 \end{aligned}$$

where we have made use of the Zassenhaus formula. The first and last blocks at the boundaries result in second-order errors, $e_{0,N}$, which are proportional to $(\Delta t)^2$ due to the lack of symmetry in their evolution. The overall error in the banded protocol is calculated as the sum of these errors, $E = \sum_n e_n = A e_n$, where the proportionality constant A is proportional to either $O(N)$ or $O(N^2)$, depending on whether the coupling configuration is NN or ATA, respectively. This is because the total number of analog blocks increases either linearly or quadratically with the number of qubits, N .

Naturally, the bDAQC does not generate the same result as the sDAQC or the DQC method, since there is an intrinsic error on the bDAQC which does not depend on either the experimental conditions or the noise sources. This error is due to the superposition between the Hamiltonians of the SQRs and the analog Hamiltonian. However, if SQRs are performed in a time Δt much smaller than the intrinsic time scale of the analog block, the error will be smaller than the one coming from switching on and off the analog Hamiltonian. Indeed, we have just shown that the additional error per SQR introduced by not turning off the evolution of the Hamiltonian is on the order $\mathcal{O}[(\Delta t)^3]$. The reason why we aim at using the bDAQC protocol despite its intrinsic error is that, it accumulates less experimental error. Experimentally, switching on and off the Hamiltonian is not an exact step function, it takes some time to stabilize. Quantum control tries to suppress these errors, but it turns cumbersome when the system scales up and cannot be solved in a classical computer. If we keep the analog block on during the evolution, we will avoid these errors. This will be of great importance when we explore a more realistic implementation of the DAQC protocol in the following chapters.

In this section, we presented systematic techniques to simulate different families of Hamiltonians using the DAQC paradigm. At some point, these Hamiltonians would represent certain quantum algorithms, such as the quantum Fourier transform or the Harrow-Hassidim-Lloyd. What we want to do next, is to compare the performance of these quantum algorithms when we implement them in a real quantum computer using either the DAQC or the DQC paradigm. For that, we first need to model the noisy environment that nowadays quantum computers represent. Afterward, we will be able to compare the results of each method in a realistic simulation.

2.4 Performance of the DAQC compared to other paradigms

In this section, we describe the noise model that we use to compare the performance of a quantum algorithm when it is implemented in a noisy quantum computer using either the DQC or the DAQC model. We begin by briefly describing the formalism that we will use to model the effect of the main noise sources of a NISQ device. Afterward, we will show how, using the aforementioned formalism, it is possible to simulate the implementation of a given quantum algorithm with experimental errors.

2.4.1 Theoretical description of the effect of noise sources in quantum systems.

Quantum operation formalism and Operator-sum representation

Quantum computers are open quantum systems, meaning that they cannot be isolated from the environment, and thus, they are susceptible to the noise caused by the control and the interaction with the environment. We can say that, in general, the noise sources depend on the experimental setup used to run the quantum experiments. As the field develops, these sources tend to be less significant, but in the context of the NISQ era, we cannot obviate the experimental constraints.

We will employ the *quantum operation formalism* as a general tool for describing the evolution of a quantum system in different scenarios. A deeper description of this formalism can be found in chapter 8 of Ref. [77], but here we will describe the main aspect of it.

The quantum states are well described by the *density operator* ρ , thus, they transform as

$$\rho' = \mathcal{E}(\rho), \quad (2.39)$$

where the map \mathcal{E} is a quantum operation. By using this formalism, we can describe any unitary transformation U as $\mathcal{E}(\rho) = U\rho U^\dagger$, and, similarly, the measurement process as $\varepsilon_m(\rho) = M_m\rho M_m^\dagger$. Thus, the quantum operation tells us about the dynamic change of a state as a result of a physical process, where the initial state is ρ and the final state is $\mathcal{E}(\rho)$.

Unitary transformations can describe the dynamics of a closed quantum system, where we have an input state ρ , and an output one $U\rho U$. We can see the unitary transform U as a black box whose internal details do not concern us. But quantum computers do not behave as closed quantum systems but rather like open quantum systems, whose dynamics can be described as an interaction between a principal system, which is the one that we want to study, and an environment. Together, they form a closed quantum system. This way, when we send the initial state ρ into a box that is coupled to the environment, the final state $\mathcal{E}(\rho)$ may not be related to the initial state by a unitary transformation. After the unitary transformation U , the system no longer interacts with the environment and we can perform the partial

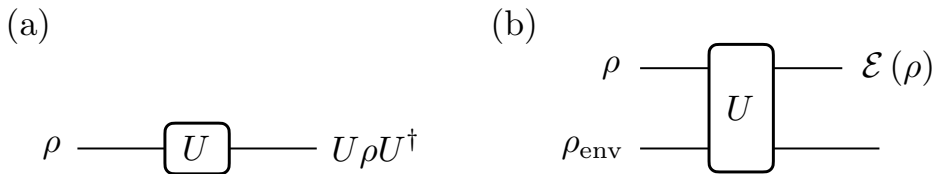


Figure 2.6: **Schematic representation of a unitary operation U in two different quantum systems.** (a) Closed quantum system. (b) Open quantum system, composed of the principal system and the environment.

trace over the environment to obtain the reduced state of the system, i.e.

$$\mathcal{E}(\rho) = \text{Tr}_{\text{env}} [U(\rho \otimes \rho_{\text{env}})U^\dagger], \quad (2.40)$$

assuming that the system-environment input state can be described as a product state, $\rho \otimes \rho_{\text{env}}$. Fig. 2.6 shows a schematic picture of a unitary transformation taking place on a closed (Fig. 2.6(a)) and an open (Fig. 2.6(b)) quantum system. A useful manner to represent quantum operations is the *operator-sum* representation, which we will introduce below.

Let us suppose that we have an open quantum system that includes our system of interest \mathcal{S} and the environment \mathcal{E} , with Hilbert spaces $\mathcal{H}_{\mathcal{S}}$ and $\mathcal{H}_{\mathcal{E}}$ respectively, and being $|e_k\rangle$ an orthonormal basis for the finite-dimensional Hilbert space corresponding to the environment. Let us assume that the initial state of the environment is $\rho_{\text{env}} = |e_0\rangle\langle e_0|$. Then, for a unitary transformation U , the quantum operation can be written as

$$\mathcal{E}(\rho) = \sum_k |e_k\rangle\langle e_0| U(\rho \otimes |e_0\rangle\langle e_0|) U^\dagger |e_k\rangle\langle e_0| \quad (2.41)$$

$$= \sum_k E_k \rho E_k^\dagger, \quad (2.42)$$

where $E_k \equiv \langle e_k|U|e_0\rangle$ is an operator which acts on the state space of the principal system. The operators E_k are the *operation elements* of the quantum operation \mathcal{E} , also known as *Kraus operators* [78] and they play a key role in the description of the effect of noise sources in quantum systems. We will use these operators to characterize the quantum channel for each noise source. In the case where the initial state of the environment is not a pure state, we can always introduce an extra system to purify the environment. This extra system can be used as an intermediate step in calculations since it does not interfere with the dynamics of the principal system, and thus there is no loss of generality in assuming that the environment starts in a pure state.

The Kraus operators E_k satisfy the completeness relation

$$\sum_k E_k^\dagger E_k = \mathbb{1}. \quad (2.43)$$

This arises from the condition that the trace of \mathcal{E} must be preserved and hence equal to one, i.e.

$$\begin{aligned} 1 &= \text{Tr}(\mathcal{E}(\rho)) \\ &= \text{Tr}\left(\sum_k E_k \rho E_k^\dagger\right) \\ &= \text{Tr}\left(\sum_k E_k E_k^\dagger \rho\right), \end{aligned}$$

and, since this condition has to be fulfilled for all ρ , then we must have $\sum_k E_k^\dagger E_k = \mathbb{1}$.

The power of the operator-sum representation is that it allows us to characterize the principal system without having to consider the properties of the environment. We just need the Kraus operators E_k that act on the principal system. In the following, we will specify the form of these operators for the different incoherent noise sources that can arise during the implementation of a quantum algorithm into a NISQ device.

Incoherent noise sources: noise channels

Noise channels are non-unitary quantum channels that reproduce the effect of noise in the state of the system of interest, in our case the qubits comprising a quantum computer. Let us first focus on single-qubit quantum channels, which are those that only act on one qubit. More concretely, we study the noise channels that account for the three most common noise sources in superconducting quantum processors. These are bit-flip, decoherence, and measurement error.

1. **Bit-flip and measurement channel.** It is a model for errors in which the state of the qubit is randomly switched with a probability p and it is also present in classical computers. The energy to cause the flip of the qubit can be accidentally provided by pulses used to control the qubits or by thermal fluctuations. The Kraus operators of the bit-flip channel are

$$E_0 = \sqrt{1-p} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = \sqrt{1-p} \mathbb{I}, \quad (2.44)$$

$$E_1 = \sqrt{p} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = \sqrt{p} X, \quad (2.45)$$

where p is the bit-flip probability, \mathbb{I} is the 2×2 identity matrix and X is the σ_x Pauli matrix. When the measurement of the system is performed, the qubits might be affected by a bit-flip error. Based on this idea, we will introduce the measurement error as a bit-flip error preceding a perfect measurement operation.

2. **Amplitude-damping channel.** Decoherence is represented by means of the generalized amplitude-damping channel. This quantum channel accounts for

the de-excitation of a two-level system, i.e., the loss of amplitude of the excited state $|1\rangle$ that will decay into the ground state $|0\rangle$ with probability p . It can describe different physical phenomena related to energy dissipation such as the spontaneous emission of a photon and the scattering and attenuation of the state of a photon in a cavity, among others [77]. The Kraus operators of the generalized amplitude-damping channel are

$$E_0 = \sqrt{p} \begin{pmatrix} 1 & 0 \\ 0 & \sqrt{1-\gamma} \end{pmatrix}, \quad (2.46)$$

$$E_1 = \sqrt{p} \begin{pmatrix} 0 & \sqrt{\gamma} \\ 0 & 0 \end{pmatrix}, \quad (2.47)$$

$$E_2 = \sqrt{1-p} \begin{pmatrix} \sqrt{1-\gamma} & 0 \\ 0 & 1 \end{pmatrix}, \quad (2.48)$$

$$E_3 = \sqrt{1-p} \begin{pmatrix} 0 & 0 \\ \sqrt{\gamma} & 0 \end{pmatrix}, \quad (2.49)$$

where the stationary state of the environment is

$$\rho_{\text{env}} = \begin{pmatrix} p & 0 \\ 0 & 1-p \end{pmatrix}, \quad (2.50)$$

with p the thermal population of the ground state.

The damping process can be understood as a time-accumulative problem, in which the probability grows with time [79]. It can be shown that γ can be written as $\gamma = 1 - e^{-t/T_1}$, where t is the time and T_1 is the thermal relaxation time, and it describes processes due to the coupling of the qubit to its neighbors, which is a large system in thermal equilibrium at a temperature much higher than the one of the qubit [77].

Coherent noise sources: quantum trajectories

For the noise model, we also took into account coherent control-related experimental errors. In this case, instead of the quantum-channel approach, we employed quantum trajectories. These errors include those that might appear when we apply a SQG in both the DQC and the DAQC paradigm, the two-qubit gate (TQG) exclusive for the DQC, and the analog block noise exclusive for the DAQC.

1. **Single-qubit gate error.** For this, we have included a magnetic-field noise ΔB added to the phases of the SQGs. This parameter ΔB is taken from a uniform probability distribution $\mathcal{U}(-r_D\Delta t/2, r_D\Delta t/2)$, where $\mathcal{U}(a, b)$ stands for a uniform noise distribution with range boundaries (a, b) . The deviation ratio is given by r_D . This can be modeled as

$$e^{iH_{\text{SQG}}} \rightarrow e^{i\Delta B H_{\text{SQG}}}, \quad (2.51)$$

where H_{SQG} is the corresponding Hamiltonian for the applied SQG.

2. **Two-qubit gate control error.** As the CZ gate can be written in terms of SQG's and $\sigma_z\sigma_z$ interactions, the effects of noise in any TQG can be modeled using the magnetic field noise for SQG's and a Gaussian phase noise for the two-qubit $\sigma_z\sigma_z$ interaction. This Gaussian phase noise is represented by a random parameter $\varepsilon \in \mathcal{N}(0, \sigma_D)$ affecting the phase of the interaction, where $\mathcal{N}(\mu, \sigma)$ stands for a Gaussian distribution with mean μ and standard deviation σ . Its effect on a fixed $\pi/4$ phase interaction, which is usually the interaction of a digital quantum computer, is described by the change

$$e^{i\frac{\pi}{4}\sigma_z\sigma_z} \rightarrow e^{i\frac{\pi}{4}(1+\varepsilon)\sigma_z\sigma_z}. \quad (2.52)$$

3. **Analog block control error.** To model this error, we include a Gaussian coherent noise to the time that the analog blocks are applied. It produces a switch in the time from $t_\alpha \rightarrow t_\alpha + \delta$, where $\delta \in \mathcal{N}(0, r_b\Delta t)$, with r_b the deviation ratio of the time Δt required for a SQR. This can be implemented as

$$e^{it_\alpha H_{\text{int}}} \rightarrow e^{i(t_\alpha+\delta)H_{\text{int}}}, \quad (2.53)$$

where H_{int} is the natural interacting Hamiltonian of the quantum processor. The Gaussian coherent phase noise has a greater value for sDAQC than for bDAQC, as a result of the effect of switching on and off the interaction in sDAQC. The value of these parameters changes each time a gate is applied to the circuit, as the effects of the noise described can vary in each application.

2.4.2 Error mitigation in DAQC

As the NISQ era progresses, new methods for mitigating errors in quantum computing have emerged [80, 81, 82]. One approach is the use of quasi-probability methods on analog, DAQC, or DQC platforms [83, 84]. These techniques have been shown to effectively eliminate or suppress errors, such as Trotter error [85]. In order to demonstrate the true performance of DAQC, it is important to apply these error mitigation techniques. In chapter 3, we demonstrate this by incorporating these techniques in the implementation of the quantum Fourier transform on a superconductive circuit quantum computer. For more detailed information on the results, we refer the reader to that chapter.

In this chapter, we have described the DAQC paradigm and presented the main tools that will allow us to work within this framework. In the following, we will show the different quantum algorithms we have proposed that emanate from the DAQC method.

3 Digital-analog quantum Fourier transform and phase estimation

THE algorithms that we presented in chapter 2 allow us to describe relevant quantum algorithms in terms of the DAQC framework. In this chapter, we present a DA implementation of both the quantum Fourier transform (QFT) and the quantum phase estimation (QPE). We chose the QFT as the first quantum subroutine since it is a key ingredient for both more elaborated quantum subroutines, such as the QPE, and relevant quantum algorithms, such as Shor’s algorithm for factorization [7]. The content of this chapter include the work developed in [52] and [53].

In section 3.1, we introduce the QFT, describing it in both the DQC and the DAQC paradigms. We compare the performance of both paradigms in a NISQ device, for which we have included the noise sources listed in section 2.4 in our simulations. We also included an analysis of how error mitigation techniques would improve the DAQC implementation. Afterward, in section 3.2, we study the QPE algorithm in a similar manner as we did for the QFT. We first find its description using DAQC and then show its performance in a realistic noisy scenario. The DAQC description of these two quantum algorithms will allow us to face more complex algorithms, such as the HHL, in future chapters.

3.1 Quantum Fourier transform

3.1.1 Description of the QFT algorithm and its DAQC implementation

The discrete Fourier transform (DFT) is a mathematical tool that is widely used in signal processing, image processing, and physics. In classical computing, the DFT is used to transform a discrete signal from the time domain to the frequency domain, which can be useful for analyzing the frequency components of a signal. It is a computationally expensive task, and it can take a significant amount of time to perform on a classical computer.

In quantum computing, the DFT is performed by the QFT algorithm. The QFT is a quantum algorithm that performs the Fourier transform on a quantum state. It is a key building block in many quantum algorithms, such as Shor's algorithm for factoring integers and the QPE. The QFT can be implemented using a series of quantum gates and can be applied to a system of qubits. The main advantage of the QFT over classical algorithms is that it can be performed exponentially faster on a quantum computer.

The classical DFT takes a complex vector of length N , $(x_0, x_1, \dots, x_{N-1})$ and transforms it into another complex vector of the same length $(y_0, y_1, \dots, y_{N-1})$ whose k -th element is defined as

$$y_k \equiv \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} x_j e^{2\pi i j k / N}. \quad (3.1)$$

QFT, its quantum counterpart, is a linear operator \mathcal{F} with the following action on the basis states:

$$\mathcal{F}|j\rangle \equiv \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{2\pi i j k / N} |k\rangle, \quad (3.2)$$

where $N = 2^n$ and n is the number of qubits of the system. In Fig. 3.1 we show the implementation of the QFT using single- and two-qubit gates. The only single-qubit gates applied are Hadamard gates H , whose unitary matrix and Hamiltonian expressions are

$$H = e^{iH_H} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \quad H_H = \frac{\pi}{2} \left[\mathbb{1} - \frac{1}{\sqrt{2}}(Z + X) \right], \quad (3.3)$$

respectively. The entangling two-qubit gates of the circuit implementation are the controlled- R_k (cR_k) rotations with

$$R_k = \begin{pmatrix} 1 & 0 \\ 0 & e^{2\pi i / 2^k} \end{pmatrix}, \quad (3.4)$$

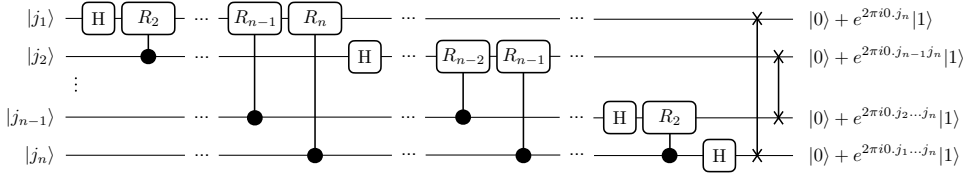


Figure 3.1: **Digital implementation of the QFT for an n -qubit system.** Each set of controlled operations cR_k is preceded by a Hadamard gate, H. The family of controlled rotations cR_k is defined as $cR_k = |0\rangle\langle 0| \otimes \mathbb{1} + |1\rangle\langle 1| \otimes R_k$, with $R_k = |0\rangle\langle 0| + e^{2\pi i/2^k} |1\rangle\langle 1|$. The final state is written using the binary representation for $|j\rangle$, where $j = j_1 j_2 \dots j_n = \sum_{l=1}^n j_l 2^{n-l}$, and $0.j_1 j_2 \dots j_m$, refers to the binary fraction $j_1/2 + j_2/4 + \dots + j_m/2^{m-l+1}$.

$$cR_k = |0\rangle\langle 0| \otimes \mathbb{1} + |1\rangle\langle 1| \otimes R_k = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & e^{2\pi i/2^k} \end{pmatrix}. \quad (3.5)$$

They appear in $(n - 1)$ different blocks of controlled rotations, all of them preceded by a Hadamard gate as shown in Fig. 3.1.

In order to apply the DAQC protocol to implement the QFT, we first have to describe the complete QFT as one unitary operation whose Hamiltonian we can simulate using the DAQC algorithms presented in section 2.1. To do that, we express the unitary matrices involved in the algorithm in terms of the resource Hamiltonian. Without loss of generality, we will assume that the interaction in the quantum processor can be described by an ATA two-body Ising Hamiltonian. Thus, a complete quantum subroutine can be defined as

$$U_{\text{QFT}} = \left[\prod_{m=1}^{n-1} U_{\text{SQG},m} U_{\text{TQG},m} \right] U_{H,m}, \quad (3.6)$$

where

$$U_{\text{SQG},m} = \exp \left[i \sum_{k=2}^{n-(m-1)} \theta_k \left(\mathbb{1}_{N \times N} - \sigma_z^{(k+m-1)} - \sigma_z^{(m)} \right) \right] \times \exp \left[\frac{i\pi}{2} \left(\mathbb{1} - \frac{\sigma_z^{(m)} + \sigma_x^{(m)}}{\sqrt{2}} \right) \right], \quad (3.7)$$

$$U_{\text{TQG}} = \exp \left(i \sum_{c < k}^n \alpha_{c,k,m} \sigma_z^{(c)} \otimes \sigma_z^{(k)} \right), \quad (3.8)$$

$$U_{H,m} = \exp \left(\frac{i\pi}{2} \left[\mathbb{1}^{(m)} - \frac{(\sigma_z^{(m)} + \sigma_x^{(m)})}{\sqrt{2}} \right] \right), \quad (3.9)$$

$$\theta_k = \frac{\pi}{2^{k+1}}, \quad \text{and} \quad \alpha_{c,k,m} = \delta_{c,m} \frac{\pi}{2^{k-m+2}}. \quad (3.10)$$

3.1 Quantum Fourier transform

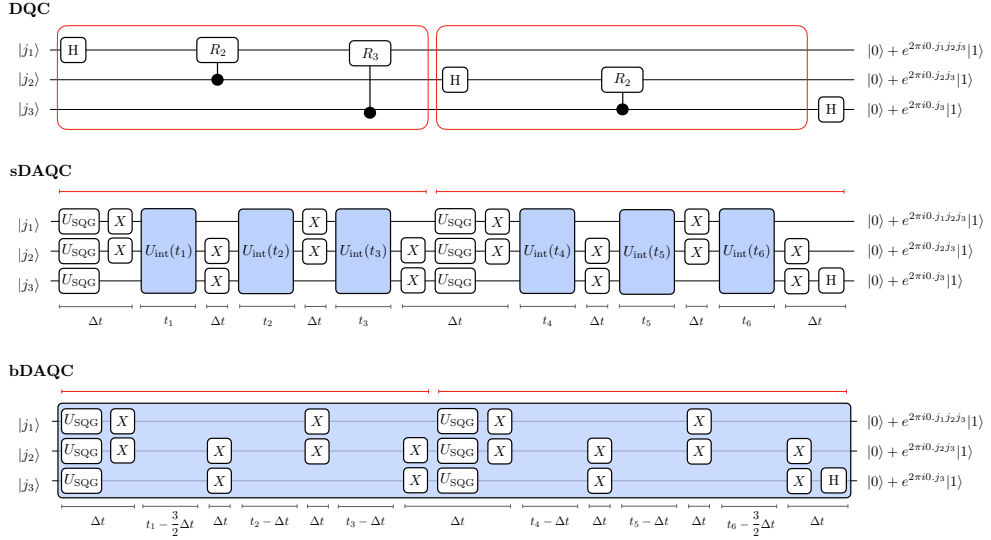


Figure 3.2: **Implementation of the QFT for a 3– qubit system using three different protocols: DQC, sDAQC and bDAQC.** The blue blocks $U_{int}(t_k)$ represent the analog blocks, and each of them is applied during different times t_k . The single-qubit rotations $R_x(\pi)$, which correspond to an X gate, act for a time Δt . We apply the DAQC protocol for each block of controlled rotations of the DQC implementation, which is detailed by the red line over each of those blocks. The sDAQC switches on and off the analog evolution before applying the single-qubit rotations X . In contrast, in the bDAQC protocol, the single-qubit rotations are performed on top of the analog evolution. The total analog interaction time in the bDAQC must equal the sum of each individual analog block in the sDAQC.

The superindices in parenthesis specify the qubit in which the unitary operation is performed and the operators σ_j refers to the Pauli matrices.

In Fig. 3.2, we depict the DQC implementation of the QFT using Eqs. (3.7)-(3.9). As one can see, each controlled-rotation block can be implemented by applying first a set of single-qubit gates, and then a set of two-qubit gates. This is why we decompose the complete unitary transformation into three different operations. The subindices SQG and TQG stand for single-qubit gates and two-qubit gates, respectively. The inhomogeneous ATA two-body Ising Hamiltonian we want to write in the DAQC framework [see Eq. (3.8)] represents a complete block of controlled rotations, and it is different for each block. This means that we need to apply the DAQC protocol $(n - 1)$ times, one time per controlled-rotation block.

Every two-qubit gate is applied following the ATA DQC protocol and using a fixed $\pi/4$ phase

$$e^{i\varphi_{jk}^{\mu\nu} \sigma_\mu^{(j)} \sigma_\nu^{(k)}} = e^{i(\pi/4)\sigma_y^{(j)}} e^{i(\pi/4)\sigma_\mu^{(j)} \sigma_\nu^{(k)}} e^{i\varphi_{jk}^{\mu\nu} \sigma_y^{(j)}} e^{-i(\pi/4)\sigma_\mu^{(j)} \sigma_\nu^{(k)}} e^{-i(\pi/4)\sigma_y^{(j)}}. \quad (3.11)$$

In our case, $\mu = \nu = Z$ and the phase $\varphi_{jk}^{\mu\nu}$ correspond to the coefficient $\alpha_{c,k,m}$, given

in Eq. (3.10),

$$\begin{aligned}
 e^{i\alpha_{c,k,m}\sigma_z^{(c)}\sigma_z^{(k)}} &= e^{i(\pi/4)\sigma_y^{(c)}} e^{i(\pi/4)\sigma_z^{(c)}\sigma_z^{(k)}} e^{i\alpha_{c,k,m}\sigma_y^{(c)}} R_x(\pi)^{(k)} \\
 &e^{i(\pi/4)\sigma_z^{(c)}\sigma_z^{(k)}} R_x(\pi)^{(k)} e^{-i(\pi/4)\sigma_y^{(c)}}, \tag{3.12}
 \end{aligned}$$

where the rotation $R_x(\theta)^{(k)}$ is applied to the k -th qubit and its defined as $R_x(\theta) = e^{-i\theta\sigma_x/2}$. We could also have decomposed the single- and two-qubit gates involved in the circuit into a base gate-set that forms a universal set of gates, such as $\{Rz(\theta), Rx(\theta), CNOT\}$. In this case, we decided to go this way so we could be as close as possible to the work of Parra et. al [51]. In the following works, we decided to go the other way, so we could include in our simulations the same kind of experimental errors that companies like IBM specify about their devices. These specifications were not available when we first presented our work on the QFT, so we took as reference the state-of-art resonators that were known at that time.

3.1.2 Realistic implementation with experimental errors

Now we will compare the performance of the QFT using two different paradigms, the DQC and the DAQC. To do so, we will compute the Fourier transform of a family of initial states $|\psi_0\rangle = \sin\beta|W_N\rangle + \cos\beta|\text{GHZ}_N\rangle$, with $\beta \in [0, \pi]$. We apply the QFT for a 3-, 5-, and 6-qubit system to gasp the effect of the noise sources as the number of qubits increases. As a figure of merit we have used the quantum fidelity,

$$F(\rho_{\text{ideal}}, \rho_{\text{noisy}}) = \left[\text{tr} \left(\sqrt{\sqrt{\rho_{\text{ideal}}}\rho_{\text{noisy}}\sqrt{\rho_{\text{ideal}}}} \right) \right]^2, \tag{3.13}$$

where ρ_{ideal} is the ideal state after the application of the QFT, while ρ_{noisy} is the resulting state when the noise model is implemented. The fidelity verifies $0 \leq F(\rho_{\text{ideal}}, \rho_{\text{noisy}}) \leq 1$, thus the larger the values the more similar are ρ_{ideal} and ρ_{noisy} , meaning that the noise sources has less effect on the performance of the experiment.

The noise model that we consider encloses the most relevant noise sources in superconducting quantum computers: bit-flip, decoherence, and measurement error, as well as control-related errors such as magnetic field fluctuations. However, the difference in the implementation of the entangling blocks in DQC and DAQC leads to a different experimental error associated with them, which is modeled as a Gaussian phase noise on the application time of the TQG's in DQC and the analog blocks in DAQC, as we discussed in section 2.4.1, chapter 2.

The values of the parameters used to simulate the different noise sources are chosen to model realistic NISQ superconducting devices. In single-qubit gates, we have introduced a magnetic-field noise ΔB_γ by adding to the Hamiltonian of the single-qubit gate a random variable taken from a uniform probability distribution centered in 1, i.e., $\mathcal{U}(1 - \text{SQGN}, 1 + \text{SQGN})$. We have chosen $\text{SQGN} = 0.0005$. For the two-qubit gates, we add a Gaussian phase noise $\epsilon \in \mathcal{N}(0, \text{TQGN})$, with variance $\text{TQGN} = 0.2000$, to the $\pi/4$ phases in the DQC protocol. Finally, to model the

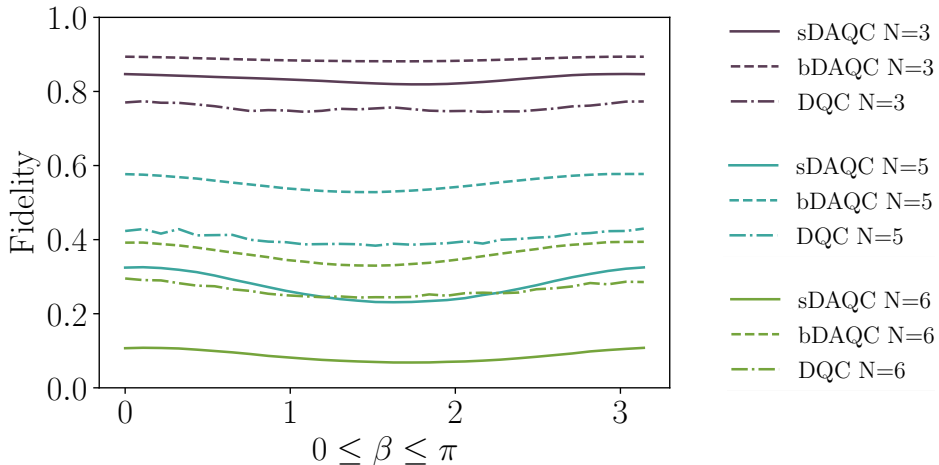


Figure 3.3: **Comparison of the fidelity of the QFT algorithm for 3-, 5-, and 6-qubit systems using the DQC, sDAQC, and bDAQC protocols.** The highest fidelity is obtained using the bDAQC protocol, which demonstrates better tolerance to decoherence and reduced bit-flip errors compared to the other protocols. The study was conducted using the state $|\psi_0\rangle = \sin \beta |W_N\rangle + \cos \beta |GHZ_N\rangle$.

experimental control error on the analog blocks, we include a Gaussian coherent noise to the time those blocks are applied, this is $t \rightarrow t + \delta$, where $\delta \in \mathcal{N}(0, \text{ABN})$. The value of the variance ABN depends on which DAQC protocol we are using. The value used on the sDAQC is double the value used for the bDAQC case. The values we have considered are $\text{ABN}_s = 0.0200$ for the sDAQC case and $\text{ABN}_b = 0.0100$ for the bDAQC case. Thus, each ideal gate transforms as

$$e^{i\theta_k Z} \rightarrow e^{i\theta_k \Delta B Z}, \quad (3.14)$$

$$e^{i(\pi/4)ZZ} \rightarrow e^{i(\pi/4)(1+\epsilon)ZZ}, \quad (3.15)$$

$$e^{it_\alpha H_{int}} \rightarrow e^{i(t_\alpha + \delta)H_{int}}. \quad (3.16)$$

On the other hand, for the incoherent noise sources introduced in section 2.4.1, we have a bit-flip error with probability $p_{b-f} = 0.005$ and a measurement error with probability $p_{meas} = 0.01$. Notice that measurement error is currently the technological limiting factor for quantum algorithms run in superconducting quantum processors. Finally, for the decoherence channel, the relaxation time T_1 is $50 \mu\text{s}$, with the thermal population of the ground state $p = 0.35$. The length of the SQGs is $\Delta t_{\text{SQG}} = 1/(100g_0)$, while for the two-qubit interaction, it is $\Delta t_{\text{TQG}} = (1 + \epsilon)100\Delta t_{\text{SQG}}\pi/4$ for TQG's, and for the analog blocks, it is the one obtained from the decomposition of the interaction.

In Fig. 3.3, we present the results of our realistic implementation of the QFT, which were obtained by averaging over 1000 repetitions to ensure a reliable statistical sample. The results indicate that the bDAQC protocol outperforms the others when

noise sources are taken into account. This is because the error caused by turning the interaction on and off is less significant than the inherent error of having the interaction always on. Additionally, the fact that the interaction is never switched off also means that bDAQC is only affected by bit-flip errors for the SQGs, as opposed to DQC and sDAQC, which are affected by bit-flip errors due to the entangling blocks. For the parameters used in our simulations, the total time of application of the QFT algorithm using the DAQC paradigm is shorter than the total time using DQC, thus making DAQC less susceptible to decoherence. The measurement error is common for all three paradigms, and its effect on fidelity is reflected as a shift in all values. However, it should be noted that as the number of qubits increases, the performance of the QFT algorithm decreases rapidly due to the increase in depth of the algorithm, which scales as $\mathcal{O}(N^2)$, where N is the number of qubits.

It is important to note that the results shown in Fig. 3.3 should not be taken as definitive, as the actual fidelity obtained in an experiment will depend on factors such as fabrication, architecture, and materials. However, the simulation results suggest that under fair comparison and considering the same noise sources, bDAQC obtains the highest value of fidelity. The distance between the fidelities of bDAQC and DQC is always remarkable in favor of bDAQC. Even though we have used a sensible choice for the noise model of the DAQC implementation, this model will not be accurate until it is compared against experimental data.

Our simulations indicate that bDAQC is a viable alternative to NISQ processors. With 3 qubits, bDAQC has a fidelity of approximately 90%, compared to 85% for sDAQC, and 80% for DQC. As the number of qubits increases, we have found that bDAQC's fidelity values continue to surpass those of DQC, although sDAQC's performance deteriorates due to the increasing number of analog blocks affected by bit-flip errors. With a larger number of qubits, the impact of noise sources results in fidelity values that are too low to yield meaningful results from the experiments.

We would like to point out that, while most quantum systems do not have ATA connectivity, with the notable exceptions of trapped ions and NMR, we have chosen to use ATA for the purpose of simplicity. Using the DAQC paradigm, it can be demonstrated that it is possible to simulate an N -qubit ATA Ising Hamiltonian with no more than $\frac{1}{2}N(N-1)$ NN Hamiltonians. This quadratic overhead is the worst-case scenario, but it can be significantly reduced if a pattern exists in our couplings [72]. However, this overhead (or worse) also applies to the DQC paradigm. Therefore, the fidelity values may differ when considering NN Hamiltonians, but the comparison between DQC and DAQC is fair, and the conclusion that bDAQC outperforms DQC remains valid.

3.1.3 Error mitigation in DA-QFT

To show the actual performance of DAQC, it is necessary to show how to adapt quantum error mitigation techniques to this paradigm. Specifically, we will prove that these techniques can effectively eliminate the inherent error that arises from bDAQC.

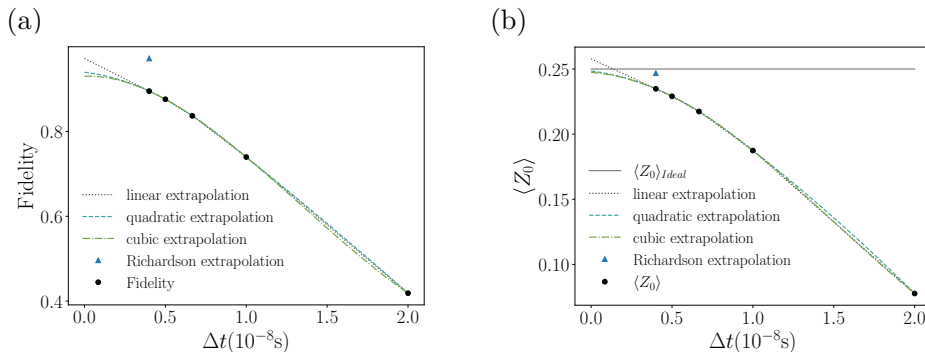


Figure 3.4: **Results of the error mitigation.** (a) Fidelity. (b) $\langle Z_0 \rangle$. Both fidelity and $\langle Z_0 \rangle$ attain accurate approximations of the ideal values. The simulations were made for the 8-qubit QFT circuit with $T_1 = 50 \mu\text{s}$, $p = 0.35$ and $g_0 = 1 \text{ MHz}$.

We utilized the zero noise extrapolation technique [86, 87] to demonstrate the impact of decoherence on the bDAQC QFT circuit. Specifically, we applied the circuit to an initial state of $|\psi_0\rangle = \sin \pi/4 |W8\rangle + \cos \pi/4 |GHZ8\rangle$ for varying coupling constants g_j , $j = 0, \dots, n_g$. By modifying the total circuit time as $t_j \alpha \propto g_j^{-1}$, we could vary the effect of decoherence. However, changes in g_j also affected the intrinsic error of the bDAQC paradigm. To address this issue, we selected $\Delta t_{\text{SQG}, i, j} = b_i/g_j$, such that the digital block time was inversely proportional to the coupling constant for a given g_j . By extrapolating the fidelity results for several values of g_j to the limit of zero decoherence and total time, we could eliminate the error caused by decoherence.

Furthermore, it is feasible to expand this method to eradicate the inherent error brought about by bDAQC. To achieve this, we iterate the prior approach for $i = 0, \dots, n_t$ values of $\Delta t_{\text{SQG}, i, j} = b_i/g_j$, utilizing the zero decoherence values to carry out the extrapolation to the zero Δt_{SQG} limit. We select an 8-qubit QFT circuit to implement this method as the number of qubits and quantum gates pose a significant challenge for cutting-edge quantum devices. We determine the fidelity of the ideal circuit, both without the error due to decoherence and bDAQC. The optimal fidelity is given by the linear extrapolation, which is approximately 0.97, indicating that the error-mitigated state is an excellent approximation of the ideal one. These results are depicted in Fig. 3.4(a). The same result is obtained for Richardson extrapolation, as in this case, the best outcome is obtained for the first-order approximation.

However, obtaining the quantum state after applying a quantum circuit is not always feasible as it can be a costly operation due to the use of tomography. Therefore, to assess the effectiveness of error mitigation for bDAQC, we need to select a second figure of merit that can be easily computed from the measurements of a quantum circuit. In this regard, we choose the expectation value of the observ-

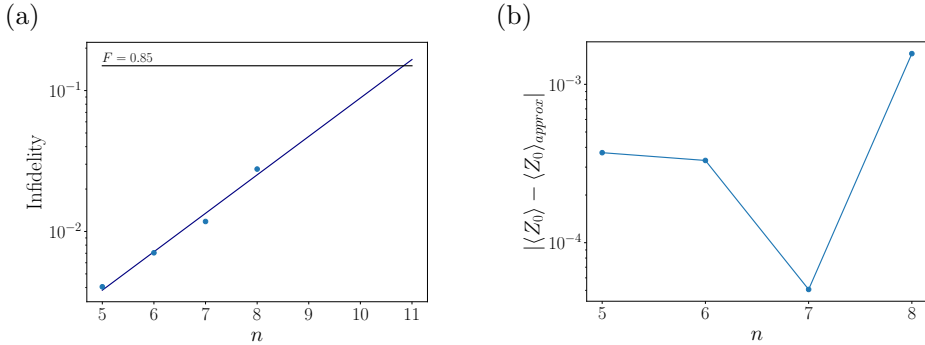


Figure 3.5: **Scaling of the error mitigation results with the number of qubits n for $n = 5, 6, 7, 8$.** (a) Infidelity. (b) $|\langle Z_0 \rangle - \langle Z_0 \rangle_{\text{approx}}|$. The simulations were made with $T_1 = 50 \mu\text{s}$, $p = 0.35$ and $g_0 = 1 \text{ MHz}$.

able $Z_0 = Z \otimes \mathbb{I} \otimes \dots \otimes \mathbb{I}$. The results of the expectation value approximation are presented in Fig. 3.4b. It can be observed that the best approximation of $\langle Z_0 \rangle$ is approximately 0.25, which is computed using Richardson extrapolation. The error $\varepsilon = |\langle Z_0 \rangle - \langle Z_0 \rangle_{\text{approx}}| = 0.0016$ indicates that we can obtain an extremely accurate approximation of the ideal expectation value from the noisy one.

The scaling behavior of the infidelity and the error in the estimation of $\langle Z_0 \rangle$ can also be studied as a function of the number of qubits. As depicted in Fig. 3.5(a), we can observe a decrease in fidelity as the number of qubits increases. However, by applying linear regression, we can still achieve fidelities of approximately 0.85 for up to 11 qubits. On the other hand, for the computation of $\langle Z_0 \rangle$ (Fig. 3.5(b)), the value to be approximated changes with each number of qubits, since the quantum circuit is modified accordingly. Therefore, we cannot observe a similar scaling as for the infidelity. Nonetheless, we obtain errors of a similar order of magnitude for the different number of qubits.

The figures presented in Fig. 3.4 demonstrate the effectiveness of using an adapted zero noise extrapolation to recover the ideal results from the bDAQC paradigm. This approach can be extended to sDAQC by solely studying the zero decoherence limit. Although other error mitigation techniques, such as an adaptation of Clifford Data Regression [88], have been considered, they result in a suboptimal outcome as they do not allow for independent error correction of the bDAQC extra interaction.

3.2 Quantum phase estimation

We studied the implementation of the QFT using the DAQC framework. Our findings prompted us to propose further investigation into the QPE algorithm. The QFT transforms the system state to reveal the phase information of eigenvalues, which can be estimated with high precision. In our previous research, we found that the DAQC paradigm outperforms the digital approach for larger numbers of qubits. As the inverse QFT is a part of the QPE algorithm, we expect similar performance. Our goal is to test the limits of these paradigms by obtaining concrete eigenvalues through the QPE algorithm. This poses a greater challenge for DQC, as it involves more two-qubit operations which are a major source of noise in current superconducting devices. We believe that using the analog evolution of DAQC will result in higher fidelity compared to a digital approach.

3.2.1 Description of the QPE algorithm and its DAQC implementation

The QPE algorithm estimates the phase φ that corresponds to the eigenvalue $e^{2\pi i\varphi}$ of the eigenvector $|u\rangle$ of a unitary operator U , where φ is unknown. This algorithm utilizes two quantum registers. The first register consists of t qubits, which determine the precision of the φ approximation. The second register encodes the eigenvector $|u\rangle$. The first register is initialized by applying a Hadamard gate to each qubit, followed by a series of controlled rotations on the second register using increasingly powers of two exponentiated by U . Finally, the inverse Fourier transform is applied to the first register, yielding an approximation of φ and the desired eigenvalue.

Given the unitary operator $P(\varphi)$,

$$P(\varphi) = \begin{pmatrix} 1 & 0 \\ 0 & e^{i2\pi\varphi} \end{pmatrix}, \quad (3.17)$$

we aim at obtaining φ such that $P(\varphi)|u\rangle = e^{i2\pi\varphi}|u\rangle$. We choose a 4-qubit register to estimate the phase φ that corresponds to the eigenvector $|u\rangle = |1\rangle = (0 \ 1)^T$. The complete QPE circuit is depicted in Fig. 3.6(a).

For the implementation of the QPE circuit, we took a different approach compared to the one used for the QFT. Rather than directly simulating the complete QPE algorithm Hamiltonian using the DAQC toolkit described in Chapter 2, we first decomposed the QPE quantum circuit into a universal set of gates, $\{R_x(\theta), R_z(\theta), \text{CNOT}\}$, which are commonly used in real superconducting quantum computers. Then, we translated each two-qubit gate (which, after the previous decomposition, would only be CNOT gates) into the DAQC paradigm by representing it as an analog block surrounded by single-qubit rotations. This approach allows us to explore different and more systematic alternatives to build DAQC quantum algorithms. In Fig. 3.6(b) we show both the decomposition of the QPE single- and two-qubit gates in terms of the

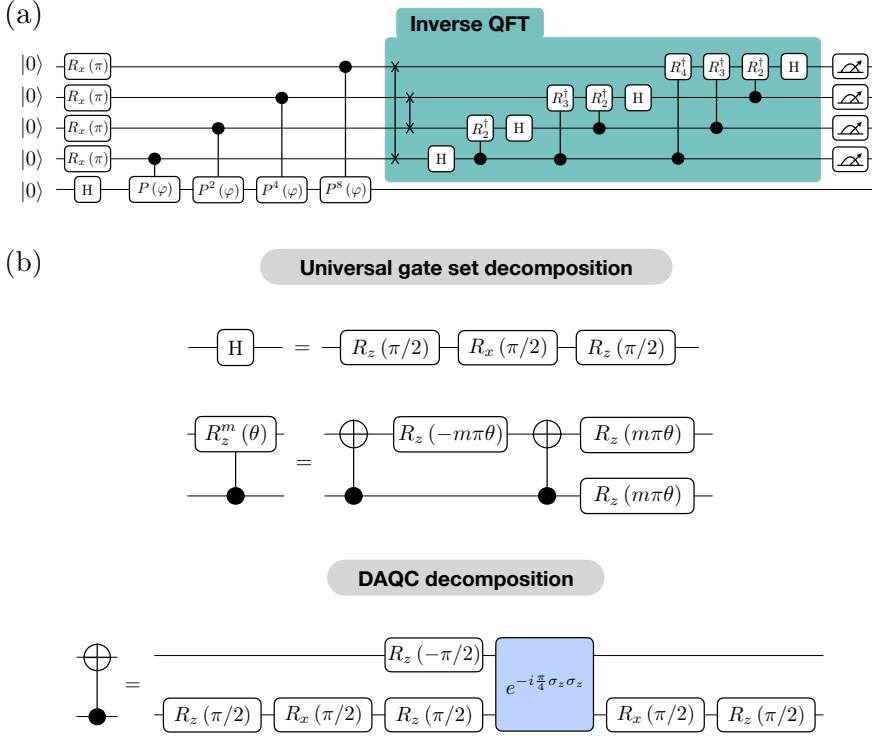


Figure 3.6: **QPE algorithm and its decomposition into a universal set of gates.** (a) QPE algorithm for $P(\varphi)$ (3.17), such that $P(\varphi)|u\rangle = e^{i2\pi\varphi}|u\rangle$. Hence, the final value of φ should be $\varphi = 1/3$. (b) Decomposition of the single- and two-qubit rotations in terms of the universal set of gates $\{R_x(\theta), R_z(\theta), \text{CNOT}\}$, as well as the decomposition of the CNOT gate using the DAQC method. Using this universal set of gate, the controlled rotations involved in the QPE algorithm are defined as $P^m(\varphi) \equiv R_z^m(\theta = \varphi)$ and $R_k^\dagger \equiv R_z^{m=1}(\theta = 1/2^k)$

mentioned set of gates, and the DAQC decomposition of the entangling CNOT gate.

3.2.2 Realistic implementation with experimental errors

Our developed tool simulates noise sources in quantum computers similarly to the QFT simulation in Sec. 3.1, but with a more versatile approach for implementing different algorithms. It takes into account the residual signals in neighboring qubits during SQG pulses (cross-talk), leading to a bit-flip error on all qubits after each digital block.

Simulations were conducted using noise parameters based on current state-of-the-art quantum computers. Approximate values of execution times for IBM quantum computer's basis gates were used, with $t_{R_x} = 10\text{ns}$, $t_{R_z} = 1\text{ns}$, and $t_{\text{CNOT}} = 300\text{ns}$.

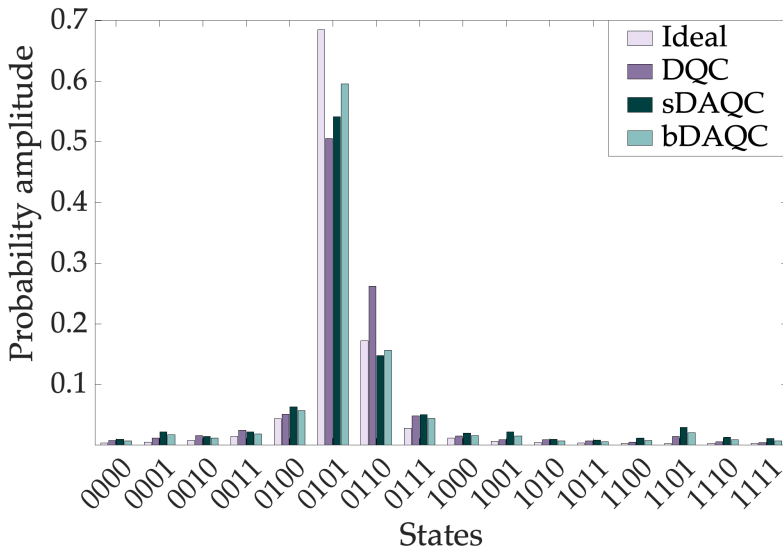


Figure 3.7: **QPE results for DQC, sDAQC and bDAQC.** Probability amplitudes of the final states generated by the QPE algorithm for various paradigms and sources of noise, with $\varphi = 1/3$ and the unitary operator specified in equation (3.17)

These values indicate that TQG execution times are one to two orders of magnitude larger than SQGs, consistent with increased noise from decoherence. The approximated error for these gates is TQGN = 0.08, with CNOT = $e^{-isH_{\text{CNOT}}}$ and $s \in \mathcal{N}(\pi/2, \text{TQGN})$. The error was also estimated from real IBM devices. A 0.0001 bit-flip probability was chosen for SQGs and $p_{\text{meas}} = 0.01$ for error measurement. Decoherence was considered with a T_1 time of $50\mu\text{s}$ and thermal ground state population of $p = 0.35$. Experimental errors defined in Ref. [?] were simulated using SQGN = 0.0005, sABN = 0.002, and bABN = 0.001. The coupling strength was set to $g_0 = 1$ MHz.

Using these parameters, we estimate the phase φ such that $P(\varphi)|u\rangle = e^{i2\pi\varphi}|u\rangle$, choosing $|u\rangle = |1\rangle = (0 \ 1)^T$. In ideal circumstances, Fig. 3.7 shows the highest probabilities belong to the 0101 and 0110 states, leading to the approximations $\varphi_{0101} = 5/16 = 0.3125$ and $\varphi_{0110} = 6/16 = 0.375$ with relative errors of 4.16% and 8.3% respectively, compared to the exact result $\varphi = 1/3$. Results show that under noisy conditions, both DQC and DAQC can still recover the maximum probability states. Fidelity was calculated with $F(\rho_{\text{ideal}}, \rho_{\text{DQC}}) = 0.88$, $F(\rho_{\text{ideal}}, \rho_{\text{bDAQC}}) = 0.87$, and $F(\rho_{\text{ideal}}, \rho_{\text{sDAQC}}) = 0.79$. DQC and bDAQC lead to similar fidelity values, but bDAQC shows better approximation of the solution, as maximum probability peaks are closer to the ideal.

If we calculate the weighted mean of the estimated phase, we obtain an estimate of phase of $\varphi_{\text{bDAQC}} = 0.348 \pm 0.119$ for bDAQC, $\varphi_{\text{sDAQC}} = 0.360 \pm 0.107$ for sDAQC

and $\varphi_{\text{DQC}} = 0.347 \pm 0.111$ for DQC. Using this figure of merit we conclude that the results for bDAQC and DQC are practically the same. However, if we use the majority voting as the estimation strategy, we can see that bDAQC outperforms the rest of circuits as we reach the closest approximation ($\varphi = 5/16$) faster than with the other approaches. This aligns with the results obtained when implementing the QFT in the previous section.

The results of our simulations of DAQC quantum algorithms, which included realistic noise sources in quantum computers and residual signals during SQG pulses, are highly promising. These findings demonstrate the significant potential of the DAQC paradigm for quantum computation. In the following chapter, we outline the implementation of a quantum algorithm to solve linear system of equations, that leverages the QPE subroutine and the QFT.

4 Digital-analog algorithm for solving linear systems of equations

IN this chapter, we will discuss the Harrow-Hassidim-Lloyd (HHL) quantum algorithm for solving linear systems of equations using the QPE as a subroutine. Similar to our approach for the QFT, we will first describe the algorithm as a Hamiltonian, which can then be simulated using the DAQC paradigm. We will mainly focus on the proposal for the DQC implementation of this algorithm since the DAQC simulations become trivial by using the toolkit presented in chapter 2. The content of this chapter is based on the work developed in [54].

In Section 4.1, we will provide a theoretical overview of the HHL algorithm, discussing how it works and its theoretical underpinnings. In Section 4.2, we will focus on the implementation of the HHL algorithm using the DQC paradigm, describing the specific steps involved and discussing potential issues that may arise during implementation. In Section 4.3, we will present an extension of the HHL algorithm to the DAQC framework, which allows for even more efficient simulations of the algorithm. Finally, in Section 4.4, we will use the HHL algorithm as an example to demonstrate how experimental noise can be mitigated through the co-design of quantum processor architecture.

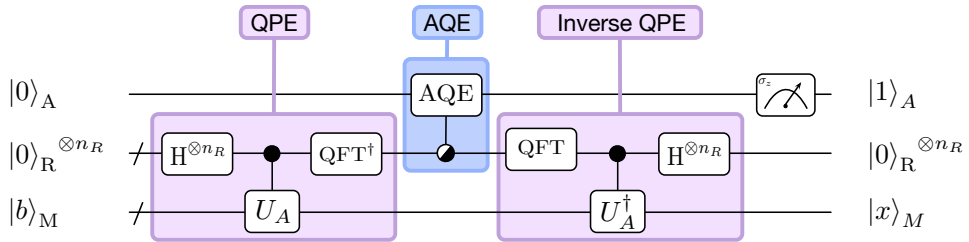


Figure 4.1: **Quantum circuit for the digital implementation of the HHL.** The implementation of the HHL algorithm can be divided into three subroutines: QPE to compute the eigenvalues of the problem matrix A , AQE to store the inverse of the eigenvalues in the amplitude of the ancilla qubit, and inverse QPE to reset the registers back to the ground state $|0\rangle_{\text{R}}^{\otimes n_{\text{R}}}$.

4.1 Quantum algorithm for solving linear systems of equations

The Harrow-Hassidim-Lloyd (HHL) quantum algorithm was proposed in 2009 [89] to solve linear systems of equations: given a matrix A and a vector \vec{b} , find a vector \vec{x} such that $A\vec{x} = \vec{b}$. Due to the significance of linear systems of equations in several fields of science and engineering, this algorithm has attracted significant attention. For a s -sparse system matrix of size $N \times N$ with condition number κ , which is given by the ratio of the maximal and minimal singular values of A , the HHL algorithm achieves a desired accuracy ϵ in a running time of $\mathcal{O}(\log(N)s^2\kappa^2\epsilon)$, compared to the best known classical algorithm with a running time of $\mathcal{O}(N_s\kappa/\log\epsilon)$ [90].

The HHL algorithm requires three sets of qubits: an ancilla qubit to store the inverse of the eigenvalues of the matrix problem in its amplitude, a register of n_{R} qubits to encode the binary representation of the eigenvalues of the problem matrix, and $n_{\text{M}} = \log_2(\dim|b\rangle)$ memory qubits to load $|b\rangle$ and store the output $|x\rangle$. The number of qubits needed depends on the size of the matrix A and the desired precision.

Additionally, the algorithm consists of three steps: QPE to compute the eigenvalues of A , ancilla quantum encoding (AQE) to map the inverse of the eigenvalues to the amplitude of the ancilla qubit, and inverse QPE to reset the registers back to the ground state $|0\rangle_{\text{R}}^{\otimes n_{\text{R}}}$. Fig. 4.1 shows a schematic representation of the algorithm, with the qubits divided into three sets and the steps delimited in colored boxes.

Since the HHL can be employed as a subroutine of a larger algorithm, the memory qubits might be initialized in the state $|b\rangle_{\text{M}} = \sum_{i=1}^N b_i|i\rangle$ as an outcome of the previous operations in the algorithm. If this is not the case, it is possible to prepare $|b\rangle$ using the procedure in Ref. [91] if b_i and $\sum_i |b_i|^2$ are efficiently computable. In

any case, the system is initially in the state $|0\rangle_A \otimes |0\rangle_R^{\otimes n_R} \otimes |b\rangle_M$.

The first step of the algorithm employs the QPE to calculate the eigenvalues λ_j of A and encode them in binary form into the state of the register qubits. Given $|b\rangle = \sum_{j=1}^N \alpha_j |u_j\rangle$, where $|u_j\rangle$ is the eigenvector basis of matrix A , the state of the system after QPE is

$$|0\rangle_A \otimes \sum_{j=1}^N \sum_{k=0}^{2^{n_R}-1} \alpha_j \beta_{k|j} |k\rangle_R \otimes |u_j\rangle_M, \quad (4.1)$$

with the coefficient $\beta_{k|j}$ defined as

$$\beta_{k|j} = \frac{1}{2^{n_R}} \sum_{y=0}^{2^{n_R}-1} e^{2\pi i y (\lambda_j - k/2^{n_R})}. \quad (4.2)$$

If the eigenvalues λ_j are perfectly n_R -estimated, they can be relabeled as $\widetilde{\lambda}_k \equiv k/2^{n_R}$ such that $\beta_{k|j} = \delta_{\lambda_k, \lambda_j}$ and the final state of the system is

$$|0\rangle_A \otimes \sum_{j=1}^N \sum_{k=0}^{2^{n_R}-1} \alpha_j |\widetilde{\lambda}_k\rangle_R \otimes |u_j\rangle_M. \quad (4.3)$$

After encoding the estimated eigenvalues in the register qubits, the AQE maps them into the amplitude of the ancilla qubit. The resulting state of the ancilla is

$$\sum_{j=1}^N \sum_{k=0}^{2^{n_R}-1} \left(\sqrt{1 - \frac{C^2}{\widetilde{\lambda}_k^2}} |0\rangle_A + \frac{C}{\widetilde{\lambda}_k} |1\rangle_A \right) \alpha_j |\widetilde{\lambda}_k\rangle |u_j\rangle, \quad (4.4)$$

with normalization constant $C \leq 1$ chosen to be $\mathcal{O}(1/\kappa)$.

Finally, the inverse QPE must be performed to uncompute the estimated eigenvalues $|\lambda_j\rangle$ on the register qubits and reset them to their initial state of $|0\rangle^{\otimes n_R}$. After this step, the resulting state of the system is given by

$$\sum_{j=1}^N \sum_{k=0}^{2^{n_R}-1} \left(\sqrt{1 - \frac{C^2}{\widetilde{\lambda}_k^2}} |0\rangle_A + \frac{C}{\widetilde{\lambda}_k} |1\rangle_A \right) \otimes |0\rangle_R \otimes \alpha_j |u_j\rangle_M \quad (4.5)$$

The ancillary qubit must be measured in the Z -basis to obtain the normalized solution of the linear equation. A successful measurement outcome of $|1\rangle_A$ results in the state

$$\frac{1}{C} \sum_{j=1}^N \frac{\alpha_j}{\widetilde{\lambda}_k} |u_j\rangle_M, \quad (4.6)$$

representing the solution of the linear equation, up to a normalization factor C . The solution is a pure state when the matrix A is perfectly n_R -estimated. However, if there is an eigenvalue of A that is not perfectly estimated, the total state in Eq.(4.5) is an entangled state, causing the solution state in Eq. (4.6) to be a mixed state.

4.2 Proposal for digital implementation: the ancilla quantum encoding

The digital implementation of the direct and inverse QPE in this algorithm is relatively simple, but, to our knowledge, there is no known explicit implementation for the AQE subroutine. In this section, we propose a systematic method for implementing the AQE that is independent of the problem matrix A and does not require classical hybridization, as found in the HHL algorithm. Due to its relevance, we focus on computing $f(\lambda) = 1/\lambda$, but this subroutine can perform any function $f(A)$ that satisfies the conditions outlined in Ref. [89].

In the literature, the AQE routine has been circumvented by employing different tricks. For instance, by using previous knowledge about the eigenvalues of the matrix A . Then, it is straightforward to tailor the best set of rotations to map those values into the amplitude of the ancilla qubit. Several experimental implementations in different experimental platforms [92, 93, 94, 95] have followed this approach to solve 2×2 linear system of equations. There is an alternative implementation of the HHL for which no spectral information of the matrix A is required. In Ref. [96] a quantum-classical hybridization of the algorithm is proposed. They repeatedly perform the QPE to obtain a n_R -bit description of the eigenvalues of A . Then, they determine the simplest circuit implementation to perform the AQE, tailored for those eigenvalues. Once the AQE is determined, it is possible to perform the complete HHL algorithm. However, this alternative fails if the vector $|b\rangle$ is not efficiently prepared. Despite the intrinsic interest of these approaches, both of them jeopardize the advantage of the algorithm, since they presume previous knowledge about the eigenvalues of the matrix A .

The AQE maps the superposition state of the registers into the amplitude of the ancilla qubit by means of the application of different controlled rotations. These rotation operations control each register qubit and act on the ancilla. To perform the AQE step, we will first rotate the system by applying the unitary gate V , which is defined as

$$V = \frac{1}{\sqrt{2}} \begin{pmatrix} -i & i \\ 1 & 1 \end{pmatrix}, \quad (4.7)$$

and then, apply a series of rotations around the Z -axis in the ancilla qubit controlled by the state of the register qubits. This operation is defined as

$$U_{\text{AQE}} = \sum_{p=0}^{2^{n_R}-1} R_z(-\phi(p))_A \otimes |\vec{b}_p\rangle\langle\vec{b}_p|_R, \quad (4.8)$$

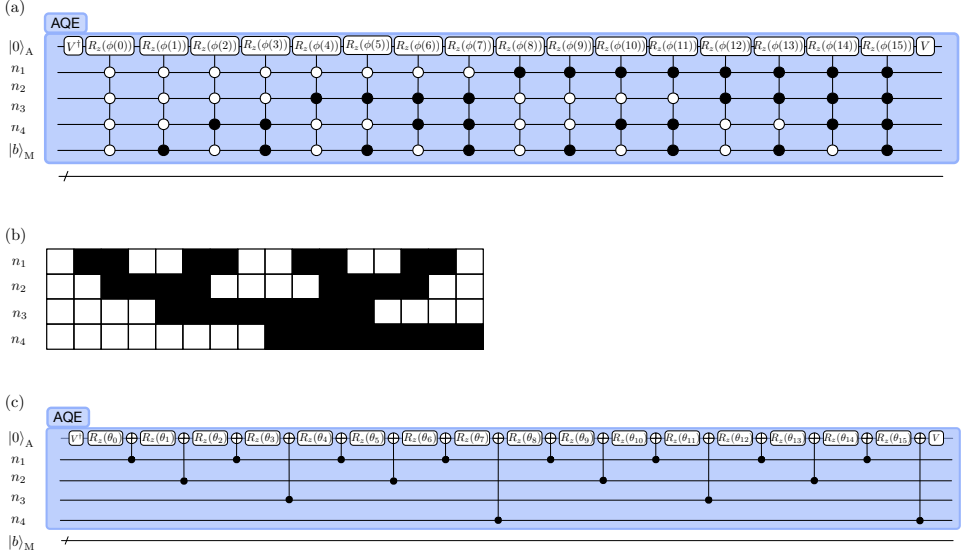


Figure 4.2: **(a) Multi-qubit decomposition of the AQE step for a $n_R = 4$ qubits example.** The rotations V and $R_z(\phi(p))$ performed on the ancilla qubit are defined in Eqs. (4.7) and (4.9), respectively. **(b) Binary reflected 4-bit Gray code to define the positions of the control nodes.** The black and white rectangles refer to bit values one and zero, respectively. **(c) AQE implementation using only single- and two-qubit gates.** Using the 4-bit Gray code and following Ref. [97], it is possible to decompose the multi-qubit AQE implementation into single-qubit rotations around the z -axis and CNOTs. The value of each θ_i follows from Eq. (4.13).

where,

$$R_z(\phi(p)) = \begin{pmatrix} e^{-i\phi(p)/2} & 0 \\ 0 & e^{i\phi(p)/2} \end{pmatrix}, \quad (4.9)$$

$$\phi(p) = \begin{cases} 0 & \text{if } p = 0 \\ 2 \arcsin \frac{1}{p} & \text{otherwise} \end{cases}. \quad (4.10)$$

The bit-string \vec{b}_p is the binary representation of the decimal number p , in other words, $p = \sum_{i=0}^{n_R} (\vec{b}_p)_i 2^{n_R-i}$. In Fig. 4.2(a), we show an example of the AQE routine for a $n_R = 4$ qubits case.

The AQE involves the application of several multi-controlled gates, leading to an exponential increase in the number of gates required for its implementation. According to Ref. [97], it can be decomposed into 4^{n_R} single-qubit gates and $4^{n_R} - 2^{n_R+1}$ CNOT gates, making it a challenging task for NISQ devices. The depth of the AQE step increases with the number of register qubits, which also determines the precision and accuracy of the HHL algorithm.

In order to implement the multi-controlled operations using only single- and two-

qubit gates acting on the ancilla, we adopt the approach from Ref.[97]. They provide an equivalent circuit that performs subsequent rotations with modified angles, alternating with CNOT gates between control and target qubits. As a result, the operation can be implemented with the circuit depicted in Fig. 4.2(c). Here, the angles θ_i are the modified angles of the single-qubit rotations, related to the original angles by

$$\begin{pmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_{2^{n_R}} \end{pmatrix} = M^{-1} \begin{pmatrix} \phi(0) \\ \phi(1) \\ \vdots \\ \phi(2^{n_R} - 1) \end{pmatrix}, \quad (4.11)$$

where M and M^{-1} are defined as:

$$M_{ij} = (-1)^{\text{bin}(i-1) \cdot g(j-1)} \text{ and } M^{-1} = \frac{1}{2^{n_R}} M^T. \quad (4.12)$$

Here, $\text{bin}(i)$ is the n_R -bit binary representation of the integer i , $g(j)$ is the j -th string of the binary reflected Gray code (counting from 0), and the dot represents the bit-wise product. By substituting the expression for $\phi(b)$, then

$$\theta_i = \frac{1}{2^{n_R}} \sum_{j=1}^{2^{n_R}-1} (-1)^{\text{bin}(j) \cdot g(i-1)} \arcsin \frac{1}{j}. \quad (4.13)$$

The accuracy of the algorithm's output in comparison with the actual solution $|x\rangle$ of the equation $A|x\rangle = |b\rangle$ is expressed as $\| |x\rangle - |\tilde{x}\rangle \| = \mathcal{O}(\kappa/2^{n_R})$, where $|\tilde{x}\rangle$ is the solution obtained by the algorithm and κ is the condition number of the problem matrix A . To estimate the expected value of an observable Ω with a limited number of samples N_s , the error of the estimation scales as $\mathcal{O}(1/\sqrt{N_s})$. Therefore, the overall expected error, ϵ , of estimating Ω in $|x\rangle$ by measuring $|\tilde{x}\rangle$ is

$$\epsilon = \mathcal{O} \left(\frac{1}{\sqrt{N_s}} + \frac{\eta}{2^{n_R}} \right), \quad (4.14)$$

where η is a constant of $\mathcal{O}(\kappa)$. To determine the optimal number of register qubits n_R , we must ensure that no term in Eq. (4.14) is dominant. This implies that the sensible amount of register qubits n_R is of $\mathcal{O}(\log[\kappa\sqrt{N_s}])$, which only depends on the number of samples, but not on the size of the problem matrix A .

The *generalized*-HHL (gHHL) algorithm aims to compute $\vec{x} = f(A)\vec{b}$, where $f(A)$ is a matrix function. The method can be easily adapted to perform this task. The AQE is used to perform the following operation on the ancilla qubit,

$$|0\rangle_A |\widetilde{\lambda}_k\rangle_R \xrightarrow{\text{AQE}} \left(\sqrt{1 - f(\lambda^2)} |0\rangle_A + f(\lambda) |1\rangle_A \right) |\widetilde{\lambda}_k\rangle_R. \quad (4.15)$$

To accomplish this, we set the rotation angles of the multi-controlled operations to $\phi(p) = 2 \arcsin f(p/2^{n_R})$. However, the error bounds are only met if the function

$f(\lambda)$ satisfies the condition

$$\left| \left(\frac{df}{dx} \right)^2 \left(1 + \frac{1}{1 - f(x)^2} \right) \right| < \eta^2 \quad \text{in the interval } x \in [0, 1), \quad (4.16)$$

which can be derived from Lemma 2 in the supplementary material of Ref. [89]. Here, η is an arbitrary constant of order $\mathcal{O}(1)$. This implies that, in practice, one would need to construct an auxiliary function F that satisfies the bound and behaves similarly to $f(\lambda)$ in a given interval, similar to normalizing a vector for its use as a quantum state.

4.3 Digital-analog implementation

As previously mentioned, the HHL algorithm can be broken down into three steps (QPE, AQE, and inverse QPE). QPE itself can be further divided into two subroutines: controlled-Hamiltonian evolution and inverse quantum Fourier transform. Dividing each step into smaller subroutines makes it easier to describe the HHL algorithm within the DAQC framework, as some of these subroutines have already been thoroughly discussed in previous chapters.

To perform the controlled-Hamiltonian evolution, which is the first part of the QPE, we assume that the problem matrix A has an M -body decomposition, where $M \ll \log_2 \dim(A)$. This allows the matrix to be efficiently uploaded using the DAQC protocol, as described in Ref. [51]. The next step in the QPE subroutine is the implementation of the inverse quantum Fourier transform on the register qubits, which was discussed in detail in chapter 3.

To implement the AQE step under the DAQC paradigm, we first decompose it into single-qubit rotations (Hadamard gates and rotations around the X-axis) and controlled-Z (cZ) gates, as shown in Fig. 4.3(a). To obtain a sDAQC decomposition, we construct the Hamiltonian of the cZ gates, which takes the form of a two-body Ising Hamiltonian:

$$H_{cZ} = -\frac{1}{2}\sigma_z \otimes \mathbb{I} + \frac{1}{4}\sigma_z \otimes \sigma_z. \quad (4.17)$$

The cZ gate can be implemented using either the step-wise or the bang approach of the DAQC framework. Fig. 4.3(b) shows a scheme of both protocols for the cZ gate.

Without loss of generality, we assume that the interaction between the qubits of our quantum processor is a homogeneous two-body Ising Hamiltonian, H_{int} . Each cZ gate performed between the ancilla and one of the register qubits can then be

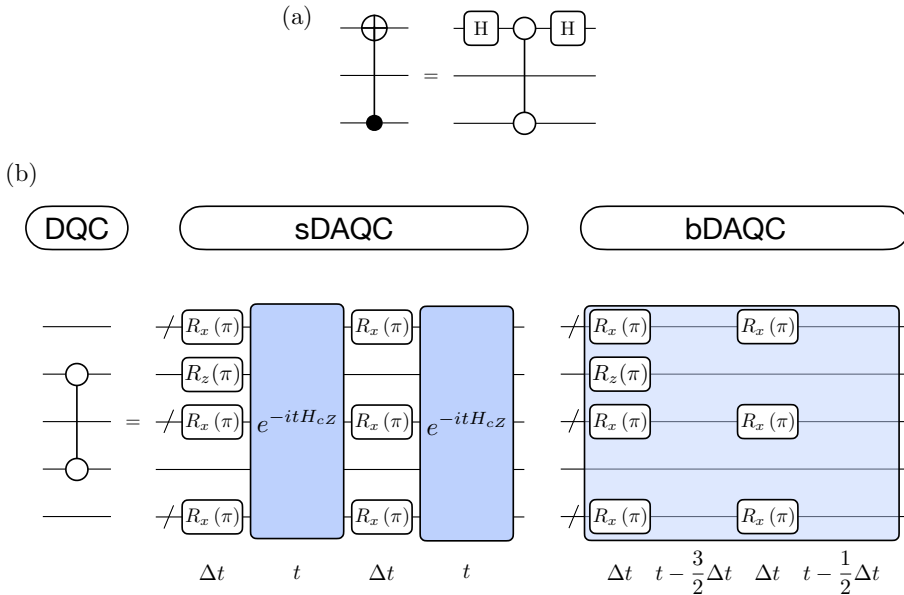


Figure 4.3: (a) **Decomposition of a CNOT gate in terms of Hadamard and cZ gates.** (b) **DAQC decomposition of a cZ gate.** To perform a cZ gate using sDAQC, two analog blocks with duration $t = \pi/8$ are necessary. For the digital blocks, all qubits except those in the cZ -gate must undergo a π -radian rotation around the x -axis before and after the first analog blocks. In the case of the cZ -gate qubits, a rotation around the z -axis should be applied either before or after the analog block, as the interaction Hamiltonian commutes with the $R_z(\pi)$ gate. In bDAQC, the single-qubit rotations are performed after the interaction.

decomposed into two analog blocks as

$$U_{cZ} = e^{i\frac{\pi}{4}Z_A \otimes Z_R} = \left(\bigotimes_{\substack{k=1 \\ k \neq i}}^{n_R} X_k \right) e^{i\frac{\pi}{8}H_I} \left(\bigotimes_{\substack{k=1 \\ k \neq i}}^{n_R} X_k \right) e^{i\frac{\pi}{8}H_I}. \quad (4.18)$$

This decomposition is illustrated in Fig. 4.3(b) with the sDAQC implementation of a cZ gate.

To implement the complete AQE in the DAQC paradigm, we decompose every cZ into digital and analog blocks, as previously described. The analog blocks take the same amount of time t to act, and the digital steps take a fixed time Δt to be applied, as illustrated in Fig. 4.3(b). As it has been repeatedly mentioned in this thesis, in terms of scalability it is preferable to follow the bDAQC approach. The explicit DAQC description of the AQE step includes at most 2^{n_R} analog blocks and $(n_R + 1)2^{n_R} + 1$ single-qubit gates, which can be improved with optimization techniques.

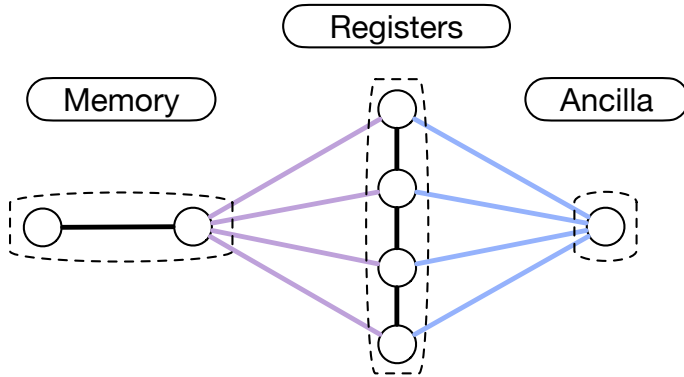


Figure 4.4: **Kite-like connectivity of the quantum processor.** The register qubits must be interconnected with each other, the ancilla qubit, and the memory qubits. However, the ancilla and memory qubits do not require a direct connection.

4.4 Co-designed processor architecture for the HHL algorithm

The concept of co-design in quantum computing involves the simultaneous design of a quantum processor together with the implementation of a specific algorithm. This approach leverages the unique features and limitations of a quantum platform to optimize both the architecture and implementation of the algorithm, resulting in a more efficient and effective system. In particular, for the implementation of the HHL algorithm on NISQ devices, in Ref. [54] we propose a co-designed “kite-like” architecture that reduces the number of required SWAP gates.

The HHL algorithm requires specific connections between its three sets of qubits: the register, ancilla, and memory qubits. This co-designed architecture, shown in Fig. 4.4, takes into account these connections and implements a kite-like structure that satisfies the requirements of the different steps in the HHL algorithm: the register qubits are connected among them and, simultaneously, to the ancilla and memory qubits, whereas the ancilla and the memory sets need not be connected directly. This optimization of the architecture leads to improved performance and reduced computational overhead.

It is important to note that the term “qubits” in this context refer to logical qubits, which may be constituted by a group of physical qubits that work together to perform the necessary operations and connectivity. By customizing both the architecture and implementation of the HHL algorithm for the specific features of the quantum platform, the co-design approach aims to maximize the performance and efficiency of the system.

With the conclusion of this chapter, we bring to an end the matryoshka construction of algorithms that we started with the QFT in chapter 3. In the next chapter, we will examine the application of the DAQC framework to variational quantum algorithms.

5

**Digital-analog quantum
approximate optimization algorithm**

IN this chapter, we will show how the DAQC paradigm can also be used with variational algorithms. These algorithms utilize classical optimization techniques and parametrized quantum circuits to increase the applicability of noisy quantum processors that have few qubits. One such algorithm that is most likely to prove useful in the near term is the quantum approximate optimization algorithm (QAOA) [98]. This chapter contains the work developed in Ref. [55], in which we show that the QAOA is a natural algorithm for the DAQC paradigm.

In Section 5.1, we introduce and thoroughly explain the QAOA. In Section 5.2, we expand the algorithm to the DAQC framework. Finally, in Section 5.3, we compare the performance of the QAOA algorithm under both the DQC and DAQC paradigms.

5.1 The Quantum Approximate Optimization Algorithm

The QAOA algorithm is a hybrid quantum-classical algorithm that operates in discrete time and is utilized for computing solutions to combinatorial optimization problems. Originally, the algorithm was found to produce superior approximation ratios compared to the best classical algorithm for solving MAX-E3LIN2 problems [99]. However, this result was later surpassed by a quantum-inspired classical algorithm [100]. Furthermore, research by Jiang et al. [101] has demonstrated that QAOA can replicate the square root scaling of Grover’s search algorithm, achieving the same speed-up without the need for the Grover mixing operator. Additionally, Hadfield et al. have discovered that QAOA driving operators can be adjusted in such a way that a diverse range of problems can be solved without resorting to high-order penalty terms, which are typically utilized in annealing- or adiabatic-based approaches [102].

QAOA is a discrete-time hybrid quantum-classical algorithm for computing solutions to problems in combinatorial optimization. The classical optimizer tunes the $2p$ parameters $\vec{\gamma}, \vec{\beta}$ of a quantum circuit to maximize an objective function corresponding to high-quality solutions of a combinatorial optimization problem. In QAOA, an ansatz state

$$|\vec{\beta}, \vec{\gamma}\rangle = \prod_{p'=1}^p e^{i\beta_{p'} H_D} e^{i\gamma_{p'} H_P} |+\rangle^{\otimes n} \quad (5.1)$$

is generated on a quantum processor using p repetitions of two Hamiltonians: a problem Hamiltonian H_P and a driver Hamiltonian H_D . A schematic quantum circuit of the algorithm is shown in Fig. 5.1. The problem and the driver Hamiltonians are defined as

$$H_P = \sum_{z=0}^{2^n-1} C(z)|z\rangle\langle z|, \quad \text{and} \quad H_D = \sum_{i=1}^n \sigma_x^{(i)}, \quad (5.2)$$

where C is the objective function value for input strings z , and $\sigma_x^{(i)}$ refers to the Pauli matrix applied to the i -th qubit of the system. H_P is chosen for its simplicity since it is a non-interacting Hamiltonian, while still enabling population transfer between any two given states.

To find a solution, the QAOA circuit is repeatedly executed, and the output string is measured to calculate the expectation value of H_P under the QAOA ansatz state for the current parameters, as shown in Figure 5.1. This value is handed to a classical optimizer, with the goal of producing new parameters through a classical black-box optimization strategy. The expectation value of the problem Hamiltonian,

$$\langle H_P \rangle_{\vec{\beta}, \vec{\gamma}} = \langle \vec{\beta}, \vec{\gamma} | H_P | \vec{\beta}, \vec{\gamma} \rangle, \quad (5.3)$$

is then recomputed, and the process is repeated until a satisfactory solution to the problem is found. The success of QAOA is measured by the *mean approximation*

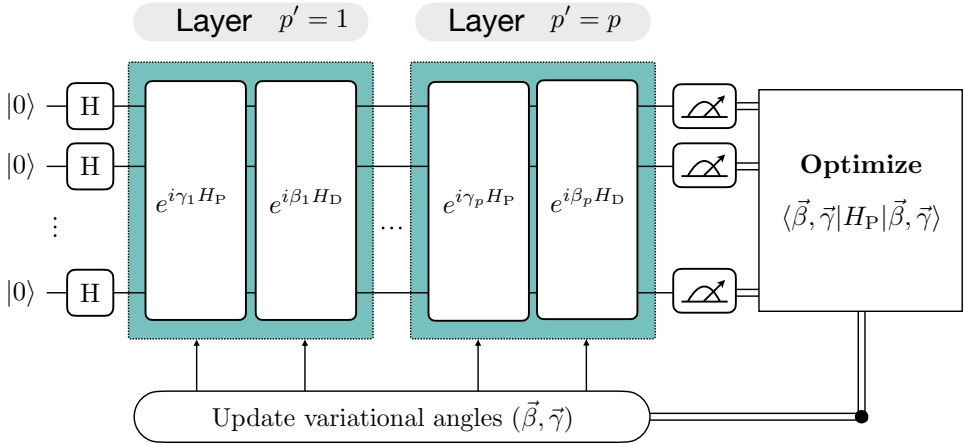


Figure 5.1: **QAOA state preparation circuit.** The algorithm alternates the application of H_P and H_D Hamiltonians to an equal superposition of qubits. The resulting qubit states are then measured and a cost is calculated. A classical optimization algorithm is utilized to minimize this cost by varying the angles γ and β .

ratio, defined as

$$\frac{\langle H_P \rangle_{\vec{\beta}, \vec{\gamma}}}{\max_{\psi} \langle \psi | H_P | \psi \rangle}. \quad (5.4)$$

Combinatorial optimization problems with clauses encompassing at most two bits can be expressed in terms of two-qubit ZZ interactions and single-qubit- Z rotations. Problems in which the clauses are local to more bits require higher order terms and are therefore generally out of reach of NISQ quantum computers. In the literature, there are two well-studied optimization problems, namely MAX-CUT and MAX-2-SAT, that involve only terms of order up to 2. MAX-CUT is defined on a problem graph whose vertices are binary variables, and the goal is to partition the graph in a way that maximizes the number of edges between different parts of the partition. The clauses of the problem or edges of the equivalent graph are of the type XOR between problem variables. XOR admits the truth table $00, 01, 10, 11 \rightarrow 0, 1, 1, 0$ which can be decomposed into a Z -based Hamiltonian following theorem 10 of Ref. [103]. A MAX-CUT clause, therefore, manifests in the problem Hamiltonian as

$$H_{C,jk} = \frac{1}{2} \left(\mathbb{I} - \sigma_z^{(j)} \sigma_z^{(k)} \right) = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}. \quad (5.5)$$

The identity in this expression has no effect other than to keep the Hamiltonian non-negative such that the diagonal corresponds to the number of edges a given allocation cuts.

5.2 Digital-Analog Quantum Approximate Optimization Algorithm

Let us assume that the interaction Hamiltonian of the quantum processor of interest is as follows

$$H_{\text{int}}(t) = f(t)H_{\text{resource}} + \alpha \sum_{i=1}^n \left(x_i(t)\sigma_x^{(i)} + z_i(t)\sigma_z^{(i)} \right), \quad (5.6)$$

with

$$H_{\text{resource}} = \sum_{j < k}^n r_{jk} \sigma_z^{(j)} \sigma_z^{(k)}. \quad (5.7)$$

In the sDAQC scheme, we assume control over the parameters f , x_i , and z_i , each of them taking values from $\{0, 1\}$. In the bDAQC scheme, f is always set to 1 and only the single-qubit parameters may be altered. The single-qubit terms are stronger than the resource Hamiltonian H_{resource} by the factor $\alpha \geq 1$ and, in typical applications, α is expected to fall between 10 – 1000 depending on architecture [104, 105]. Though current devices tend to exhibit a ratio of single-qubit rotation speed to interaction strength at the lowest end of this range, they have little to gain from faster single-qubit operations, since they are typically limited by two-qubit interaction times and fidelity. We, therefore, expect that a device optimizing for DA applications could be engineered for greatly higher ratios α . During the driving in bDAQC, all single-qubit x_i operations are set to 1, z_i terms to 0, giving a driver Hamiltonian of

$$H_{\text{bDAQC-Driver}} = \sum_{j < k} r_{jk} \sigma_z^{(j)} \sigma_z^{(k)} + \alpha \sum_{i=0}^n \sigma_x^{(i)} \quad (5.8)$$

applied for device time given by the variational parameter β divided by the driver strength α with $\beta \in [0, \pi]$. During the analog block steering operations, we use a similar Hamiltonian in which only a specific set of single-qubit x_i terms are active. As described in chapter 2, we wish to implement a full X -gate before and after each resource block. The time to apply this gate will be $\Delta t = \frac{\pi}{\alpha}$. Applying the digital-analog QAOA (DA-QAOA) device Hamiltonian for a single QAOA layer thus affects the following unitary

$$U_{\text{DA-QAOA}} = \mathcal{T} \exp \left(-i \int_{t=0}^{t_{\text{total}}} H_{\text{int}}(t) dt \right), \quad (5.9)$$

with $\vec{x}(t)$ defined by the matrix inversion procedure described in chapter 2, \mathcal{T} is the time-ordering meta-operator and $\vec{z}(t)$ used in the case that we are solving a SAT problem. t_{total} is the sum of all times in the non-negative DAQC time vector

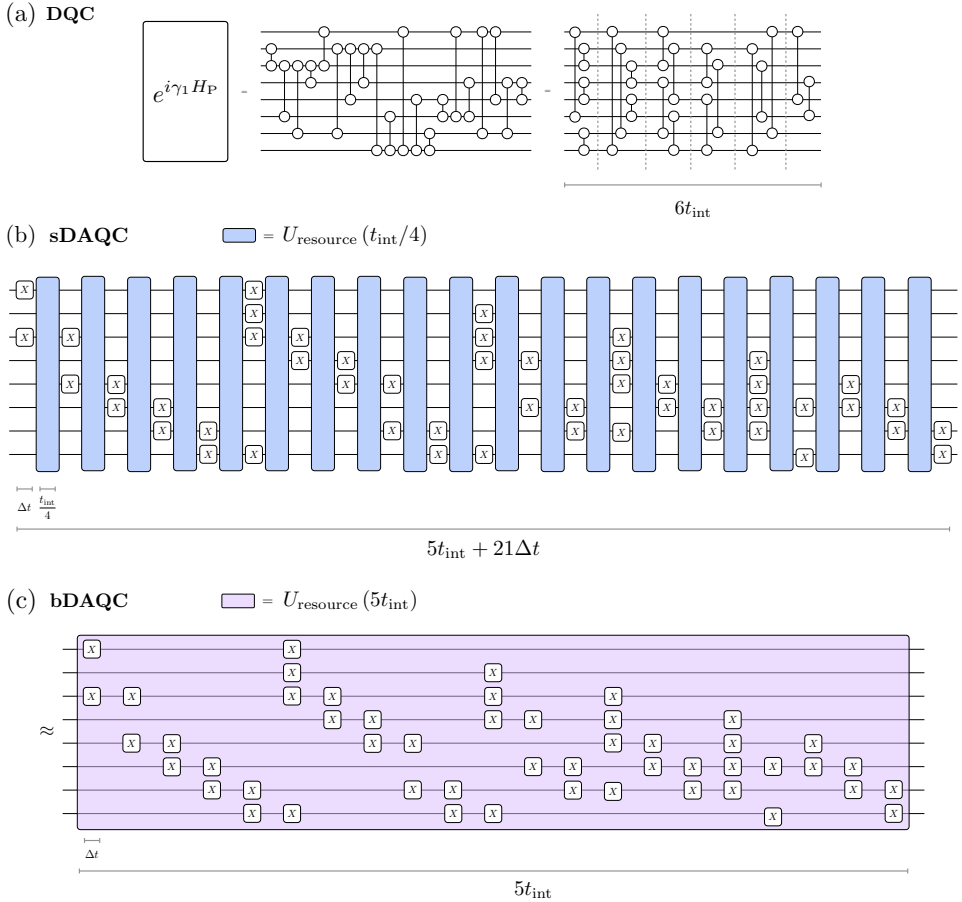


Figure 5.2: **Circuit compilations for an 8-qubit MAX-CUT problem on a 5-regular graph.** (a) **DQC decomposition of the MAX-CUT problem Hamiltonian into $\sigma_x\sigma_x$ interactions.** The lines connected to two open circles represent those interactions applied for time t_{int} . On the right-hand side, we show how to parallelize the circuit into six time steps, each of which sees one qubit interact with only one other qubit at a time. Six time steps correspond to the maximum degree of a vertex plus one. (b) **sDAQC decomposition of the same circuit.** A resource all-to-all homogeneous Ising Hamiltonian is turned on and off, punctuated by single-qubit gates. The X gate corresponds to a rotation of π radians around the x -axis and takes Δt time to be applied. This decomposition requires 20 uses of the resource Hamiltonian and 50 single-qubit operations. The time taken to apply the problem Hamiltonian is $5t_{\text{int}} + 21\Delta t$. (c) **Decomposition of the same problem into the bDAQC scheme.** The resource Hamiltonian is turned on throughout the entire procedure. This circuit only approximates the time evolution invoked by the QAOA problem Hamiltonian but can be carried out in time $5t_{\text{int}}$ and also with 49 single-qubit X operations. In the limit of infinitely fast X gates, (c) is equivalent to (a) and (b).

multiplied by the variational parameter γ in addition to the driving time β/α . A depiction of this device Hamiltonian used to apply a MAX-CUT problem Hamiltonian is presented in Fig. 5.2.

5.3 Performance of the QAOA under different computational paradigms

In this section, we present the main computational results of the work in Ref. [55]. In section 5.3.1, we compare the performance of bDA-QAOA in comparison with the standard QAOA circuit for a set of chosen randomly generated problems. Afterward, in section 5.3.2 we analyze this further and show the benefits of the DAQC framework in the application of variational quantum algorithms. Finally, in section 5.3.3, we examine the impact of experimental errors inherent to NISQ computing on the DAQC framework when utilizing variational algorithms such as the QAOA.

5.3.1 Performance of the bDA-QAOA

In bDA-QAOA, we perform QAOA using the ansatz state prepared by applying QAOA layers of the form described in equation (5.9) as

$$|\vec{\beta}, \vec{\gamma}\rangle^{\alpha, \text{DA}} = U_{\alpha\text{-DA-QAOA}}|+\rangle^{\otimes n}. \quad (5.10)$$

The bDA-QAOA algorithm introduces errors through the misspecification of the problem and driver Hamiltonians. Among these two, the misspecification of the problem Hamiltonian is likely to cause more significant error, as it requires a longer time to perform on the device and it is not tailored to a specific problem. This problem is not unique to quantum optimization and is known in quantum annealing literature as “ J chaos” where key aspects of the problem are not accurately represented in the dynamics of the annealing device. Without error mitigation strategies, these issues can severely impact the performance of adiabatic quantum computing [106].

bDA-QAOA is related to quantum random walk algorithms [107] and adiabatic quantum computing in that both the problem Hamiltonian and single-qubit driving operators are executed simultaneously. Therefore, one might expect that running the problem Hamiltonian alongside the driver in QAOA would not have a negative effect, as quantum random walks and diabatic quantum computing are active research areas [108, 109]. Simulations of bDA-QAOA, where the resource Hamiltonian is identical to the problem Hamiltonian, showed no significant detrimental impact when compared to error-free QAOA, as no DA steering single-qubit operations were required.

However, coherent errors in the bDA-QAOA algorithm, which utilizes a non-problem-specific resource Hamiltonian, are expected to be more detrimental than

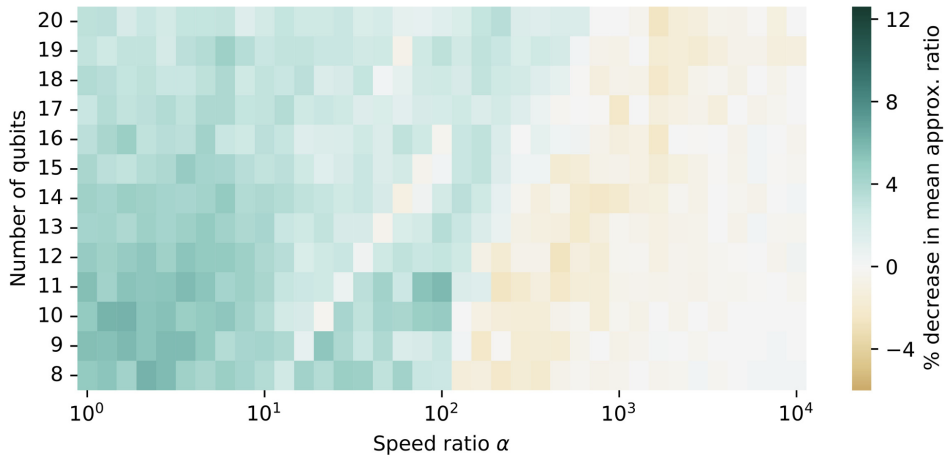


Figure 5.3: **Average percentage difference between the mean approximation ratio of bDA-QAOA and error-free QAOA.** We have calculated it by averaging over 50 randomly generated MAX-CUT problems with a constant filling factor of $p_{\text{clause}} = 0.7$. Blue (dark region on the far left side of the figure) colors indicate that the bDA-QAOA ansatz state is worse than that provided by error-free QAOA, whereas shades of brown (the light gray region immediately left of the white region of zero error) indicate an improvement of the bDAQC over error-free QAOA. On the x -axis, the ratio α of single-qubit to problem Hamiltonian term strength is seen where error-free QAOA exists in the limit as this ratio becomes infinite.

errors in QAOA with an always-on active problem Hamiltonian. These errors will result in a less problem-specific QAOA ansatz state, leading to a lower expected approximation ratio at a given depth. Figure 5.3 displays the mean approximation ratio achieved by the bDA-QAOA ansatz state. For high values of α , bDA-QAOA performs similarly to error-free QAOA, as expected. In the intermediate regime, minor increases in the mean approximation ratio are observed. Conversely, in the case of low α , we observe consistently worse results for bDA-QAOA, due to problem misspecification induced by coherent DAQC errors.

5.3.2 Variational resilience of DA-QAOA to digital-analog errors

Variational quantum algorithms, like QAOA, are considered to have better error tolerance compared to non-variational algorithms. This is because the classical optimizer can compensate for systematic coherent over- or under-rotations and other errors [110], making variational algorithms a suitable choice for NISQ quantum computing. QAOA finds the optimal parameters $\vec{\beta}^*$, $\vec{\gamma}^*$ by maximizing $\langle H_P \rangle_{\vec{\beta}, \vec{\gamma}}$. However, when the QAOA ansatz operators are changed to those of bDAQC, the parameters $\vec{\beta}^*$, $\vec{\gamma}^*$ that maximize $\langle H_P \rangle_{\vec{\beta}, \vec{\gamma}}$ may not necessarily maximize $\langle H_P \rangle_{\vec{\beta}, \vec{\gamma}}^{\alpha, \text{DA}}$,

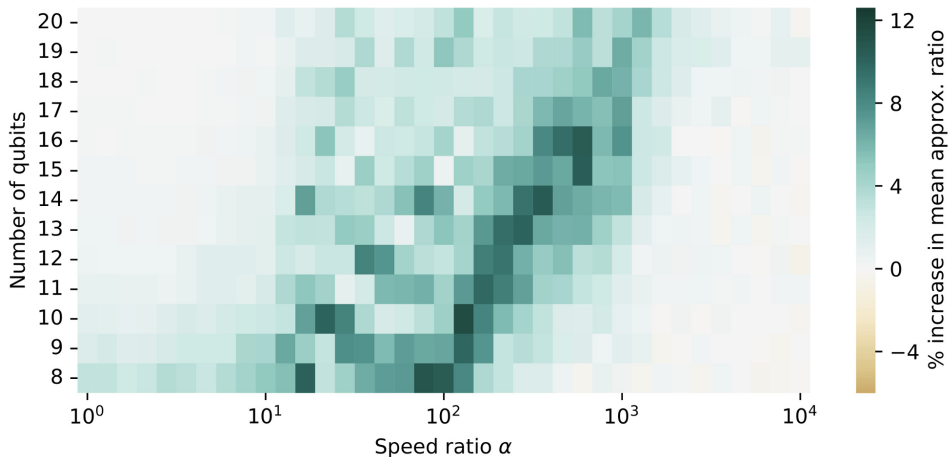


Figure 5.4: **Percentage difference in mean approximation ratio attained by bDA-QAOA at parameters maximizing error-free QAOA and bDA-QAOA with optimized parameters.** We have computed it over 50 randomly generated MAX-CUT problems with a constant filling factor $p_{\text{clause}} = 0.7$. On the x -axis, the ratio α of single-qubit to problem Hamiltonian term strength is displayed, with error-free QAOA existing in the limit as this ratio approaches infinity. Darker colors imply that for the concerned speed ratio and qubit number, the variational nature of QAOA can account for differences between bDA-QAOA and the ideal algorithm. The plot highlights the advantage of using a variational algorithm like QAOA compared to a non-variational algorithm in the presence of coherent errors in the DAQC framework.

where

$$\langle H_P \rangle_{\vec{\beta}, \vec{\gamma}}^{\alpha, \text{DA}} = \langle \vec{\beta}, \vec{\gamma} |^{\alpha, \text{DA}} H_P | \vec{\beta}, \vec{\gamma} \rangle^{\alpha, \text{DA}}. \quad (5.11)$$

In fact, Fig. 5.4 suggests that this is not the case and shows that a significant increase in the success probability of QAOA results from the variational freedom of the algorithm. Figure 5.4 should be understood to demonstrate the parameter regimes for which it makes more sense to perform a variational algorithm such as QAOA rather than a fixed gate sequence algorithm such as the quantum Fourier transform. For high α , the error introduced by the scheme is negligible, and both variational algorithms and fixed sequence algorithms will perform similarly. In the middle of the plot, a dark turquoise band can be seen implying that, while a non-variational algorithm will have low fidelity due to the presence of DA-induced coherent errors, the variational algorithm still functions. For sufficiently low α , we enter a regime in which even the variational algorithm fails to recover any performance through altering parameters. We interpret that this lack of ability of DA-QAOA to absorb error in the low α regime is a manifestation of barren plateaus in the objective function [111]. Barren plateaus are a feature discovered to occur in the optimization landscapes of quantum neural networks. When parameterized random circuits are used as ansatz

in variational algorithms, the gradient of the objective function with respect to the variational parameters is observed to become exponentially small in the number of qubits used. When α reduces to a certain value, we interpret that the DA-QAOA ansatz loses specificity to the problem Hamiltonian of interest. The variational form used for optimization no longer bears similarity to the objective function used and is, consequently, no better an ansatz than a random quantum circuit. At this point of low α , we observe that the gradient of our objective function with respect to the variational parameters $\vec{\gamma}, \vec{\beta}$ tends to become prohibitively small and the approximation ratio attained therefore varies little with differing parameters.

5.3.3 Sensitivity to other errors

Next to the errors discussed in the previous sections that are imminent to the hardware simplification provided by our digital-analog approach, the algorithm is exposed to other sources of errors common to NISQ computing. As detailed error budgets of concrete hardware are currently hard to determine, we would like to qualitatively evaluate their impact on our technique.

On the one hand, single-qubit gate errors induced by decoherence measured by $T_{1/2}$ will have a full impact on this algorithm as these are repeatedly executed. Small errors in the rotation axis will also have a full impact as they can be mistaken for a modified problem Hamiltonian. Errors in the rotation angle can be expected to be less critical as some of them can be accommodated in the classical optimization process. So all in all, single-qubit errors have the same if somewhat smaller impact than in a compiled gate model QAOA.

Two-qubit gates do not appear directly in our scheme thus avoiding two-qubit gate control errors as well as the additional entry points for noise through fast two-qubit control ports. However, the interaction mediated by the problem Hamiltonian can still create entangled states, which decay faster than non-entangled states. Notably, an n -qubit GHZ state dephases in a time T_2/n [112]. The precise degree of entanglement needed for a specific problem instance is currently unknown for any quantum optimization algorithms. Yet, we can summarize that the sensitivity of digital-analog QAOA to two-qubit errors is lower than the compiled version. Given a single qubit error rate, alongside the total execution time of the algorithm relative to T_2 , the depth at which this algorithm can be faithfully executed could be inferred.

In this estimate, we need to keep in mind whether coherent over-rotation errors have an effect different from incoherent errors. This case could occur if they interfered in a structured way. Given the randomization effect of the problem Hamiltonian to any state, this is unlikely and we expect that their impact is faithfully represented by the measured fidelity.

In this chapter, we have highlighted the advantages of applying the DAQC paradigm for performing the QAOA algorithm. Our findings show that the DAQC setting allows for efficient implementation of the problem Hamiltonian, with minimal

overhead for swapping resources. Additionally, we have shown that QAOA in the digital-analog setting is more resilient to errors compared to pre-programmed algorithms, making it a promising approach for near-term quantum devices designed for combinatorial optimization. This research offers a new avenue for the development of NISQ-era devices and brings us closer to realizing a quantum advantage for solving real-world problems.

6 Digital-analog simulations using the cross-resonance effect

AT the moment, superconducting circuits have been established as a leading quantum platform in terms of controllability and scalability, mainly caused by the introduction of the transmon qubit [113]. Implementations controlled by microwave pulses have achieved very low errors on single-qubit gates [114], and the most common two-qubit gate for fixed frequency transmons is based on the cross-resonance (CR) interaction [115, 116, 117]. In this chapter, we explore the potential use of the CR effect as an interaction Hamiltonian between transmon qubits, that would allow us to implement the DAQC paradigm in superconducting circuit quantum processors.

In section 6.1, we derive the effective interaction Hamiltonian of the CR effect. First, in section 6.1.1, we consider a CR gate interaction between two superconducting qubits in order to obtain a purely non-local, in a particular frame, effective interaction Hamiltonian. Further, in section 6.1.2, we consider a multi-qubit extension and derive the generalized effective multi-qubit two-local Hamiltonian. Next, in section 6.2, we consider how the multi-qubit Hamiltonian may be toggled into a variety of forms using digital single-qubit gates. Employing this set of Hamiltonians, we design DAQC protocols to simulate Ising (section 6.3.1), $\sigma_x\sigma_y$ (section 6.3.2), and Heisenberg (section 6.3.3) spin models. The resulting DAQC sequences are in some cases Trotter-error-free in 1D. We compute the Trotter error when it is present and

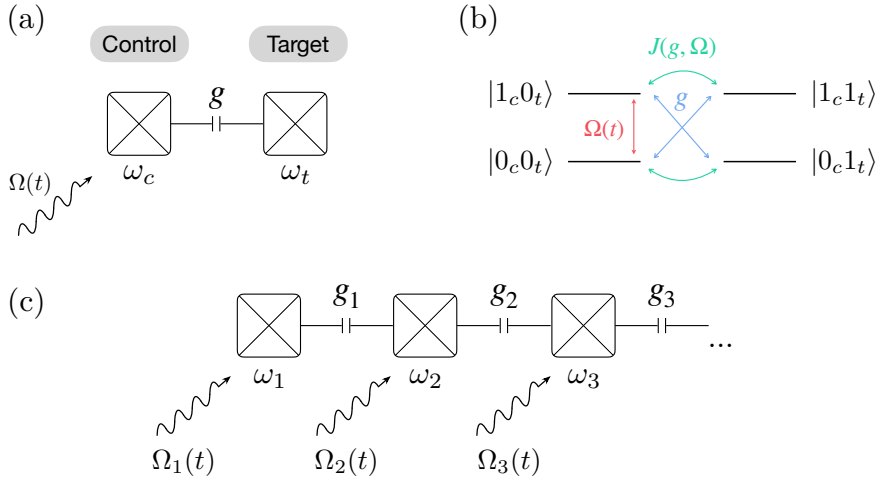


Figure 6.1: **Graphical representation of the cross-resonance effect:** (a) Two qubits, the first one being the control qubit with resonance frequency ω_c and the second one the target qubit with resonance frequency ω_t , are interacting with strength g . The control qubit is driven at the resonance frequency of the target qubit, with driving amplitude $\Omega(t)$. (b) State space representation of the transitions between levels of the control and target qubits, in the presence of a driving of amplitude $\Omega(t)$ on the control qubit. The effective CR interaction is described by strength $J(g, \Omega)$. (c) N qubits with nearest neighbor interaction, all of them are driven at the resonance frequency of their neighbor to the right, illustrating the scenario we describe in section 6.1.2

find that it is reduced by a constant factor with respect to a DQC decomposition of the same model. In section 6.4, we discuss some practical implementations of our findings, and in section 6.5, we conclude the chapter with a discussion and a summary of the extensive work that we present. The content of this chapter is based on the work developed in [56].

6.1 Deriving the effective cross-resonance Hamiltonian

In this section, we present the effective CR Hamiltonian, following the derivation introduced in Ref. [116]. We first introduce the two-qubit scenario, in order to develop an intuition for the effective coupling, and then generalize the results to the case of N qubits. Further details of the calculations, supporting the main text, can be found in Appendix B.1. Note that, in this manuscript, we are working with $\hbar = 1$.

6.1.1 Two qubits

Our starting point is the laboratory frame Hamiltonian, written as

$$\begin{aligned}
 H_{\text{LAB}} = & \frac{1}{2}(\omega_1^g \sigma_z^{(1)} + \omega_2^g \sigma_z^{(2)}) + \Omega_1 \sigma_x^{(1)} \cos(\omega_1 t + \phi_1) \\
 & + \Omega_2 \sigma_x^{(2)} \cos(\omega_2 t + \phi_2) + \frac{g}{2} \sigma_x^{(1)} \sigma_x^{(2)}, \tag{6.1}
 \end{aligned}$$

where $\sigma_x^{(i)}, \sigma_y^{(i)}, \sigma_z^{(i)}$ are the Pauli matrices supported on site i , and ω_k^g and ω_k are the resonance and the driving frequencies of qubit k , respectively. Ω_k represents the amplitude of the driving field, while g denotes the strength of the interaction between the qubits.

The effective Hamiltonian is derived by applying a series of unitary transformations — described in detail in Appendix B.1.1 — to Eq. 6.1. First, we apply a double rotation into the frame co-rotating at the driving frequency of the qubits (ω_1, ω_2). After this, we apply the rotating wave approximation (RWA), valid for $\omega_1, \omega_2 \gg \delta_i = \omega_1^g - \omega_1, \Omega_i, g$, to drop fast terms rotating with frequency $\pm 2\omega_1, \pm 2\omega_2, \pm(\omega_1 + \omega_2)$. We then proceed by applying two new rotations in order to express the Hamiltonian in a more convenient frame, named the quad frame (QF). In this frame, all local terms are eliminated and the result is a purely two-local Hamiltonian. The next step is to consider the case in which we drive the first qubit at the resonance frequency of the second qubit, $\omega_1 = \omega_2^g$, while the second one is not driven, as can be seen in Fig. 6.1(a). After a final RWA, valid for $\Omega_1 \gg g$ or $\delta \gg g$, we end up with the effective Hamiltonian

$$H_{\text{QF}} = \frac{g\Omega_1}{4\delta} (\cos \phi_1 \sigma_x^{(1)} \sigma_x^{(2)} + \sin \phi_1 \sigma_x^{(1)} \sigma_y^{(2)}). \tag{6.2}$$

As ϕ_1 is a controllable phase, we can set $\phi_1 = 0$, resulting in

$$H_{\text{QF}} = \frac{g\Omega_1}{4\delta} \sigma_x^{(1)} \sigma_x^{(2)}. \tag{6.3}$$

6.1.2 N qubits

The N -qubit Hamiltonian, in the laboratory frame, is given by

$$H_{\text{LAB}} = \sum_{k=1}^N \left[\frac{\omega_k^g}{2} \sigma_z^{(k)} + \Omega_k \sigma_x^{(k)} \cos(\omega_k t + \phi_k) \right] + \sum_{k=1}^{N-1} \frac{g_k}{2} \sigma_x^{(k)} \sigma_x^{(k+1)} \tag{6.4}$$

We proceed by moving to the QF by means of appropriate rotations (see Appendix B.1.2 for details). The driving field is then applied to all qubits at the resonance frequency of their neighbor to the right, as shown in Fig. 6.1(c), except for case of open boundary conditions in which the last qubit is not driven. Similar to the two-qubit case, the frame transformations re-express the Hamiltonian in a purely

two-local form. Keeping only terms linear in Ω_i/δ_i , and neglecting fast oscillating terms $\delta \gg g$ by RWA, we arrive at the effective Hamiltonian

$$H_{\text{QF}} = \sum_{k=1}^{N-1} \frac{g_k \Omega_k}{4\delta_k} \sigma_x^{(k)} \left(\sigma_y^{(k+1)} \sin(\phi_k - \phi_{k+1}) - \sigma_z^{(k+1)} \cos(\phi_k - \phi_{k+1}) \right). \quad (6.5)$$

Once again, we have the freedom to set $\phi_k = \phi$ for all k . The Hamiltonian then reduces to

$$H_{\text{QF}} = \sum_{k=1}^{N-1} J_k \sigma_x^{(k)} \sigma_z^{(k+1)}, \quad (6.6)$$

where we have defined $J_k = -g_k \Omega_k / 4\delta_k$. As seen in the two-qubit case, the Hamiltonian only contains two-qubit interaction terms. In the next sections, we will discuss the use of this Hamiltonian to generate the analog dynamics of a DA computation.

6.2 Digital-analog dynamics from the CR Hamiltonian

We take Eq. 6.6 as a starting point, and consider $\Omega_k = \Omega$, $\delta_k = \delta$, $g_k = g$, $J_k = J$, for simplicity. Then, we write the effective Hamiltonian in the QF as

$$H_A = J \sum_{k=1}^{N-1} \sigma_x^{(k)} \sigma_z^{(k+1)}. \quad (6.7)$$

6.2.1 Synthesis Error

Given that the effective Hamiltonian is the centerpiece of the simulation protocols, we need to estimate the synthesis error associated with the fact that it is an approximation of the original Hamiltonian. In the weak-driving regime $\Omega_k \ll \delta_k$, the original Hamiltonian without the QF RWA is

$$\begin{aligned} H^{\text{org}} = & \frac{g}{4} \sum_{k=1}^{N-1} \left\{ \left(\sigma_z^{(k)} \sigma_z^{(k+1)} + \sigma_y^{(k)} \sigma_y^{(k+1)} \right) \cos \delta t \right. \\ & + \left(\sigma_y^{(k)} \sigma_z^{(k+1)} - \sigma_z^{(k)} \sigma_y^{(k+1)} \right) \sin \delta t \\ & \left. - \frac{\Omega}{\delta} \left[\sigma_x^{(k)} \sigma_z^{(k+1)} + \left(\sigma_z^{(k)} \cos 2\delta t + \sigma_y^{(k)} \sin 2\delta t \right) \sigma_x^{(k+1)} \right] \right\}. \quad (6.8) \end{aligned}$$

In order to compute the synthesis error, we focus on the Frobenius norm,

$$\|A\|_F = \sqrt{\text{tr}(A^\dagger A)}, \quad (6.9)$$

which provides an upper bound for the spectral norm. Let us compute the norm for the difference between the two Hamiltonians, $\Delta H = H^{\text{org}} - H_A$,

$$\begin{aligned} \Delta H &= \frac{g}{4} \sum_{k=1}^{N-1} \left\{ \left(\sigma_z^{(k)} \sigma_z^{(k+1)} + \sigma_y^{(k)} \sigma_y^{(k+1)} \right) \cos \delta t \right. \\ &\quad + \left(\sigma_y^{(k)} \sigma_z^{(k+1)} - \sigma_z^{(k)} \sigma_y^{(k+1)} \right) \sin \delta t \\ &\quad \left. - \frac{\Omega}{\delta} \left(\sigma_z^{(k)} \cos 2\delta t + \sigma_y^{(k)} \sin 2\delta t \right) \sigma_x^{(k+1)} \right\}. \end{aligned} \quad (6.10)$$

The latter part of this operator contributes with Ω^2/δ^2 to the Frobenius norm, so we will neglect that part in the approximation $\Omega/\delta \ll 1$. The rest can be written as

$$\begin{aligned} \Delta H &= \frac{g}{4} \sum_{k=1}^{N-1} \left\{ \left(\sigma_z^{(k)} \cos \delta t + \sigma_y^{(k)} \sin \delta t \right) \sigma_z^{(k+1)} \right. \\ &\quad \left. + \left(\sigma_y^{(k)} \cos \delta t - \sigma_z^{(k)} \sin \delta t \right) \sigma_y^{(k+1)} \right\}, \end{aligned} \quad (6.11)$$

which corresponds to the result of a rotation given by $U_k = e^{-i\delta t \sigma_x^{(k)}/2}$. This norm can be computed analytically by rewriting the last expression as

$$\Delta H = \frac{g}{4} \sum_{k=1}^{N-1} U_k^\dagger \left(\sigma_z^{(k)} \sigma_z^{(k+1)} + \sigma_y^{(k)} \sigma_y^{(k+1)} \right) U_k. \quad (6.12)$$

Then, we see that the only terms that survive the trace of $(\Delta H)^\dagger \Delta H$, which is computed as

$$\frac{g^2}{16} \sum_{k,k'=1}^{N-1} U_k^\dagger \left(\sigma_z^{(k)} \sigma_z^{(k+1)} + \sigma_y^{(k)} \sigma_y^{(k+1)} \right) U_k U_{k'}^\dagger \left(\sigma_z^{(k')} \sigma_z^{(k'+1)} + \sigma_y^{(k')} \sigma_y^{(k'+1)} \right) U_{k'}, \quad (6.13)$$

are those which satisfy $k = k'$. Consequently, we obtain

$$\text{tr} [(\Delta H)^\dagger \Delta H] = \frac{g^2}{16} \text{tr} \left(2 \sum_{k=1}^{N-1} \mathbb{I} \right) = \frac{g^2}{8} (N-1) \text{tr}(\mathbb{I}), \quad (6.14)$$

where \mathbb{I} actually represents $\bigotimes_{k=1}^N \mathbb{I}_k$. We want to set the normalization to $\text{tr}(\mathbb{I}) = 1$, which corresponds to a factor of $2^{-N/2}$ on the Frobenius norm, since

$$\left\| \bigotimes_{k=1}^N \mathbb{I}_k \right\|_F = 2^{N/2}. \quad (6.15)$$

Then, we find the Frobenius norm for N qubits ($N \geq 2$) to be

$$\|\Delta H\|_F = \frac{g}{2\sqrt{2}} \sqrt{N-1}. \quad (6.16)$$

See that this norm diverges with the square root of the number of qubits. Notice however that the Frobenius norm per qubit decreases with N . Furthermore, we have computed the norm of the difference between the propagators, $\Delta\mathcal{U} = \mathcal{U}^{\text{org}} - \mathcal{U}_A$,

$$\|\Delta\mathcal{U}\|_F = \frac{g}{\delta\sqrt{2}} \left| \sin \frac{\delta t}{2} \right| \sqrt{N-1}. \quad (6.17)$$

Here, the propagators are computed up to the first order in the Dyson series. Again, the norm of the difference of propagators per qubit decreases with N . Note that, for $\delta t \ll 1$,

$$\|\Delta\mathcal{U}\|_F \approx t \cdot \|\Delta H\|_F. \quad (6.18)$$

The synthesis errors corresponding to the Hamiltonians derived in further sections can be found in Appendix B.2.

6.2.2 Hamiltonian toggling

Let us now consider DA quantum simulations of the spin-1/2 Ising, XY , and Heisenberg models in 1 and 2 dimensions. We designate the effective Hamiltonian in the QF, given in Eq. 6.6, as our fundamental DAQC Hamiltonian from which all others will be generated. By rotating to the reference frame in which the Hadamard transformation is applied to all *even* qubits, i.e. $V_{\text{H}}^{\text{even}} = \bigotimes_i H^{(2i)}$, the Hamiltonian transforms into

$$H^{\text{even}} = J \sum_{k=1}^{\frac{N}{2}} \sigma_x^{(2k-1)} \sigma_x^{(2k)} + J \sum_{k=1}^{\frac{N-1}{2}} \sigma_z^{(2k)} \sigma_z^{(2k+1)}. \quad (6.19)$$

From this reference frame, Hadamard transforming all qubits will toggle the Hamiltonian into its odd form, i.e. translating the Hamiltonian by one site,

$$H^{\text{odd}} = J \sum_{k=1}^{\frac{N}{2}} \sigma_z^{(2k-1)} \sigma_z^{(2k)} + J \sum_{k=1}^{\frac{N-1}{2}} \sigma_x^{(2k)} \sigma_x^{(2k+1)}. \quad (6.20)$$

6.2.3 Two-dimensional generalization

Let us also consider the extension of the Hamiltonian to two dimensions. Consider a single target qubit in a two-dimensional lattice that is driven at the frequencies of its neighbors in the $+\hat{i}$ and $+\hat{j}$ directions. This realizes a $\sigma_x^{(c)} \sigma_z^{(t)}$ -interaction between the control qubit located at (i, j) and target qubits at sites $(i+1, j)$ and $(i, j+1)$.

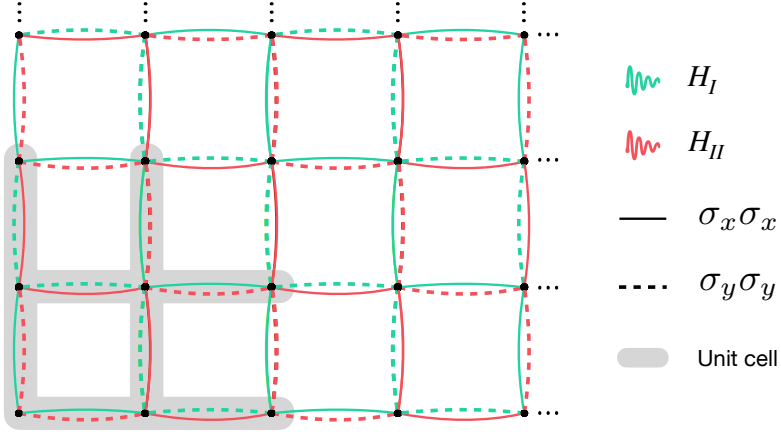


Figure 6.2: **Illustration of analog Hamiltonian interactions on a 2-dimensional lattice.** The green (red) lattice represents the Hamiltonian given by Eq. 6.22 (6.23). Vertices correspond to qubits in a 2D lattice and the solid and dashed edges correspond to the $\sigma_x \sigma_x$ and $\sigma_y \sigma_y$ interactions, respectively.

The extension of H^{odd} in Eq. 6.20 is

$$\begin{aligned}
 H_{2D}^{\text{odd}} = & J \left[\sum_{i,j=1}^{\frac{N}{2}} \sigma_z^{(2i-1,2j-1)} \left(\sigma_z^{(2i-1,2j)} + \sigma_z^{(2i,2j-1)} \right) \right. \\
 & + \sum_{i,j=1}^{\frac{N-1}{2}} \sigma_z^{(2i,2j)} \left(\sigma_z^{(2i,2j+1)} + \sigma_z^{(2i+1,2j)} \right) \\
 & + \sum_{i=1}^{\frac{N}{2}} \sum_{j=1}^{\frac{N-1}{2}} \sigma_x^{(2i-1,2j)} \left(\sigma_x^{(2i-1,2j+1)} + \sigma_x^{(2i,2j)} \right) \\
 & \left. + \sum_{i=1}^{\frac{N-1}{2}} \sum_{j=1}^{\frac{N}{2}} \sigma_x^{(2i,2j-1)} \left(\sigma_x^{(2i,2j)} + \sigma_x^{(2i+1,2j-1)} \right) \right], \quad (6.21)
 \end{aligned}$$

where summations run over repetitions of the unit cell illustrated in Fig. 6.2. Likewise, the extension of H^{even} in Eq. 6.19 is $H_{2D}^{\text{even}} = H_{2D}^{\text{odd}}(x \leftrightarrow z)$, which is easily realized by applying a Hadamard on every site of the lattice. Applying a global

$R_x(\pi/2) = e^{-i\pi\sigma_x/4}$ transformation on Hamiltonian H_{2D}^{even} , we obtain

$$\begin{aligned}
 H_I &= J \sum_{i,j=1}^{\frac{N}{2}} \left(\sigma_x^{(2i-1,2j-1)} \sigma_x^{(2i,2j-1)} + \sigma_y^{(2i,2j-1)} \sigma_y^{(2i+1,2j-1)} \right. \\
 &\quad + \sigma_y^{(2i-1,2j)} \sigma_y^{(2i,2j)} + \sigma_x^{(2i,2j)} \sigma_x^{(2i+1,2j)} \\
 &\quad + \sigma_x^{(2i-1,2j-1)} \sigma_x^{(2i-1,2j)} + \sigma_y^{(2i,2j-1)} \sigma_y^{(2i,2j)} \\
 &\quad \left. + \sigma_y^{(2i-1,2j)} \sigma_y^{(2i-1,2j+1)} + \sigma_x^{(2i,2j)} \sigma_x^{(2i,2j+1)} \right), \tag{6.22}
 \end{aligned}$$

where we have simplified the summation limits by considering that the Hamiltonian acts on a system with periodic boundary conditions. If we rotate H_{2D}^{odd} by $R_x(\pi/2)$, we have

$$\begin{aligned}
 H_{II} &= J \sum_{i,j=1}^{\frac{N}{2}} \left(\sigma_y^{(2i-1,2j-1)} \sigma_y^{(2i,2j-1)} + \sigma_x^{(2i,2j-1)} \sigma_x^{(2i+1,2j-1)} \right. \\
 &\quad + \sigma_x^{(2i-1,2j)} \sigma_x^{(2i,2j)} + \sigma_y^{(2i,2j)} \sigma_y^{(2i+1,2j)} \\
 &\quad + \sigma_y^{(2i-1,2j-1)} \sigma_y^{(2i-1,2j)} + \sigma_x^{(2i,2j-1)} \sigma_x^{(2i,2j)} \\
 &\quad \left. + \sigma_x^{(2i-1,2j)} \sigma_x^{(2i-1,2j+1)} + \sigma_y^{(2i,2j)} \sigma_y^{(2i,2j+1)} \right). \tag{6.23}
 \end{aligned}$$

Note that H_{II} is just a translation of H_I by the vector $(1,1)$. The interactions described by these Hamiltonians are represented in Fig. 6.2, where H_I 's and H_{II} 's interactions are illustrated by the green and red edges, respectively. In both cases, the solid (dashed) edges correspond to $\sigma_x\sigma_x$ ($\sigma_y\sigma_y$) interactions between adjacent qubits, and the summations in Eqs. 6.22, and 6.23 correspond to a tiling of the 2D lattice using the unit cell, highlighted in blue in Fig. 6.2.

6.3 Many-body compilation

We now discuss how to simulate a variety of paradigmatic spin models with the Hamiltonians discussed above.

6.3.1 Ising model

So far we have considered a multi-qubit framework in which we drive all qubits at the resonance frequency of their neighbors to the right. For this particular case, let us now explore a scenario in which we drive only odd or even qubits, which can be

achieved by tuning the system's parameters in the following way:

$$\begin{aligned}
 k \text{ control} &\rightarrow \left\{ \omega_k = \omega_{k+1}^q, \varphi_k(t) = \delta_{k+1}t + \phi_k - \phi_{k+1}, \right. \\
 &\quad \left. \eta_k \approx \delta_k, \sin \xi_k \approx 1, \cos \xi_k \approx \frac{\Omega_k}{\delta_k} \right\}, \\
 k \text{ target} &\rightarrow \left\{ \varphi_k(t) = (\omega_k - \omega_{k+1})t - \phi_{k+1}, \Omega_k = 0, \delta_k = 0, \right. \\
 &\quad \left. \eta_k = 0, \phi_k = 0, \sin \xi_k = 0, \cos \xi_k = 1 \right\},
 \end{aligned} \tag{6.24}$$

where the qubit we drive is the control qubit and its neighbor to the right is the corresponding target qubit. Assuming we drive only odd qubits, the choice of parameters leads to a particular QF transformation, represented by

$$V_{\text{QF}}^{\text{odd}} = \bigotimes_{k \text{ odd}} V_{\text{QF}}^{(k)} V_I^{(k+1)}, \tag{6.25}$$

where $V_{\text{QF}}^{(k)}$ is the QF transformation applied on qubit k (this transformation is discussed in Appendix B.3), and $V_I^{(k+1)} = e^{-\frac{it}{2}\omega_{k+1}^q\sigma_z^{(k+1)}}$ is the transformation to the interaction picture of qubit $k+1$. After applying a RWA, by keeping the static terms, we write the Hamiltonian in the QF as

$$H_{\text{QF}}^{\text{odd}} = J \sum_{k=1}^{\frac{N}{2}} \sigma_x^{(2k-1)} \left(\sigma_x^{(2k)} \cos \phi + \sigma_y^{(2k)} \sin \phi \right), \tag{6.26}$$

after setting $\delta_{2k-1} = \delta$, $\Omega_{2k-1} = \Omega$, $g_{2k-1} = g$, $\phi_{2k-1} = \phi$, and defining $J = g\Omega/4\delta$. See that this is a straightforward multi-qubit extension of the Hamiltonian in Eq. 6.2. If we do the same, in the case in which we drive only even qubits, the transformation is

$$V_{\text{QF}}^{\text{even}} = \bigotimes_{k \text{ even}} V_{\text{QF}}^{(k)} V_I^{(k+1)}, \tag{6.27}$$

and we obtain

$$H_{\text{QF}}^{\text{even}} = J \sum_{k=1}^{\frac{N-1}{2}} \sigma_x^{(2k)} \left(\sigma_x^{(2k+1)} \cos \phi + \sigma_y^{(2k+1)} \sin \phi \right). \tag{6.28}$$

Considering $\phi = 0$, these Hamiltonians transform into

$$H_{\text{QF}}^{\text{odd}} = J \sum_{k=1}^{\frac{N}{2}} \sigma_x^{(2k-1)} \sigma_x^{(2k)}, \quad \text{and} \quad H_{\text{QF}}^{\text{even}} = J \sum_{k=1}^{\frac{N-1}{2}} \sigma_x^{(2k)} \sigma_x^{(2k+1)}, \tag{6.29}$$

and we see that $[H_{\text{QF}}^{\text{odd}}, H_{\text{QF}}^{\text{even}}] = 0$. If we rotate all qubits by a Hadamard gate, we obtain

$$\begin{aligned} V_{\text{H}}^{\dagger} H_{\text{QF}}^{\text{odd}} V_{\text{H}} &= J \sum_{k=1}^{\frac{N}{2}} \sigma_z^{(2k-1)} \sigma_z^{(2k)} \equiv H_1, \\ V_{\text{H}}^{\dagger} H_{\text{QF}}^{\text{even}} V_{\text{H}} &= J \sum_{k=1}^{\frac{N-1}{2}} \sigma_z^{(2k)} \sigma_z^{(2k+1)} \equiv H_2, \end{aligned} \quad (6.30)$$

which leads to

$$H_{ZZ} = H_1 + H_2 = J \sum_{k=1}^{N-1} \sigma_z^{(k)} \sigma_z^{(k+1)}. \quad (6.31)$$

This sequence for simulating the evolution of H_{ZZ} can be interpreted as the combination of two blocks: the first one represents the evolution given by $\mathcal{U}_1(t) = e^{-iH_1 t}$, where we only drive odd qubits, and the second one represents the evolution given by $\mathcal{U}_2(t) = e^{-iH_2 t}$, where we only drive even qubits, both in a frame rotated by Hadamard gates. The integrity of these simulation blocks relies on the fact that $[H_1, H_2] = 0$, meaning that the pairwise combination of propagators is exact. Then, the propagators corresponding to the two blocks can exactly describe the evolution of the whole,

$$\mathcal{U}_{ZZ}(t) = e^{-iH_{ZZ}t} = e^{-i(H_1+H_2)t} = \mathcal{U}_1(t)\mathcal{U}_2(t). \quad (6.32)$$

The propagator corresponding to H_{ZZ} is computed as

$$\mathcal{U}_{ZZ}(t)|\psi\rangle = \mathcal{U}_1(t)\mathcal{U}_2(t)|\psi\rangle = V_{\text{H}}^{\dagger} \mathcal{U}_{\text{QF}}^{\text{odd}}(t) V_{\text{H}} V_{\text{H}}^{\dagger} \mathcal{U}_{\text{QF}}^{\text{even}}(t) V_{\text{H}} |\psi\rangle \quad (6.33)$$

where $\mathcal{U}_{\text{QF}}^{\text{odd}}(t)$ and $\mathcal{U}_{\text{QF}}^{\text{even}}(t)$ are the propagators generated by $H_{\text{QF}}^{\text{odd}}$ and $H_{\text{QF}}^{\text{even}}$, respectively. The former is achieved by rotating all qubits by $V_{\text{QF}}^{\text{odd}}$, and the latter is achieved by rotating all qubits by $V_{\text{QF}}^{\text{even}}$. Furthermore, $V_{\text{H}} = \bigotimes_k \mathbb{H}^{(k)}$, where $\mathbb{H}^{(k)} = e^{i\pi/2} e^{-i\pi\sigma_y^{(k)}/4} e^{-i\pi\sigma_z^{(k)}/2}$ represents the application of a Hadamard gate on qubit k . The simulation protocol comprises the following steps:

1. Prepare an initial product state $\bigotimes_k |\psi_k\rangle$.
2. Apply Hadamard gates, \mathbb{H} , on all qubits.
3. Let the states evolve according to the underlying analog Hamiltonian with analog propagator $\mathcal{U}_{\text{QF}}^{\text{even}}(\tau)$ for time τ .
4. Let the states evolve according to the underlying analog Hamiltonian with analog propagator $\mathcal{U}_{\text{QF}}^{\text{odd}}(\tau)$ for time τ .
5. Apply Hadamard gates, \mathbb{H} , on all qubits.

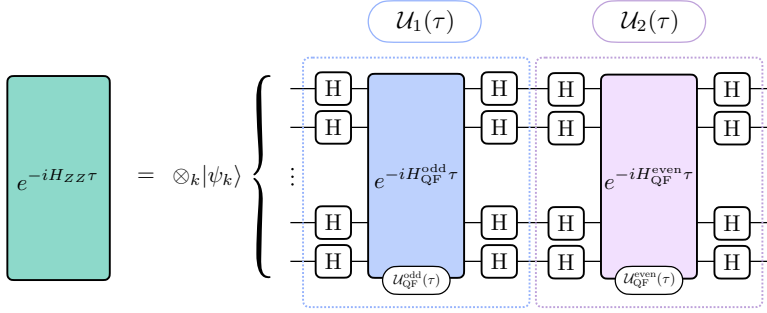


Figure 6.3: **Digital-analog quantum circuit to simulate the evolution under Hamiltonian H_{ZZ} for a time τ .** This simulation is carried out by transforming all qubits by $V_{\text{QF}}^{\text{even}}$, which entails transforming even qubits to the QF (V_{QF}) and odd qubits to the interaction picture (V_I), in a setup in which only even qubits are being driven. In this scenario, the analog propagator $\mathcal{U}_{\text{QF}}^{\text{even}}$ – which describes the evolution under analog Hamiltonian $H_{\text{QF}}^{\text{even}}$ – is then conjugated by Hadamard gates (H) on all qubits. This segment of the circuit simulates the evolution given by $\mathcal{U}_2(\tau)$. The circuit is repeated with the QF transformation being applied to odd qubits and the interaction picture transformation to even qubits ($V_{\text{QF}}^{\text{odd}}$), while only odd qubits are driven. This segment simulates the evolution given by $\mathcal{U}_1(\tau)$.

Note that operations that consist of the application of a unitary U , followed by their inverse U^\dagger , render the identity as the result, and thus are not mentioned in the simulation protocol steps. However, these operations are included in the figures for illustrative purposes. Due to the idiosyncrasies of the Hamiltonians derived in this protocol, we benefit from the absence of Trotter error, which implies no limits on application time, τ , of the block. This block is represented in Fig. 6.3. To evolve a state $|\psi\rangle$ with Hamiltonian $H_{\sigma_z\sigma_z}$, one must re-apply the block M times where the total evolution time is $T = M\tau$. Gate-based quantum circuits describe the application of quantum gates following the usual flow of time. That is, from left to right, following the order in which the operators are applied on a quantum state represented by a ket.

6.3.2 XY model

Let us now describe a protocol to simulate a XY model in which all adjacent spins interact by $\sigma_x\sigma_x + \sigma_y\sigma_y$ terms.

1D Simulation

In the 1D case, we start from the Hamiltonians in Eqs. 6.19, and 6.20. By performing the same $R_x(\pi/2) \equiv R$ about each qubit, we find

$$\begin{aligned} H^{\text{odd}'} &= R^\dagger H^{\text{odd}} R = J \sum_{k=1}^{\frac{N}{2}} \sigma_y^{(2k-1)} \sigma_y^{(2k)} + J \sum_{k=1}^{\frac{N-1}{2}} \sigma_x^{(2k)} \sigma_x^{(2k+1)}, \\ H^{\text{even}'} &= R^\dagger H^{\text{even}} R = J \sum_{k=1}^{\frac{N}{2}} \sigma_x^{(2k-1)} \sigma_x^{(2k)} + J \sum_{k=1}^{\frac{N-1}{2}} \sigma_y^{(2k)} \sigma_y^{(2k+1)}, \end{aligned} \quad (6.34)$$

which, upon summing, realize the 1D XY chain Hamiltonian

$$H_{XY} = H^{\text{even}'} + H^{\text{odd}'} = J \sum_{k=1}^{N-1} \left(\sigma_x^{(k)} \sigma_x^{(k+1)} + \sigma_y^{(k)} \sigma_y^{(k+1)} \right). \quad (6.35)$$

The key to this protocol is that $[H^{\text{even}'}, H^{\text{odd}'}] = R^\dagger [H^{\text{even}}, H^{\text{odd}}] R = 0$, which implies

$$\mathcal{U}_{XY}(t) = e^{-iH_{XY}t} = e^{-i(H^{\text{odd}'} + H^{\text{even}'})t} = \mathcal{U}^{\text{odd}'}(t) \mathcal{U}^{\text{even}'}(t). \quad (6.36)$$

This allows us to decompose the total XY propagator into the product of two toggled Hamiltonians which results in a Trotter-error-free dynamics simulation protocol. The propagator $\mathcal{U}_{XY}(t)$ is further decomposed as

$$\mathcal{U}_{XY}(t)|\psi\rangle = \mathcal{U}^{\text{odd}'} \mathcal{U}^{\text{even}'} |\psi\rangle = W^{\text{odd}\dagger} \mathcal{U}_A(t) W^{\text{odd}} W^{\text{even}\dagger} \mathcal{U}_A(t) W^{\text{even}} |\psi\rangle, \quad (6.37)$$

where $\mathcal{U}_A(t)$ is the original analog propagator generated by H_A , of Eq. 6.7, and

$$W^{\text{odd}} = \bigotimes_{k \text{ odd}} H^{(k)} R^{(k)} R^{(k+1)}, \quad (6.38)$$

$$W^{\text{even}} = \bigotimes_{k \text{ even}} H^{(k)} R^{(k-1)} R^{(k)}, \quad (6.39)$$

where $H^{(k)}$ and $R^{(k)}$ represent the application of a Hadamard gate and a $\pi/2$ x -rotation, respectively, on qubit k . The simulation protocol consists of the following steps:

1. Prepare an initial product state $\otimes_k |\psi_k\rangle$.
2. Apply a x - $\pi/2$ rotation on all qubits with $R_x(\pi/2)$.
3. Apply Hadamard gates, H , on even qubits.
4. Let the states evolve according to the underlying analog Hamiltonian with analog propagator $\mathcal{U}_A(\tau)$ for time τ .
5. Apply Hadamard gates, H , on all qubits.

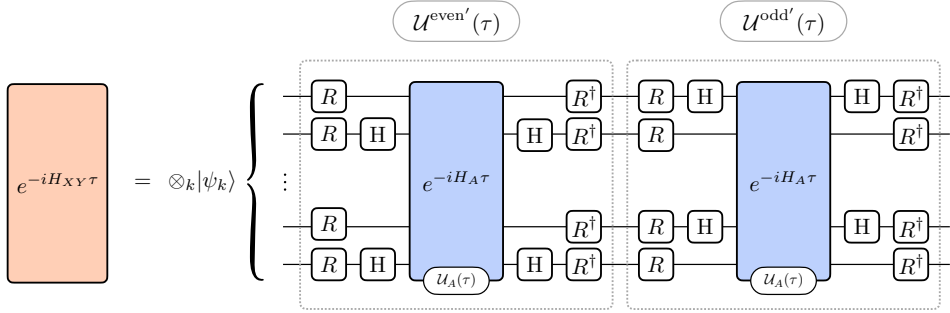


Figure 6.4: **Digital-analog quantum circuit to simulate the evolution of an initial quantum state under Hamiltonian H_{XY} for a time τ .** This simulation is carried out by conjugating the analog propagator \mathcal{U}_A – which describes the evolution under analog Hamiltonian H_A – by Hadamard gates (H) on even qubits, combined with rotations R on all qubits, which are x rotations by $R_x(\pi/2)$. This segment of the circuit simulates the evolution given by $\mathcal{U}^{\text{even}'}$. The circuit is then repeated with Hadamard gates applied to odd qubits, and this segment simulates the evolution given by $\mathcal{U}^{\text{odd}'}$.

6. Let the states evolve according to the underlying analog Hamiltonian with analog propagator $\mathcal{U}_A(\tau)$ for time τ .
7. Apply Hadamard gates, H, on odd qubits.
8. Undo the x - $\pi/2$ rotation on all qubits by $R_x^\dagger(\pi/2)$.

The entire sequence of operations needed to evolve by the XY Hamiltonian is depicted in Fig. 6.4. To evolve for a total time T with Hamiltonian H_{XY} , one must re-apply the block $M = T/\tau$ times. Note that the three layers of single qubit rotations in between evolution by the analog propagators simplify into the product of single-qubit gates, which in this case simplifies to $R^\dagger H R = (x + y)/\sqrt{2}$.

2D Simulation and Digital vs. Digital-Analog Trotter Errors

The same two-Hamiltonian decomposition may be performed in two dimensions, taking the Hamiltonians in Eqs. 6.22, and 6.23, such that $H_{XY}^{2D} = H_I + H_{II}$. However, since $[H_I, H_{II}] \neq 0$ in two dimensions, we must resort to an approximate Trotter decomposition of the 2D XY propagator. If we compute $[H_I, H_{II}]$ we find 16 non-commuting terms.

Let us now compare the errors arising from a first order Trotter decomposition of our target evolution unitary. Overall, our goal is to determine the gate complexity of an approximate product decomposition $\tilde{\mathcal{U}}_{XY}^{2D}$ such that $\|\mathcal{U}_{XY}^{2D}(\tau) - \tilde{\mathcal{U}}_{XY}^{2D}(\tau)\| \leq \epsilon$ for an ϵ of our choosing. Here the target propagator is generated by exponentiating the target Hamiltonian H_{XY}^{2D} , while $\tilde{\mathcal{U}}_{XY}^{2D}$ is generated by a first order Trotter decom-

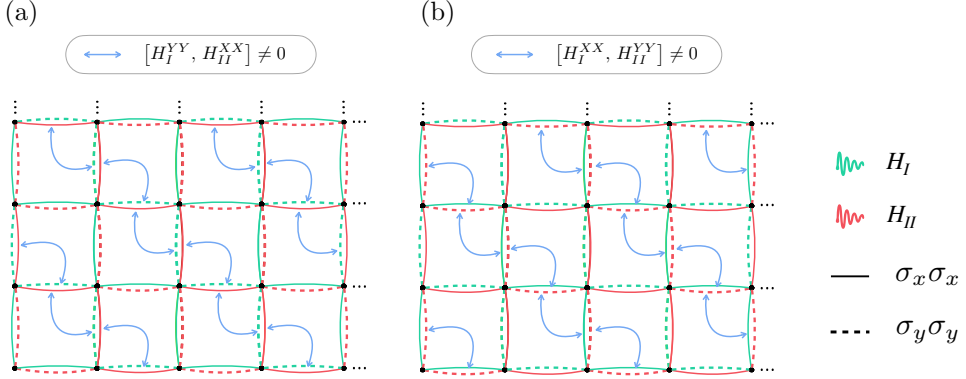


Figure 6.5: **Lattice representation of the interactions featured on H_I (green) and H_{II} (red).** The blue arrows indicate the non-commuting terms between H_I and H_{II} . These Hamiltonians are split into xx and yy - supported operators, $H_i = H_i^{XX} + H_i^{YY}$, in order to identify the two non-commuting operators: **(a)** Non-zero terms of $[H_I^{YY}, H_{II}^{XX}]$, **(b)** Non-zero terms of $[H_I^{XX}, H_{II}^{YY}]$. Jointly, these terms estimate the total Trotter error of the DAQC decomposition.

position which may be implemented through our DAQC Hamiltonians or through a digitized decomposition.

A first order Trotterization approximates an operator exponential of two generally non-commuting operators, α and β , as $e^{\Delta t \alpha} e^{\Delta t \beta} = e^{\Delta t(\alpha + \beta)} + \mathcal{O}(\Delta t^2[\alpha, \beta])$ by discarding the Δt^2 terms in the small Δt regime. This quantity can be made arbitrarily small by breaking up the total evolution time into sufficiently small pieces $\Delta t = \tau/N$. Bounding the error in the DAQC case reduces to computing $\|[H_I, H_{II}]\|$. Breaking down each Hamiltonian into its X and Y components, such that $H_i = H_i^{XX} + H_i^{YY}$, simplifies the commutator norm to $\|[H_I^{YY}, H_{II}^{XX}] + [H_I^{XX}, H_{II}^{YY}]\| = \|A + B\|$. These operators are

$$\begin{aligned}
 A &= J^2 \sum_{i,j,i',j'} \left[\sigma_y^{(2i,2j-1)} \sigma_y^{(2i+1,2j-1)} + \sigma_y^{(2i-1,2j)} \sigma_y^{(2i,2j)} \right. \\
 &\quad + \sigma_y^{(2i,2j-1)} \sigma_y^{(2i,2j)} + \sigma_y^{(2i-1,2j)} \sigma_y^{(2i-1,2j+1)}, \\
 &\quad \sigma_x^{(2i',2j'-1)} \sigma_x^{(2i'+1,2j'-1)} + \sigma_x^{(2i'-1,2j')} \sigma_x^{(2i',2j')} \\
 &\quad \left. + \sigma_x^{(2i',2j'-1)} \sigma_x^{(2i',2j')} + \sigma_x^{(2i'-1,2j')} \sigma_x^{(2i'-1,2j'+1)} \right] \\
 &= -2iJ^2 \sum_{i,j} \left[\sigma_x^{(2i,2j+1)} \sigma_y^{(2i+1,2j)} \sigma_z^{(2i+1,2j+1)} + \sigma_x^{(2i+1,2j-1)} \sigma_y^{(2i,2j)} \sigma_z^{(2i,2j-1)} \right. \\
 &\quad \left. + \sigma_x^{(2i,2j)} \sigma_y^{(2i-1,2j+1)} \sigma_z^{(2i-1,2j)} + \sigma_x^{(2i,2j-1)} \sigma_y^{(2i-1,2j)} \sigma_z^{(2i,2j)} + (y \leftrightarrow x) \right],
 \end{aligned} \tag{6.40}$$

$$\begin{aligned}
 B &= J^2 \sum_{i,j,i',j'} \left[\sigma_x^{(2i-1,2j-1)} \sigma_x^{(2i-1,2j)} + \sigma_x^{(2i-1,2j-1)} \sigma_x^{(2i,2j-1)} \right. \\
 &\quad + \sigma_x^{(2i,2j)} \sigma_x^{(2i,2j+1)} + \sigma_x^{(2i,2j)} \sigma_x^{(2i+1,2j)}, \\
 &\quad \sigma_y^{(2i'-1,2j'-1)} \sigma_y^{(2i'-1,2j')} + \sigma_y^{(2i'-1,2j'-1)} \sigma_y^{(2i',2j'-1)} \\
 &\quad \left. + \sigma_y^{(2i',2j')} \sigma_y^{(2i',2j'+1)} + \sigma_y^{(2i',2j')} \sigma_y^{(2i'+1,2j')} \right] \\
 &= 2iJ^2 \sum_{i,j} \left[\sigma_x^{(2i,2j)} \sigma_y^{(2i-1,2j+1)} \sigma_z^{(2i,2j+1)} + \sigma_x^{(2i+1,2j)} \sigma_y^{(2i,2j+1)} \sigma_z^{(2i,2j)} \right. \\
 &\quad \left. + \sigma_x^{(2i,2j)} \sigma_y^{(2i+1,2j-1)} \sigma_z^{(2i+1,2j)} + \sigma_x^{(2i,2j-1)} \sigma_y^{(2i-1,2j)} \sigma_z^{(2i-1,2j-1)} + (y \leftrightarrow x) \right].
 \end{aligned} \tag{6.41}$$

Alternatively, from visually inspecting supports and Pauli character of the Hamiltonians H_I and H_{II} denoted in red and green in Fig. 6.5, we can see that there are 8 terms per unit cell in A and that there are likewise 8 similar, but differently supported terms in B. Summing over the two sets of terms in the bulk, we obtain

$$\begin{aligned}
 \|[H_I, H_{II}]\| &= \|2iJ^2 \sum_{i,j} (-1)^{i+j} \sigma_z^{(i,j)} [(\sigma_x^{(i-1,j)} \sigma_y^{(i,j-1)} - \sigma_x^{(i,j+1)} \sigma_y^{(i+1,j)}) + (x \leftrightarrow y)]\| \\
 &\leq 2J^2 \sum_{i,j} \|(-1)^{i+j} \sigma_z^{(i,j)} [(\sigma_x^{(i-1,j)} \sigma_y^{(i,j-1)} - \sigma_x^{(i,j+1)} \sigma_y^{(i+1,j)}) + (x \leftrightarrow y)]\| \\
 &= 2J^2 N^2 \|\sigma_z^{(i,j)} [(\sigma_x^{(i-1,j)} \sigma_y^{(i,j-1)} - \sigma_x^{(i,j+1)} \sigma_y^{(i+1,j)}) + (x \leftrightarrow y)]\| \leq 8J^2 N^2,
 \end{aligned}$$

where we have used the triangle inequality on the spectral norms of the operators.

In order to get a better insight into the performance of the DA computation of the two-dimensional XY model, we need to compare the Trotter error of both DQC and DAQC approaches. This error is proportional to the commutator of $[H_I, H_{II}]$ given in Eq. 6.42 in the DAQC case. In the purely digital case, the commutator that we need to compute is $[H_{xx}, H_{yy}]$, where H_{xx} contains all $\sigma_x \sigma_x$ qubit interactions and H_{yy} all the $\sigma_y \sigma_y$ interactions. Independent of the order in which the gates are implemented, the digital error is bounded by

$$\begin{aligned}
 \|[H_{xx}, H_{yy}]\| &= \|J^2 \sum_{i,j} [(\sigma_x^{(i,j)} \sigma_x^{(i+1,j)} + \sigma_x^{(i,j)} \sigma_x^{(i,j+1)}), (x \leftrightarrow y)]\| \\
 &\leq 24J^2 N^2
 \end{aligned} \tag{6.42}$$

where the final factor arises from a product of the factor of two for the N^2 vertical and horizontal edges, a factor of 6 counting all the non-commuting $\sigma_y \sigma_y$ neighbors of each $\sigma_x \sigma_x$ interaction, and a final factor of two arising from the Pauli commutation relations. Alternatively, by analyzing the form of Eqs. 6.40 and 6.41 we note that the A and B components of the commutator can be identified with free fermions hopping along the diagonal loops of the two-dimensional lattice as defined by the blue arrows in Fig. 6.5. Next, we Jordan-Wigner transform to a Majorana representation, take

periodic boundary conditions, and Fourier transform along the loops. As a result, the spectral norm of A and B is tightened from $\mathcal{O}((NJ)^2) \rightarrow \mathcal{O}(J^2)$ which removes the extensive factor. This tighter bound is proved in Appendix B.4. Likewise, we may use similar techniques to decompose the Digital commutator of Eq. 6.42 into a sum of three times as many free fermion Hamiltonians. The resulting ratio of purely DQC to DAQC commutator norms is still a factor of three. In either case, the DAQC protocol improves the Trotter error bound by a constant factor of three. This constant factor speedup can be used to extend the simulation time by the same factor.

6.3.3 Heisenberg model

We now consider the task of simulating the more complex Heisenberg spin model. The Hamiltonian describing the Heisenberg chain in 1 dimension is $H_{\text{Heis}} = \sum_i \mathbf{S}_i \cdot \mathbf{S}_{i+1}$, with $\mathbf{S} = (\sigma_x, \sigma_y, \sigma_z)$. Consider the general Bloch sphere rotation $R_{\hat{n}}(2\theta) = e^{-i\theta(\hat{n} \cdot \vec{\sigma})}$. We can set the angle θ such that this rotation is cyclic; that is, $\theta = \pi/(3\sqrt{3})$ leads to a cyclic permutation $\sigma_x \rightarrow \sigma_z$, $\sigma_y \rightarrow \sigma_x$ and $\sigma_z \rightarrow \sigma_y$. This transformation is realized by

$$R_E \equiv R_{\hat{n}} \left(2 \frac{\pi}{3\sqrt{3}} \right) = e^{-i \frac{\pi}{3\sqrt{3}} (\sigma_x + \sigma_y + \sigma_z)} = \frac{1}{2} [\mathbb{I} - i(\sigma_x + \sigma_y + \sigma_z)] \quad (6.43)$$

which can easily be implemented on individual qubits by the Euler decomposition $R_E = e^{-i\sigma_y \frac{\pi}{4}} e^{-i\sigma_z \frac{\pi}{4}}$. The cyclic nature of this transformation is manifested through the property $R_E^3 = -\mathbb{I}$. Applying this transformation on all qubits of the Hamiltonian in Eq. 6.19 zero, one, and two times, leads to the following Hamiltonians,

$$\begin{aligned} H_E &= H^{\text{even}} = J \sum_{k=1}^{\frac{N}{2}} \sigma_x^{(2k-1)} \sigma_x^{(2k)} + J \sum_{k=1}^{\frac{N-1}{2}} \sigma_z^{(2k)} \sigma_z^{(2k+1)}, \\ H'_E &= R_E^\dagger H^{\text{even}} R_E = J \sum_{k=1}^{\frac{N}{2}} \sigma_z^{(2k-1)} \sigma_z^{(2k)} + J \sum_{k=1}^{\frac{N-1}{2}} \sigma_y^{(2k)} \sigma_y^{(2k+1)}, \\ H''_E &= R_E^{2\dagger} H^{\text{even}} R_E^2 = J \sum_{k=1}^{\frac{N}{2}} \sigma_y^{(2k-1)} \sigma_y^{(2k)} + J \sum_{k=1}^{\frac{N-1}{2}} \sigma_x^{(2k)} \sigma_x^{(2k+1)}. \end{aligned} \quad (6.44)$$

Summing them together, we obtain the Heisenberg Hamiltonian,

$$H_{\text{Heis}} = H_E + H'_E + H''_E = J \sum_{k=1}^{N-1} (\sigma_x^{(k)} \sigma_x^{(k+1)} + \sigma_y^{(k)} \sigma_y^{(k+1)} + \sigma_z^{(k)} \sigma_z^{(k+1)}). \quad (6.45)$$

In this case, the Hamiltonians do not commute with each other, which means that the construction of the propagator will include Trotter error (analyzed below),

$$\begin{aligned} \mathcal{U}_{\text{Heis}}(t) &= e^{-iH_{\text{Heis}}t} = e^{-i(H_E + H'_E + H''_E)t} \\ &= \mathcal{U}_E(t) \mathcal{U}'_E(t) \mathcal{U}''_E(t) + \mathcal{O}(J^2 t^2). \end{aligned} \quad (6.46)$$

The propagator $\mathcal{U}_{\text{Heis}}(t)$ is constructed as

$$\mathcal{U}_{\text{Heis}}(t)|\psi\rangle \approx \mathcal{U}_E(t)\mathcal{U}'_E(t)\mathcal{U}''_E(t)|\psi\rangle \quad (6.47)$$

$$\begin{aligned} &= V_{\text{H}}^{\text{even}\dagger} \mathcal{U}_A(t) V_{\text{H}}^{\text{even}} \\ &\quad R_E^\dagger V_{\text{H}}^{\text{even}\dagger} \mathcal{U}_A(t) V_{\text{H}}^{\text{even}} R_E \\ &\quad R_E^{2\dagger} V_{\text{H}}^{\text{even}\dagger} \mathcal{U}_A(t) V_{\text{H}}^{\text{even}} R_E^2 |\psi\rangle, \end{aligned} \quad (6.48)$$

where $\mathcal{U}_A(t)$ is the analog propagator generated by H_A , and $V_{\text{H}}^{\text{even}} = \bigotimes_i \text{H}^{(2i)}$. $\text{H}^{(k)}$ represents the application of a Hadamard gate on qubit k . This protocol comprises the following steps:

1. Prepare an initial product state $\otimes_k |\psi_k\rangle$.
2. Apply the cyclic transformation twice with R_E^2 , which is equivalent to R_E^\dagger , on all qubits.
3. Apply Hadamard gates, H, on even qubits.
4. Let the states evolve according to the underlying analog Hamiltonian with analog propagator $\mathcal{U}_A(\tau)$ for time τ .
5. Apply Hadamard gates, H, on even qubits.
6. Undo the double cyclic transformation by applying R_E on all qubits.
7. Apply the cyclic transformation with R_E on all qubits.
8. Apply Hadamard gates, H, on even qubits.
9. Let the states evolve according to the underlying analog Hamiltonian with analog propagator $\mathcal{U}_A(\tau)$ for time τ .
10. Apply Hadamard gates, H, on even qubits.
11. Undo the cyclic transformation with R_E^\dagger on all qubits.
12. Apply Hadamard gates, H, on even qubits.
13. Let the states evolve according to the underlying analog Hamiltonian with analog propagator $\mathcal{U}_A(\tau)$ for time τ .
14. Apply Hadamard gates, H, on even qubits.

This sequence of quantum gates constitutes a block, which can be seen in Fig. 6.6. To evolve with Hamiltonian H_{Heis} for a total time T , one must re-apply the block $M = T/\tau$ times.

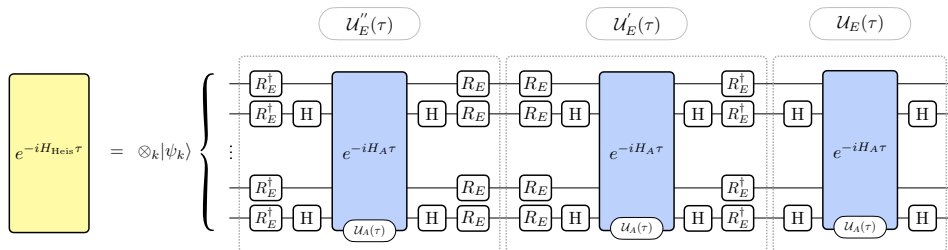


Figure 6.6: **Digital-analog quantum circuit to simulate the evolution of an initial quantum state under Hamiltonian H_{Heis} for a time τ .** This simulation is carried out by conjugating the analog propagator \mathcal{U}_A – which describes the evolution under analog Hamiltonian H_A – by Hadamard gates (H) on even qubits. It is additionally conjugated by $R_E^2 = -R_E^\dagger$, where R_E is a cyclic transformation that allows us to obtain all $S_i S_{i+1}$ interactions. This segment of the circuit simulates the evolution given by $\mathcal{U}_E''(\tau)$. The first repetition of the circuit simulates the evolution given by $\mathcal{U}_E'(\tau)$. This is done by conjugating $\mathcal{U}_A(\tau)$ by Hadamard gates on even qubits, followed by a permutation by R_E . Then, the circuit is repeated a final time, solely conjugating the analog propagator by Hadamard gates on even qubits, to simulate the evolution given by $\mathcal{U}_E(\tau)$.

Digital vs. Digital-Analog Synthesis Errors

In order to quantify the computational benefit of this method, let us compute and compare the above Trotterized error against that of a digitized two-local decomposition. A digitized decomposition we will employ alternating layers of $\sigma_x \sigma_x$, $\sigma_y \sigma_y$, and $\sigma_z \sigma_z$ interactions applied to all even bonds, followed by the same operator action on odd bonds. Such a decomposition is based on the fact that all interactions, on a single bond, commute, but the interactions on adjacent bonds, which share a single spin, do not commute. To first order, the Trotter error is given by

$$\begin{aligned}
 J^2 \|\mathbf{S}_{i-1} \cdot \mathbf{S}_i, \mathbf{S}_i \cdot \mathbf{S}_{i+1}\| &= J^2 \left\| \sum_{\mu, \nu} \sigma_{i-1}^\mu [\sigma_i^\mu, \sigma_i^\nu] \sigma_{i+1}^\nu \right\| \\
 &= 2J^2 \|\mathbf{S}_{i-1} \cdot \mathbf{S}_i \times \mathbf{S}_{i+1}\| \\
 &\leq 12J^2,
 \end{aligned} \tag{6.49}$$

where we have used the fact that $\mathbf{S}_{i-1} \cdot \mathbf{S}_i \times \mathbf{S}_{i+1}$ contains 6 Pauli terms. For a 1D Heisenberg chain, the total commutator is bounded by $12J^2 N$. Meanwhile, on the DAQC side we need to bound

$$\begin{aligned}
 e^{-it(H_E + H'_E + H''_E)} &= e^{-itH_E} e^{-it(H'_E + H''_E)} \\
 &+ \mathcal{O}(t^2[H_E, H'_E + H''_E]) \\
 &= e^{-itH_E} e^{-itH'_E} e^{-itH''_E} \\
 &+ \mathcal{O}(t^2([H_E, H'_E + H''_E] + [H'_E, H''_E]))
 \end{aligned} \tag{6.50}$$

These commutators are

$$\begin{aligned}
 [H_E, H'_E] &= 2iJ^2 \sum_k \sigma_x^{(k-1)} \sigma_z^{(k)} \sigma_y^{(k+1)}, \\
 [H_E, H''_E] &= -2iJ^2 \sum_k \sigma_z^{(k-1)} \sigma_x^{(k)} \sigma_y^{(k+1)}, \\
 [H'_E, H''_E] &= 2iJ^2 \sum_k \sigma_z^{(k-1)} \sigma_y^{(k)} \sigma_x^{(k+1)},
 \end{aligned} \tag{6.51}$$

and their sum can be bounded by

$$\begin{aligned}
 & \left| [H_E, H'_E] + [H_E, H''_E] + [H'_E, H''_E] \right| \\
 &= 2J^2 \left\| \sum_k \sigma_x^{(k-1)} \sigma_z^{(k)} \sigma_y^{(k+1)} - \sigma_z^{(k-1)} \sigma_x^{(k)} \sigma_y^{(k+1)} + \sigma_z^{(k-1)} \sigma_y^{(k)} \sigma_x^{(k+1)} \right\| \\
 &\leq 2J^2 \sum_k \left\| \sigma_x^{(k-1)} \sigma_z^{(k)} \sigma_y^{(k+1)} - \sigma_z^{(k-1)} \sigma_x^{(k)} \sigma_y^{(k+1)} + \sigma_z^{(k-1)} \sigma_y^{(k)} \sigma_x^{(k+1)} \right\| \\
 &\leq 6J^2 N.
 \end{aligned} \tag{6.52}$$

We again find that the bound on the error in the DAQC protocol is smaller by a constant factor than in the DQC approach.

6.4 Practical Implementation

In order to experimentally realize our DA simulation protocols in an accurate manner further practical experimental steps are required. The critical steps for doing so, whose details depend on the user's specific goals, are broadly partitioned as either i) characterization or ii) Hamiltonian optimization. Since each of these steps bring its own theoretical and experimental challenges, we now describe promising paths forward for each step.

A critical step towards validating the accuracy of DA simulations, thereby quantifying their error, is to accurately characterize the analog many-body Hamiltonian at the center of our protocols. While conceptually simple, the characterization of a many-body Hamiltonian is not scalable (with exponentially growing complexity) by naive process tomography [77]. To aid in the scalable characterization of our Hamiltonians, we note that all of the expected interactions are geometrically local and, using this information, one should take advantage of Hamiltonian tomography schemes with polynomial growing model spaces as constricted by locality [118, 119, 120, 121]. Hamiltonian estimation is further complicated by interactions coupling the principle system to unwanted environmental degrees of freedom and, to address this complication, we point the interested reader to recently developed open quantum systems

characterization techniques [122, 123]. Additionally, Bayesian Hamiltonian learning [124] techniques may also be considered, although efficient importance sampling is required to adequately update models in this case.

After experimentally identifying the dominant interactions, a natural next step is to eliminate unwanted couplings. Our analog Hamiltonian arises from a model relying on a two-level approximation and perturbation theory in Ω/δ . However, it is known that the CR-operation comes with a variety of additional terms [125, 126, 127], such as $\sigma_z^{(2)}, \sigma_y^{(2)}, \sigma_z^{(1)}\sigma_z^{(2)}$ as well as spectator phase errors, in practice. One may consider a few routes in order to combat these additional terms. For example, tailoring echo sequences can eliminate certain unwanted interactions [128] and, in addition, it has been shown that residual single qubit interactions can be removed by applying active cancellation tones [125]. Another promising avenue for removing residual interactions comes from judiciously arranging, or actively controlling, qubit frequencies or their relative anharmonicities. For example, Ref. [127] provides a detailed analysis of the role qubit frequencies play and have shown that certain bands in the space of frequency detunings (see regions I and IV in Fig. 4 (f)) maximize the signal-to-noise ratio $\left| \frac{\sigma_z \sigma_x}{\sigma_z \sigma_z} \right|$. Even more recent work [129] has highlighted how additional fixed frequency coupling elements, which dress the qubit level spacings, may also remove unwanted $\sigma_z \sigma_z$ interactions.

Lastly, instead of removing the residual couplings, one may leverage the additional interactions to define new classes of analog Hamiltonians. These analog Hamiltonians would be useful in simulating the dynamics of different spin models. In the limit that these additional terms are sufficiently small, one would expect them to contribute as disorder or small fluctuations in the system parameters. In this case, the (low energy theory and effective) model is expected to still lie in the parent model's universality class. Alternatively, outside this limit, the presence of the additional terms may potentially enrich the computational capability of the analog Hamiltonian as applied to more complex spin models.

6.5 Discussions

In this chapter, we start with a Hamiltonian based on the Rabi model describing two superconducting qubits interacting through the CR effect and propose an extension to a multi-qubit scenario. The resulting Hamiltonian is transformed into a reference frame where only two-body interactions remain, resulting in our analog Hamiltonian. With it, we have assembled a Hamiltonian toolbox through toggling by different single-qubit gates.

The variety of Hamiltonians we have obtained were efficiently combined to simulate Ising, XY , and Heisenberg spin models on a 1-dimensional chain, as well as

the XY model on a 2-dimensional lattice. For the 1D Ising and XY models, our simulation protocols are Trotter-error free up to first order in Ω/δ , meaning that the full time evolution is given by a single DA block. For the 2D XY and 1D Heisenberg chain, we were able to reduce the error in a first order Trotter approximation by a constant factor of 3 for 2D XY and of 2 for the Heisenberg chain. Our techniques, therefore, extend the duration of possible time evolutions by a constant factor. While the constant factor improvement does not provide a polynomial speedup in the asymptotic limit, it does provide a meaningful and practical advantage for near-term, noisy, simulations. A natural avenue of future research could be to explore the possible reach of quantum computation by offering a larger collection of analog Hamiltonians which naturally arise in superconducting platforms. Going beyond our simple Trotter analysis, it would also be interesting to investigate the scaling improvements resulting from the use of the DA Hamiltonians within more advanced product formulas [130] or alternative Hamiltonian simulation techniques [131].

Finally, we have provided a succinct discussion regarding the steps which are necessary to implement our DA protocols in practice. In doing so we outlined promising routes toward scalable characterization and tailoring the precise nature of the analog interactions. Another issue that must be tackled is the problem of geometrically designing the qubit detunings such that all qubits are kept within a particular range. Then, given these detunings, one should increase or decrease the individual driving to maintain a constant ratio Ω/δ for all neighboring pairs. In reality, one must go beyond this simple approximation and will need to calibrate each of the individual drivings as the CR interaction may be highly sensitive to resonances that depend not only on the detuning but also on the qubit's anharmonicities [127].

IN this thesis, we present a comprehensive overview of the work carried out in the field of DAQC over the course of four years. Our aim is to provide a comprehensive reference guide for the implementation of quantum algorithms within the DAQC framework. Through our research, we demonstrate that this computational paradigm presents a promising solution for the constraints of NISQ devices. We have shown that the DAQC framework enables the implementation of quantum algorithms with lower resource consumption, resulting in high-performing solutions. Additionally, we have equipped the reader with a toolbox for simulating various important Hamiltonians within the DAQC framework, which serve as building blocks for the quantum algorithms we have presented in this thesis. These algorithms demonstrate the full potential of the DAQC framework and its ability to effectively tackle complex quantum computing problems.

The thesis begins with an exploration of the construction of DA quantum algorithms based on a combination of single-qubit rotations and multiqubit entangling dynamics, referred to as digital and analog blocks respectively. We provide systematic protocols for constructing generic Ising, 2-body, and M -body Hamiltonians. Through our analysis, we compare the bDAQC and sDAQC approaches of the paradigm and conclude that the bDAQC protocol enhances the fidelity of the global quantum dynamics by reducing the impact of switching on/off the entangling evolution. We also delve into the most relevant noise sources in current superconducting quantum

processors and evaluate the resilience of the DAQC paradigm under realistic implementation circumstances. These results, in conjunction with the systematic protocols outlined in the thesis, enable us to successfully adapt several quantum algorithms.

Our first algorithm of interest is the QFT. This algorithm serves as a critical component in many other quantum algorithms, such as the QPE. To evaluate its performance under realistic conditions, we apply the noise model presented earlier and test the performance of both the QFT and the QPE using both the DQC and DAQC paradigms. Our results demonstrate that the fidelity achieved by the DAQC paradigm improves with an increasing number of qubits.

Next, we delve into the study of the HHL algorithm for solving linear systems of equations. Despite its broad range of applications, this algorithm poses a significant challenge for NISQ devices due to its high number of two-qubit gates and specific qubit connectivity requirements. Additionally, an efficient implementation of the AQE subroutine remains unclear. To tackle these challenges, we provide a systematic approach to implement the AQE subroutine using both the DQC and DAQC paradigms. To mitigate the impact of noise sources when running a quantum algorithm, we also propose a technique to reduce them by minimizing algorithm's depth. This is achieved by co-designing the quantum processor to be tailored to the specific implementation of the desired algorithm, thus reducing the number of SWAP gates needed to artificially connect physically unconnected qubits. We obtain a kite-like quantum processor for the implementation of a generalized version of the HHL algorithm, which is at the heart of many quantum subroutines across various fields. Our findings make a significant contribution to the field and highlight the importance of optimizing quantum algorithms for NISQ devices.

In our quest to extend the application of the DAQC paradigm, we explored its potential in the field of variational quantum algorithms. Our focus was on QAOA and we proposed its implementation using DAQC methods. Our results indicate that the banded DAQC approach provides a more efficient representation of the problem Hamiltonian, with reduced swapping requirements. Additionally, our findings demonstrate that, under the DAQC paradigm, QAOA is more robust to errors compared to traditional approaches, making it a promising candidate for NISQ devices aimed at solving combinatorial optimization problems. This study, together with Ref. [59], marks a significant step towards integrating variational algorithms into the DAQC framework and holds great potential for future research and exploration.

Finally, focusing on superconducting circuits quantum processors, we use the cross-resonance effect as the analog component of the DAQC paradigm. We proposed an extension of the cross-resonance interaction between two superconducting qubits to a multiqubit system. By transforming the resulting Hamiltonian into a reference frame that only includes two-body interactions, we were able to create a toolbox of analog Hamiltonians by toggling different single-qubit gates. Our results, combined with the proposed architecture in [61], provide a strong foundation for continued exploration of quantum algorithms implementation using current technology.

In conclusion, this thesis has presented a comprehensive study on the application of the DAQC paradigm to a wide range of quantum algorithms. Through the com-

combination of single-qubit rotations and multiqubit entangling dynamics, it has been demonstrated a systematic approach to construct generic Ising, 2-body, and M -body Hamiltonian. Our findings showed that the bDAQC protocol offers improved fidelity compared to the stepwise approach, and our noise analysis revealed that it is possible to reduce the impact of noise sources by co-designing the quantum processor. We have applied the DAQC paradigm to the QFT, QPE, HHL, and QAOA quantum algorithms and have shown that it is possible to improve their performance under realistic noise circumstances. Additionally, the study of CR dynamics within the DAQC paradigm provides a promising direction for future research in the field of superconducting circuits quantum processors. Overall, this thesis provides a significant contribution to the field of quantum computing and demonstrates the potential of the DAQC paradigm as a robust and flexible framework for implementing quantum algorithms on NISQ devices. We believe that our work will serve as a foundation for further research and development in this area.

The results of this thesis demonstrate the potential of the DAQC paradigm in solving various problems in quantum computing. With the development of new and more advanced quantum processors, the potential applications of DAQC algorithms will only continue to grow. Furthermore, the study of these algorithms sheds light on the possible techniques to optimize the implementation of quantum algorithms in NISQ devices and reduce their susceptibility to errors. As we look to the future, it is clear that there is much room for further exploration in this field, as the DAQC paradigm has the potential to play a critical role in the development of practical quantum computing solutions. The future direction of this work could include studying the scalability of DAQC algorithms, investigating the use of DAQC in hybrid quantum-classical algorithms, and exploring new techniques to mitigate errors and improve performance. In any case, this thesis represents an important step toward unlocking the full potential of DAQC for quantum computing.

Appendices

A Additional details and proofs regarding the fundamentals of DAQC

A.1 Universality

We briefly recall that a machine able to implement single qubit rotations and an entangling gate, e.g., a controlled-phase gate, has the ability to perform universal quantum computation efficiently. It can be easily proven that a two-qubit ZZ gate, $ZZ_{ij}(\Phi) = e^{-i\Phi\sigma_z^{(i)}\sigma_z^{(j)}}$, combined with single qubit rotations can be used to implement a controlled-phase gate $CZ_{ij}(\phi) = (Z_i(-\phi) \otimes Z_j(-\phi))ZZ_{ij}(\phi/2)$ (up to a general phase) where

$$Z(\phi) = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\phi} \end{pmatrix}, \quad CZ(\phi) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & e^{-i2\phi} \end{pmatrix}. \quad (\text{A.1})$$

A.2 Negative times in the Ising model

Some of the times $t_\alpha = M_{\alpha\beta}^{-1}g_\beta(t_F/g)$ of the analog blocks to implement the Ising model can be negative. In an sDAQC protocol, one solution would be to do such analog evolutions with inverted coupling signs. However, in the bDAQC paradigm, we must keep the base entangling Hamiltonian \bar{H}_{ZZ} untouched. There is a simple way to mimic such behavior by using an equivalent set of times that produces the same evolution.

Given that the vector of times $t_\alpha = (t_1, t_2, \dots, t_{\min}, \dots)$ with $t_{\min} < 0$ and $t_{\min} < t_\alpha \forall \alpha$, there is an equivalent evolution with $N(N-1)/2 - 1$ blocks of times $\tilde{t}_\alpha = t_\alpha + |t_{\min}|e_1$ and an extra analog block with time $|t_{\min}|\lambda_1$, where λ_1 is the eigenvalue of a matrix $M_{\alpha\beta}$ with corresponding constant eigenvector $e_1 = \gamma(1, 1, 1\dots)^T$.

Let us first focus on the properties of the matrix $M_{\alpha\beta}$ defined in Eq. (2.9), created from the signs of applying sequentially gates $\sigma_x^{(n)}\sigma_x^{(m)}$ before and after the analog block evolutions. This matrix has an eigenvector $e_1 = \gamma(1, 1, 1\dots)^T$ which corresponds to the eigenvalue $\lambda_1 = N(N-9)/2 + 8$, that is negative for $N \in \{3, 5, 6\}$, and positive thereafter $N \in \mathbb{Z} \geq 7$. Notice that we omit the case for $N = 4$ as the matrix is non-invertible and different set of gates must be performed.

For $N < 7$ we observe the following identity

$$\begin{aligned} M_{\alpha\beta}(t_\alpha + |t_{\min}|e_1 - |t_{\min}|e_1) &= M_{\alpha\beta}(t_\alpha + |t_{\min}|e_1) - \lambda_1|t_{\min}|e_1 \\ &= M_{\alpha\beta}\tilde{t}_\alpha + |\lambda_1 t_{\min}|e_1 \end{aligned} \quad (\text{A.2})$$

which corresponds to an evolution with $N(N-1)/2 - 1$ blocks of times \tilde{t}_α . One block has zero time ($t_{\min} + |t_{\min}| = 0$), and there is an extra analog block of time $|\lambda_1 t_{\min}|$ not sandwiched by any SQR. The evolution decomposes into

$$U_{ZZ}(t_F) = e^{i \sum_\beta g M_{\alpha\beta} \tilde{t}_\alpha \sigma_z^{(j)} \sigma_z^{(k)}} e^{i |\lambda_1 t_{\min}| \sum_\beta g \sigma_z^{(j)} \sigma_z^{(k)}}, \quad (\text{A.3})$$

where the dependence of j, k on β has not been explicitly written and is the same as in Eq. (2.14).

A.3 Independent coupling parameters in XZ model

The implementation of the XZ model with the protocol of Subsec. 2.1 requires the solution of the following linear system

$$\left[\alpha_j^{(\mu,s)} \alpha_k^{(\nu,s)} \right] \begin{pmatrix} g_{12}^{(1)} \\ g_{12}^{(2)} \\ g_{12}^{(3)} \\ g_{12}^{(4)} \\ g_{13}^{(1)} \\ \vdots \\ g_{N-1,N}^{(4)} \end{pmatrix} = \begin{pmatrix} g_{12}^{XX} \\ g_{12}^{XZ} \\ g_{12}^{ZX} \\ g_{12}^{ZZ} \\ g_{13}^{XX} \\ \vdots \\ g_{N-1,N}^{ZZ} \end{pmatrix}, \quad (\text{A.4})$$

with the matrix

$$\left[\alpha_j^{(\mu,s)} \alpha_k^{(\nu,s)} \right] = \bigoplus_{j < k}^N \begin{pmatrix} S_{\theta_j}^1 S_{\theta_k}^1 & S_{\theta_j}^2 S_{\theta_k}^2 & S_{\theta_j}^3 S_{\theta_k}^3 & S_{\theta_j}^4 S_{\theta_k}^4 \\ S_{\theta_j}^1 C_{\theta_k}^1 & S_{\theta_j}^2 C_{\theta_k}^2 & S_{\theta_j}^3 C_{\theta_k}^3 & S_{\theta_j}^4 C_{\theta_k}^4 \\ C_{\theta_j}^1 S_{\theta_k}^1 & C_{\theta_j}^2 S_{\theta_k}^2 & C_{\theta_j}^3 S_{\theta_k}^3 & C_{\theta_j}^4 S_{\theta_k}^4 \\ C_{\theta_j}^1 C_{\theta_k}^1 & C_{\theta_j}^2 C_{\theta_k}^2 & C_{\theta_j}^3 C_{\theta_k}^3 & C_{\theta_j}^4 C_{\theta_k}^4 \end{pmatrix}, \quad (\text{A.5})$$

where we have defined the parameters

$$S_{\theta_j}^s = \sin(\theta_j^{(s)}) = \alpha_j^{(x,s)} \quad \text{and} \quad C_{\theta_j}^s = \cos(\theta_j^{(s)}) = \alpha_j^{(z,s)}.$$

This matrix is invertible for a dense set of phase values. That is, the sets of phases that make it singular has measure zero. From a practical perspective, we do not want eigenvalues close to zero, because after inversion we would have long simulating times. One useful and well-behaved array of phases is $\theta_w^{(s)} = \frac{s\pi(w)}{2(w+1)}$, with distance between two nearest-neighbour qubit phases scaling polynomially as

$$d(s, w) = |\theta_w^{(s)} - \theta_{w+1}^{(s)}| = \frac{s\pi}{2(w^2 + 3w + 2)}.$$

A.4 M-body Hamiltonians

We extend here the steps explained in subsection 2.2 to simulate a NN Hamiltonian evolution with up to 4-body interactions like

$$\begin{aligned}
H_{4b} &= \sum_{j,\chi\eta} g_{(2,j)}^{\chi\eta} \sigma_{\chi}^{(j)} \sigma_{\eta}^{(j+1)} + \sum_{j,\chi\eta\gamma} g_{(3,j)}^{\chi\eta\gamma} \sigma_{\chi}^{(j)} \sigma_{\eta}^{(j+1)} \sigma_{\gamma}^{(j+2)} \\
&+ \sum_{j,\chi\eta\gamma\rho} g_{(4,j)}^{\chi\eta\gamma\rho} \sigma_{\chi}^{(j)} \sigma_{\eta}^{(j+1)} \sigma_{\gamma}^{(j+2)} \sigma_{\rho}^{(j+3)}, \tag{A.6}
\end{aligned}$$

where $\{\chi, \eta, \gamma, \rho\} = \{x, y, z\}$ and $j = \{1, \dots, N\}$, starting with NN fixed coupling ZZ Ising models. To create terms with support in all interactions by a generalized Mølmer-Sørensen type of gate, we need to interleave inhomogeneous Ising Hamiltonians with two different and rotated XX-Ising evolutions as

$$H_1 = e^{-iO_{XX}^1} H_{ZZ}^1 e^{iO_{XX}^1}, \tag{A.7}$$

$$H_2 = e^{-iO_{XX}^2} H_{ZZ}^2 e^{iO_{XX}^2}, \tag{A.8}$$

where

$$O_{XX}^1 = \Phi_1 \sigma_x^{(1)} \sigma_x^{(2)} + \Phi_3 \sigma_x^{(3)} \sigma_x^{(4)} + \Phi_5 \sigma_x^{(5)} \sigma_x^{(6)} + \dots$$

and its translationally shifted

$$O_{XX}^2 = \Phi_2 \sigma_x^{(2)} \sigma_x^{(3)} + \Phi_4 \sigma_x^{(4)} \sigma_x^{(5)} + \Phi_6 \sigma_x^{(6)} \sigma_x^{(7)} + \dots$$

are built from evolutions of ZZ models rotated with SQRs in all qubits as

$$R = \otimes_j^N \left(\cos(\pi/4) \sigma_z^{(j)} + \sin(\pi/4) \sigma_x^{(j)} \right).$$

For $M = 4$, O_{XX}^1 and O_{XX}^2 contain interacting operators separated by the interaction length $L = M/2 = 2$, e.g., O_{XX}^1 has a term $\sigma_x^{(1)} \sigma_x^{(2)}$ but not $\sigma_x^{(2)} \sigma_x^{(3)}$. Had we wanted to simulate a five-/six- (seven-/eight-) body Hamiltonian, we would need a different decomposition with 3 (4) translationally invariant sets of blocks. The ZZ-Ising NN Hamiltonians $H_{ZZ}^s = \sum_j g_j^s \sigma_z^{(j)} \sigma_z^{(j+1)}$, with $s = \{1, 2\}$.

H_1 contains all two-body and three-body terms with different supports but not

in four-body terms, i.e., for a chain of 8 qubits it looks like

$$\begin{aligned}
 H_1 = & g_1^1 \cos(2\theta_2) \sigma_z^{(1)} \sigma_z^{(2)} + g_1^1 \sin(2\theta_2) \sigma_z^{(1)} \sigma_y^{(2)} \sigma_x^{(3)} + g_2^1 \sigma_z^{(2)} \sigma_z^{(3)} \\
 & + g_3^1 \cos(2\theta_2) \cos(2\theta_4) \sigma_z^{(3)} \sigma_z^{(4)} + g_3^1 \sin(2\theta_2) \cos(2\theta_4) \sigma_x^{(2)} \sigma_y^{(3)} \sigma_z^{(4)} \\
 & + g_3^1 \cos(2\theta_2) \sin(2\theta_4) \sigma_z^{(2)} \sigma_y^{(3)} \sigma_x^{(4)} + g_3^1 \sin(2\theta_2) \sin(2\theta_4) \sigma_x^{(2)} \sigma_y^{(3)} \sigma_y^{(4)} \sigma_x^{(5)} \\
 & + g_4^1 \sigma_z^{(4)} \sigma_z^{(5)} + g_5^1 \cos(2\theta_4) \cos(2\theta_6) \sigma_z^{(5)} \sigma_z^{(6)} \\
 & + g_5^1 \sin(2\theta_4) \cos(2\theta_6) \sigma_x^{(4)} \sigma_y^{(5)} \sigma_z^{(6)} \\
 & + g_5^1 \cos(2\theta_4) \sin(2\theta_6) \sigma_z^{(4)} \sigma_y^{(5)} \sigma_x^{(6)} + g_5^1 \sin(2\theta_4) \sin(2\theta_6) \sigma_x^{(4)} \sigma_y^{(5)} \sigma_y^{(6)} \sigma_x^{(7)} \\
 & + g_6^1 \sigma_z^{(6)} \sigma_z^{(7)} + g_7^1 \cos(2\theta_6) \sigma_z^{(7)} \sigma_z^{(8)} + g_7^1 \sin(2\theta_6) \sigma_x^{(6)} \sigma_y^{(7)} \sigma_z^{(8)}. \tag{A.9}
 \end{aligned}$$

On the other hand, H_2 contains (again) terms in all supports for two-body and three-body interactions and the complementary four-body terms

$$\begin{aligned}
 H_2 = & g_1^2 \sigma_z^{(1)} \sigma_z^{(2)} + g_2^2 \cos(2\theta_1) \cos(2\theta_3) \sigma_z^{(2)} \sigma_z^{(3)} \\
 & + g_2^2 \sin(2\theta_1) \cos(2\theta_3) \sigma_x^{(1)} \sigma_y^{(2)} \sigma_z^{(3)} \\
 & + g_2^2 \cos(2\theta_1) \sin(2\theta_3) \sigma_z^{(2)} \sigma_y^{(3)} \sigma_x^{(4)} + g_2^2 \cos(2\theta_1) \cos(2\theta_3) \sigma_x^{(1)} \sigma_y^{(2)} \sigma_y^{(3)} \sigma_x^{(4)} \\
 & + g_3^2 \sigma_z^{(3)} \sigma_z^{(4)} + g_4^2 \cos(2\theta_3) \cos(2\theta_5) \sigma_z^{(4)} \sigma_z^{(5)} \\
 & + g_4^2 \sin(2\theta_3) \cos(2\theta_5) \sigma_x^{(3)} \sigma_y^{(4)} \sigma_z^{(5)} + g_4^2 \cos(2\theta_3) \sin(2\theta_5) \sigma_z^{(4)} \sigma_y^{(5)} \sigma_x^{(6)} \\
 & + g_4^2 \cos(2\theta_3) \cos(2\theta_5) \sigma_x^{(3)} \sigma_y^{(4)} \sigma_y^{(5)} \sigma_x^{(6)} + g_5^2 \sigma_z^{(5)} \sigma_z^{(6)} \\
 & + g_6^2 \cos(2\theta_5) \cos(2\theta_7) \sigma_z^{(6)} \sigma_z^{(7)} \\
 & + g_6^2 \sin(2\theta_5) \cos(2\theta_7) \sigma_x^{(5)} \sigma_y^{(6)} \sigma_z^{(7)} + g_6^2 \cos(2\theta_5) \sin(2\theta_7) \sigma_z^{(6)} \sigma_y^{(7)} \sigma_x^{(8)} \\
 & + g_6^2 \sin(2\theta_5) \sin(2\theta_7) \sigma_x^{(5)} \sigma_y^{(6)} \sigma_y^{(7)} \sigma_x^{(8)} + g_7^2 \sigma_y^{(7)} \sigma_z^{(8)}. \tag{A.10}
 \end{aligned}$$

The constant coefficients of operators in (A.9) and (A.10) are entangled in groups of maximum size 4. For the simulation of the four-body generalized Hamiltonian, it is thus enough to sum 4 of each of the blocks $H_0 = \sum_{k=1}^4 H_1^{(k)} + H_2^{(k)}$ to decouple the parameters of at least one term operating in each support. We have again a dense set of phases such that randomly chosen ones would make the system of equations invertible. A particular choice of sets that would work are $\theta_1^{(k)} = \theta_{1+4n}^{(k)} = \theta_2^{(k)} = \theta_{2+4n}^{(k)} = (2\pi k/3)$ and $\theta_3^{(k)} = \theta_{3+4n}^{(k)} = \theta_4^{(k)} = \theta_{4+4n}^{(k)} = (2\pi k/5)$ with $n = \{1, \dots, [N/4]\}$ and $k = \{1, \dots, 4\}$.

Finally, we use more local rotations to generate the arbitrary $M \leq 4$ Hamiltonian. In particular, we need to concatenate $3^M = 81$ H_0 blocks, maximum number of independent parameters in an M -body interaction, interleaved by generalized SQRs $R^{(l)} = \otimes_j^N r_j^{(l)} \sigma_x^{(j)} + s_j^{(l)} \sigma_y^{(j)} + t_j^{(l)} \sigma_z^{(j)}$

$$H_{4b} = \sum_l R^{(l)} H_0^{(l)} R^{(l)}, \tag{A.11}$$

where $(r_j^{(l)}, s_j^{(l)}, t_j^{(l)})$ are unit-sphere cartesian decompositions that fulfill the constraint $|r_j^{(l)}|^2 + |s_j^{(l)}|^2 + |t_j^{(l)}|^2 = 1$. As it is common when simulating non-commuting

Hamiltonians, we need to repeat the whole process for each Trotter step $e^{-iH_{4b}t} \approx (e^{-iH_{4b}t/n_T})^{n_T}$ to approximate the evolution.

B Detailed calculations regarding the cross-resonance effect

B.1 CR Hamiltonian

Here we provide further details of the derivation of the effective Hamiltonians described in section 6.1 of chapter 6.

B.1.1 Two-qubit case

The transformation that takes the Hamiltonian in Eq. 6.1 into a doubly rotating frame is given by

$$U_{12} = \exp \left[-\frac{i}{2}((\omega_1 t + \phi_1)z_1 + (\omega_2 t + \phi_2)z_2) \right]. \quad (\text{B.1})$$

This operation results in

$$\begin{aligned}
 H_2 &= \frac{1}{2}(\delta_1 z_1 + \delta_2 z_2) + \Omega_1 \cos(\omega_1 t + \phi_1)[x_1 \cos(\omega_1 t + \phi_1) - y_1 \sin(\omega_1 t + \phi_1)] \\
 &+ \Omega_2 \cos(\omega_2 t + \phi_2)[x_2 \cos(\omega_2 t + \phi_2) - y_2 \sin(\omega_2 t + \phi_2)] \\
 &+ \frac{g}{2}[x_1 \cos(\omega_1 t + \phi_1) - y_1 \sin(\omega_1 t + \phi_1)] \\
 &\quad [x_2 \cos(\omega_2 t + \phi_2) - y_2 \sin(\omega_2 t + \phi_2)], \tag{B.2}
 \end{aligned}$$

with $\delta_k = \omega_k^q - \omega_k$. Next a rotating wave approximation (RWA) is performed by dropping terms proportional to $e^{\pm 2i\omega_1}$, $e^{\pm 2i\omega_2}$, and $e^{\pm i(\omega_1 + \omega_2)}$. The validity of this approximation relies on a time-average of the Hamiltonian and noting that $\Omega/(\omega_i + \omega_j) \ll 1$ and $g/(\omega_i + \omega_j) \ll 1$, $\forall i, j$. The remaining terms are either static, or rotating at $\pm(\omega_1 - \omega_2)$:

$$\begin{aligned}
 H_2 &= \frac{1}{2}(\delta_1 z_1 + \delta_2 z_2) + \frac{1}{2}(\Omega_1 x_1 + \Omega_2 x_2) \\
 &+ \frac{g}{4}[\cos \varphi_{12}(t)(x_1 x_2 + y_1 y_2) + \sin \varphi_{12}(t)(x_1 y_2 - y_1 x_2)], \tag{B.3}
 \end{aligned}$$

where we defined $\varphi_{12}(t) = (\omega_1 - \omega_2)t + \phi_1 - \phi_2$. Next we apply the rotation

$$U_3 = \exp \left[\frac{i}{2}(\xi_1 y_1 + \xi_2 y_2) \right], \tag{B.4}$$

with $\tan \xi_k = \delta_k / \Omega_k$. The resulting Hamiltonian is

$$\begin{aligned}
 H_3 &= \frac{1}{2} \left(\frac{\Omega_1}{\cos \xi_1} x_1 + \frac{\Omega_2}{\cos \xi_2} x_2 \right) \\
 &+ \frac{g}{4} \{ \cos \varphi_{12}(t)[(x_1 \cos \xi_1 - z_1 \sin \xi_1)(x_2 \cos \xi_2 - z_2 \sin \xi_2) + y_1 y_2] \\
 &\quad + \sin \varphi_{12}(t)[(x_1 \cos \xi_1 - z_1 \sin \xi_1)y_2 - y_1(x_2 \cos \xi_2 - z_2 \sin \xi_2)] \}, \tag{B.5}
 \end{aligned}$$

where we have used $\delta_k \cos \xi_k - \Omega_k \sin \xi_k = 0$ and $\delta_k \sin \xi_k + \Omega_k \cos \xi_k = \Omega_k / \cos \xi_k$. The last transformation is given by

$$U_4 = \exp \left[-\frac{it}{2}(\eta_1 x_1 + \eta_2 x_2) \right], \tag{B.6}$$

where $\eta_k = \sqrt{\delta_k^2 + \Omega_k^2}$, such that $\eta_k = \Omega_k / \cos \xi_k = \delta_k / \sin \xi_k$. This takes our Hamiltonian into the quad frame (QF),

$$\begin{aligned}
 H_4 = & \frac{g}{4} \{ \cos \varphi_{12}(t) [x_1 x_2 \cos \xi_1 \cos \xi_2 - x_1 \cos \xi_1 \sin \xi_2 (z_2 \cos \eta_2 t + y_2 \sin \eta_2 t) \\
 & - (z_1 \cos \eta_1 t + y_1 \sin \eta_1 t) x_2 \sin \xi_1 \cos \xi_2 \\
 & + (z_1 \cos \eta_1 t + y_1 \sin \eta_1 t) (z_2 \cos \eta_2 t + y_2 \sin \eta_2 t) \sin \xi_1 \sin \xi_2 \\
 & + (y_1 \cos \eta_1 t - z_1 \sin \eta_1 t) (y_2 \cos \eta_2 t - z_2 \sin \eta_2 t)] \\
 & + \sin \varphi_{12}(t) [x_1 \cos \xi_1 (y_2 \cos \eta_2 t - z_2 \sin \eta_2 t) \\
 & - (z_1 \cos \eta_1 t + y_1 \sin \eta_1 t) (y_2 \cos \eta_2 t - z_2 \sin \eta_2 t) \sin \xi_1 \\
 & - (y_1 \cos \eta_1 t - z_1 \sin \eta_1 t) x_2 \cos \xi_2 \\
 & + \sin \xi_2 (y_1 \cos \eta_1 t - z_1 \sin \eta_1 t) (z_2 \cos \eta_2 t + y_2 \sin \eta_2 t)] \}. \tag{B.7}
 \end{aligned}$$

Now, we consider the scenario in which we drive the first qubit at the resonance frequency of the second qubit by imposing that $\omega_1 = \omega_2^g$, while the second qubit is not driven, i.e. $\Omega_2 = 0$, $\eta_2 = 0$, $\delta_2 = 0$, $\omega_2 = \omega_2^g$, $\xi_2 = 0$, $\phi_2 = 0$ which implies $\varphi_{12}(t) = \phi_1$. The resulting Hamiltonian is

$$\begin{aligned}
 H_4 = & \frac{g}{4} \{ \cos \phi_1 [x_1 x_2 \cos \xi_1 - (z_1 \cos \eta_1 t + y_1 \sin \eta_1 t) x_2 \sin \xi_1 \\
 & + (y_1 \cos \eta_1 t - z_1 \sin \eta_1 t) y_2] \tag{B.8}
 \end{aligned}$$

$$\begin{aligned}
 & + \sin \phi_1 [x_1 y_2 \cos \xi_1 - (z_1 \cos \eta_1 t + y_1 \sin \eta_1 t) y_2 \sin \xi_1 \\
 & - (y_1 \cos \eta_1 t - z_1 \sin \eta_1 t) x_2 \}, \tag{B.9}
 \end{aligned}$$

where we see that static terms have developed from the slowly rotating terms we kept in the RWA, since with the cross-resonant driving $\omega_1 - \omega_2 = \omega_2^g - \omega_2 = 0$. Finally, we perform a second RWA by dropping any term proportional to $e^{\pm i \eta_1 t}$, and keep only the static terms. Additionally, we consider the weak-driving regime ($\Omega_1 / \delta_1 \ll 1$), which simplifies $\cos \xi_1 \approx \Omega_1 / \delta_1$. Then, we arrive at the Hamiltonian

$$H_4 = \frac{g \Omega_1}{4 \delta_1} (x_1 x_2 \cos \phi_1 + x_1 y_2 \sin \phi_1), \tag{B.10}$$

presented in Eq. 6.2. The validity of this approximation relies on $g / \eta_1 \approx g / \delta_1 \ll 1$, which is enforced in the weak-coupling regime. See that the remaining terms after this second RWA are those we kept as slow-rotating after the first RWA, and the terms neglected in this case oscillate with $\delta_1 = \omega_1^g - \omega_1 = \omega_1^g - \omega_2^g$.

B.1.2 N-qubit case

We start with the N-qubit Hamiltonian in the laboratory frame, given by Eq. 6.4 in the main text. We can move to the QF by applying the following transformations

$$U_{12} = \exp \left[-\frac{i}{2} \sum_{k=1}^N (\omega_k t + \phi_k) z_k \right], \quad (\text{B.11})$$

$$U_3 = \exp \left[\frac{i}{2} \sum_{k=1}^N \xi_k y_k \right], \quad \text{and} \quad (\text{B.12})$$

$$U_4 = \exp \left[-\frac{it}{2} \sum_{k=1}^N \eta_k x_k \right]. \quad (\text{B.13})$$

Now, as stated in the main text, we drive all qubits at the resonance frequency of their neighbour to the right (except for the last one when applicable). This implies that $\omega_k = \omega_{k+1}^q$, $\varphi_k(t) = \delta_{k+1}t + \phi_k - \phi_{k+1}$ and, in the weak-driving regime $\Omega_k \ll \delta_k$, $\eta_k \approx \delta_k$. This results in

$$\begin{aligned} H_4 = & \frac{1}{4} \sum_{k=1}^{N-1} g_k \left\{ \cos(\delta_k t + \phi_k - \phi_{k+1})(y_k y_{k+1} + z_k z_{k+1}) \right. \\ & + \sin(\delta_k t + \phi_k - \phi_{k+1})(y_k z_{k+1} - z_k y_{k+1}) \\ & + \frac{\Omega_k}{\delta_k} \left[\sin(\phi_k - \phi_{k+1}) x_k y_{k+1} - \cos(\phi_k - \phi_{k+1}) x_k z_{k+1} \right] \\ & - \frac{\Omega_{k+1}}{\delta_{k+1}} \left[\sin[(\delta_k + \delta_{k+1})t + \phi_k - \phi_{k+1}] y_k x_{k+1} \right. \\ & \left. \left. + \cos[(\delta_k + \delta_{k+1})t + \phi_k - \phi_{k+1}] z_k x_{k+1} \right] \right\}. \end{aligned} \quad (\text{B.14})$$

The next step is to perform the RWA by neglecting all fast oscillating terms, with frequencies δ_k and $\delta_k + \delta_{k+1}$, while keeping the static ones. The resulting Hamiltonian, in the QF, is given by

$$H_4 = \sum_{k=1}^{N-1} \frac{g_k \Omega_k}{4\delta_k} x_k (y_{k+1} \sin(\phi_k - \phi_{k+1}) - z_{k+1} \cos(\phi_k - \phi_{k+1})), \quad (\text{B.15})$$

as appears in Eq. 6.5.

which, when written back in the DF results in

$$H_{DF} = \frac{1}{2} \sum_{k=1}^N (\Omega_k x_k + \delta_k z_k) + \sum_{k=1}^{N-1} \frac{g_k \Omega_k}{4\delta_k} z_k (x_{k+1} \cos \varphi_k(t) + y_{k+1} \sin \varphi_k(t)). \quad (\text{B.16})$$

Again, for completeness, we can write this effective Hamiltonian

$$\begin{aligned}
 H_{\text{IBM}}^{\text{eff}} &= \sum_{k=1}^{N-1} \frac{\Omega_k}{2} \left[x_k \cos(\delta_k t - \phi_k) - y_k \sin(\delta_k t - \phi_k) \right. \\
 &\quad \left. + \frac{g_k}{2\delta_k} z_k (x_{k+1} \cos \phi_k + y_{k+1} \sin \phi_k) \right], \tag{B.17}
 \end{aligned}$$

as well as the original one

$$\begin{aligned}
 H_{\text{IBM}} &= \frac{1}{2} \sum_{k=1}^{N-1} \left[\Omega_k (x_k \cos(\delta_k t - \phi_k) - y_k \sin(\delta_k t - \phi_k)) \right. \\
 &\quad \left. + \frac{g_k}{2} (\cos \delta_k t (x_k x_{k+1} + y_k y_{k+1}) + \sin \delta_k t (x_k y_{k+1} - y_k x_{k+1})) \right] \tag{B.18}
 \end{aligned}$$

in IBM's frame.

To move to the IBM's frame we need to apply the following transformation

$$V = \exp \left[-\frac{i}{2} \sum_{k=1}^N (\delta_k t - \phi_k) z_k \right], \tag{B.19}$$

which leads directly to the equations shown in the main text.

B.2 Synthesis errors

In this appendix, we want to show the synthesis errors corresponding to the toggled Hamiltonians. For the XY model, the original Hamiltonian is

$$\begin{aligned}
 H_{XY}^{\text{org}} &= \frac{g}{4} \sum_{k=1}^{N-1} \left\{ -\frac{\Omega}{\delta} (y_k y_{k+1} + x_k x_{k+1}) + (x_k y_{k+1} + y_k x_{k+1} - 2z_k z_{k+1}) \cos \delta t \right. \\
 &\quad + [z_k (y_{k+1} - x_{k+1}) + (x_k - y_k) z_{k+1}] \sin \delta t \\
 &\quad \left. + \frac{\Omega}{\delta} [z_k (y_{k+1} - x_{k+1}) \sin 2\delta t - (y_k y_{k+1} + x_k x_{k+1}) \cos 2\delta t] \right\}. \tag{B.20}
 \end{aligned}$$

Then, the difference between original and effective Hamiltonians,

$$\begin{aligned}
 \Delta H_{XY} &= \frac{g}{4} \sum_{k=1}^{N-1} \left\{ (x_k y_{k+1} + y_k x_{k+1} - 2z_k z_{k+1}) \cos \delta t \right. \\
 &\quad + [z_k (y_{k+1} - x_{k+1}) + (x_k - y_k) z_{k+1}] \sin \delta t \\
 &\quad \left. + \frac{\Omega}{\delta} [z_k (y_{k+1} - x_{k+1}) \sin 2\delta t - (y_k y_{k+1} + x_k x_{k+1}) \cos 2\delta t] \right\}, \tag{B.21}
 \end{aligned}$$

constitutes the error we want to estimate. We find the Frobenius norm is given by

$$\|\Delta H_{XY}\|_F = \frac{g}{2}\sqrt{N-1}. \quad (\text{B.22})$$

On the other hand, the original ZZ toggled Hamiltonian is

$$\begin{aligned} H_{ZZ}^{org} &= \frac{g}{4} \sum_{k=1}^{N-1} \left\{ \left[z_k \frac{\Omega}{\delta} - x_k \cos \delta t + y_k \sin \delta t \right] z_{k+1} \right. \\ &+ (y_k \cos \delta t + x_k \sin \delta t) y_{k+1} \\ &+ \cos \varphi_k(t) \left[z_k (z_{k+1} \frac{\Omega}{\delta} - x_{k+1} \cos \delta t + y_{k+1} \sin \delta t) \right. \\ &+ \left. \left. y_k (y_{k+1} \cos \delta t + x_{k+1} \sin \delta t) \right] \right. \\ &+ \left. \sin \varphi_k(t) \left[-z_k (y_{k+1} \cos \delta t + x_{k+1} \sin \delta t) \right. \right. \\ &+ \left. \left. y_k (z_{k+1} \frac{\Omega}{\delta} - x_{k+1} \cos \delta t + y_{k+1} \sin \delta t) \right] \right\}, \quad (\text{B.23}) \end{aligned}$$

so that the difference is

$$\begin{aligned} \Delta H_{ZZ} &= \frac{g}{4} \sum_{k=1}^{N-1} \left\{ (-x_k \cos \delta t + y_k \sin \delta t) z_{k+1} + (y_k \cos \delta t + x_k \sin \delta t) y_{k+1} \right. \\ &+ \cos \varphi_k(t) \left[z_k (z_{k+1} \frac{\Omega}{\delta} - x_{k+1} \cos \delta t + y_{k+1} \sin \delta t) \right. \\ &+ \left. y_k (y_{k+1} \cos \delta t + x_{k+1} \sin \delta t) \right] \\ &+ \sin \varphi_k(t) \left[-z_k (y_{k+1} \cos \delta t + x_{k+1} \sin \delta t) \right. \\ &+ \left. y_k (z_{k+1} \frac{\Omega}{\delta} - x_{k+1} \cos \delta t + y_{k+1} \sin \delta t) \right] \left. \right\}. \quad (\text{B.24}) \end{aligned}$$

The Frobenius norm is then given by

$$\|\Delta H_{ZZ}\|_F = \frac{g}{2\sqrt{2}}\sqrt{N-1} \sqrt{2 + \cos \delta t \cos(\varphi_k(t) - \delta t) + \frac{\Omega}{\delta} \sin \delta t \sin \varphi_k(t)}. \quad (\text{B.25})$$

B.3 Unitary transformation to the Quad Frame

In order to perform a quantum simulation on the QF, we need to translate the state of our circuit to this frame. Then, considering a simulation scenario in any IBM

superconducting chip, we want to find a simple expression for the combination of rotations we need to apply in order to move from IBM's frame into the QF. For that, we will expand the product

$$U_{\text{IBM}}^\dagger U_{12} U_3 U_4 = e^{\frac{it}{2} \sum_{k=1}^{N-1} \omega_k^q z_k} e^{-\frac{it}{2} \sum_{k=1}^{N-1} \omega_k z_k} e^{\frac{i}{2} \sum_{k=1}^{N-1} \xi_k y_k} e^{-\frac{it}{2} \sum_{k=1}^{N-1} \eta_k x_k}, \quad (\text{B.26})$$

having set $\phi_k = \phi = 0$. See that the first two exponentials can be combined, such that

$$U_{\text{IBM}}^\dagger U_{12} U_3 U_4 = e^{\frac{it}{2} \sum_{k=1}^{N-1} \delta_k z_k} e^{\frac{i}{2} \sum_{k=1}^{N-1} \xi_k y_k} e^{-\frac{it}{2} \sum_{k=1}^{N-1} \eta_k x_k}, \quad (\text{B.27})$$

where $\delta_k = \omega_k^q - \omega_k$. Now, it is satisfied that

$$\exp \left[i \sum_{k=1}^N \theta_k \sigma_k \right] = \prod_{k=1}^N e^{i \theta_k \sigma_k} \quad (\text{B.28})$$

for $\sigma = x, y$, or z . This means that we can write

$$U_{\text{IBM}}^\dagger U_{12} U_3 U_4 = \prod_{k=1}^{N-1} e^{\frac{it}{2} \delta_k z_k} e^{\frac{i}{2} \xi_k y_k} e^{-\frac{it}{2} \eta_k x_k}, \quad (\text{B.29})$$

and we can use the Euler form for Pauli matrices,

$$e^{i\theta\sigma} = \cos \theta \mathbb{I} + i \sin \theta \sigma, \quad (\text{B.30})$$

to express these rotations as

$$\begin{aligned} U_{\text{IBM}}^\dagger U_{12} U_3 U_4 &= \prod_{k=1}^{N-1} \left(\cos \frac{\delta_k t}{2} \mathbb{I}_k + i \sin \frac{\delta_k t}{2} z_k \right) \left(\cos \frac{\xi_k}{2} \mathbb{I}_k + i \sin \frac{\xi_k}{2} y_k \right) \\ &\quad \left(\cos \frac{\eta_k t}{2} \mathbb{I}_k - i \sin \frac{\eta_k t}{2} x_k \right). \end{aligned} \quad (\text{B.31})$$

Recall that, working in the regime $\Omega \ll \delta$, we had approximated $\eta \approx \delta$, $\sin \xi \approx 1$, and $\cos \xi \approx \Omega/\delta$. Knowing that $\sin \theta/2 = \sqrt{(1 - \cos \theta)/2}$ and $\cos \theta/2 = \sqrt{(1 + \cos \theta)/2}$, we can simplify

$$\cos \frac{\xi_k t}{2} \mathbb{I}_k + i \sin \frac{\xi_k t}{2} y_k \approx \frac{1}{\sqrt{2}} \left(\mathbb{I}_k + i y_k + \frac{\Omega_k}{2\delta_k} (\mathbb{I}_k - i y_k) \right), \quad (\text{B.32})$$

where we have used $\sqrt{1 \pm x} \approx 1 \pm x/2$ for small x . In this expansion, we eventually find

$$U_{\text{IBM}}^\dagger U_{12} U_3 U_4 = \prod_{k=1}^{N-1} \frac{1}{\sqrt{2}} \left[\mathbb{I}_k + i y_k + \frac{\Omega_k}{2\delta_k} ((\mathbb{I}_k - i y_k) \cos \delta_k t + i (z_k - x_k) \sin \delta_k t) \right], \quad (\text{B.33})$$

which we will denote by V_{QF} . Let us check the unitarity of this operator by computing

$$V_{\text{QF}} V_{\text{QF}}^\dagger = \mathbb{I} + \mathcal{O} \left(\frac{\Omega^2}{\delta^2} \right). \quad (\text{B.34})$$

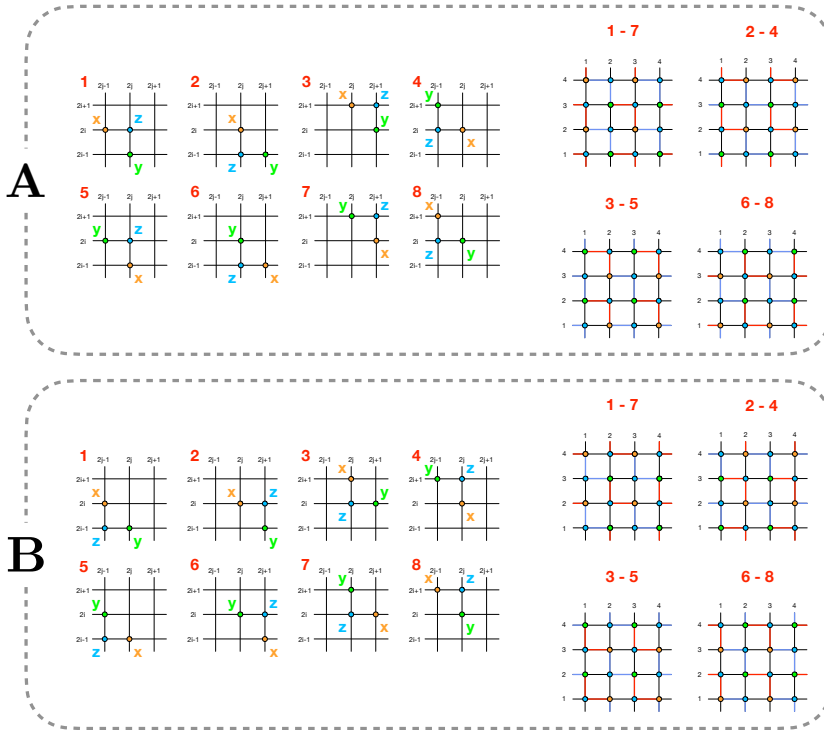


Figure B.1: **Representation of the spin triplets, grouped in A and B .** These triplets, inside either A or B , can be grouped in pairs, which can be used to tile the entire lattice with staircase patterns. These tilings form strings with periodic boundary conditions along the diagonals of the lattice, which we transform from spins to fermions in order to estimate $\|A\| + \|B\|$, the Trotter error associated with the DA simulation of the XY model in 2D.

The previous calculations were set in the weak-driving regime ($\Omega/\delta \ll 1$), considering terms up to first order in Ω/δ and neglecting higher orders. This is consistent with the approximations we have made here, and thus the unitarity of V_{QF} relies on these approximations.

B.4 2D XY model

In this appendix, we describe the transformation from a spin lattice to a string of fermions with 2-site hopping, which allows us to estimate more accurately the Trotter error associated to the simulation of the XY model in 2 dimensions. This

error will be given by the commutator $[H_I, H_{II}]$, split into $A = [H_I^{yy}, H_{II}^{xx}]$ and $B = [H_I^{xx}, H_{II}^{yy}]$. Both A and B include 8 terms, each one having the form x - z - y at three different vertices, as we can see in Fig. B.1. Assembling pairs of these terms, joining $x_i z_{i+1} y_{i+2}$ with $y_{i+2} z_{i+3} x_{i+4}$ or $y_i z_{i+1} x_{i+2}$ with $x_{i+2} z_{i+3} y_{i+4}$, either in A or in B , we can construct diagonal strings in the lattice, which we also represent in Fig. B.1. These diagonals can then be thought of as 1-d strings, and given this outline, we can apply a Jordan-Wigner transformation and introduce Majorana fermion operators. Let us take a couple of terms (1 & 7) in B to illustrate this issue:

$$\sum_{i,j} (x_{2i,2j-1} z_{2i-1,2j-1} y_{2i-1,2j} + x_{2i,2j+1} z_{2i,2j} y_{2i+1,2j}), \quad (\text{B.35})$$

can be turned into a string as

$$\sum_{j=1}^{N^2/4} (x_{4j-3} z_{4j-2} y_{4j-1} + y_{4j-1} z_{4j} x_{4j+1}). \quad (\text{B.36})$$

The Jordan-Wigner transformation, followed by defining Majorana operators $\gamma_i^{(1)} = c_i + c_i^\dagger$ and $\gamma_i^{(2)} = -i(c_i^\dagger - c_i)$, transforms spins as

$$x_j z_{j+1} y_{j+2} = -i \gamma_j^{(2)} \gamma_{j+2}^{(2)} \quad (\text{B.37})$$

and

$$y_j z_{j+1} x_{j+2} = i \gamma_j^{(1)} \gamma_{j+2}^{(1)}. \quad (\text{B.38})$$

These transformations lead to the Hamiltonian

$$-i \sum_{j=1}^{N^2/4} \left(\gamma_{4j-3}^{(2)} \gamma_{4j-1}^{(2)} - \gamma_{4j-1}^{(1)} \gamma_{4j+1}^{(1)} \right). \quad (\text{B.39})$$

The Fourier transform of Majorana operators is given by

$$\begin{aligned} \gamma_j^{(1)} &= \frac{1}{\sqrt{N}} \sum_k \left(\gamma_k^{(1)} \cos kj + \gamma_k^{(2)} \sin kj \right), \\ \gamma_j^{(2)} &= \frac{1}{\sqrt{N}} \sum_k \left(\gamma_k^{(2)} \cos kj - \gamma_k^{(1)} \sin kj \right), \end{aligned} \quad (\text{B.40})$$

together with

$$\begin{aligned} \sum_j \gamma_j^{(1)} \gamma_{j+2}^{(1)} &= \frac{1}{2} \sum_k \left[\cos 2k \left(\gamma_k^{(1)} \gamma_k^{(1)} + \gamma_k^{(2)} \gamma_k^{(2)} + \gamma_k^{(1)} \gamma_{-k}^{(1)} - \gamma_k^{(2)} \gamma_{-k}^{(2)} \right) \right. \\ &\quad \left. + \sin 2k \left(\gamma_k^{(1)} \gamma_k^{(2)} - \gamma_k^{(2)} \gamma_k^{(1)} - \gamma_k^{(1)} \gamma_{-k}^{(2)} - \gamma_k^{(2)} \gamma_{-k}^{(1)} \right) \right] \\ \sum_j \gamma_j^{(2)} \gamma_{j+2}^{(2)} &= \frac{1}{2} \sum_k \left[\cos 2k \left(\gamma_k^{(1)} \gamma_k^{(1)} + \gamma_k^{(2)} \gamma_k^{(2)} - \gamma_k^{(1)} \gamma_{-k}^{(1)} + \gamma_k^{(2)} \gamma_{-k}^{(2)} \right) \right. \\ &\quad \left. + \sin 2k \left(\gamma_k^{(1)} \gamma_k^{(2)} - \gamma_k^{(2)} \gamma_k^{(1)} + \gamma_k^{(1)} \gamma_{-k}^{(2)} + \gamma_k^{(2)} \gamma_{-k}^{(1)} \right) \right], \end{aligned} \quad (\text{B.41})$$

which leads to the expression of a string in Fourier space,

$$i \sum_k \left[\cos 2k \left(\gamma_k^{(1)} \gamma_{-k}^{(1)} - \gamma_k^{(2)} \gamma_{-k}^{(2)} \right) - \sin 2k \left(\gamma_k^{(1)} \gamma_{-k}^{(2)} + \gamma_k^{(2)} \gamma_{-k}^{(1)} \right) \right]. \quad (\text{B.42})$$

The resulting matrix is block-diagonal, and it has two types of blocks. The first block, Γ_1 , contains the elements $\cos 2k$ for $k \in [-\pi, \pi]$, and the second block, Γ_2 , contains the elements $\sin 2k$. Both blocks have elements only along the anti-diagonal. Then, the matrix in Fourier space looks like

$$\begin{pmatrix} \Gamma_1 & -\Gamma_2 \\ -\Gamma_2 & -\Gamma_1 \end{pmatrix}. \quad (\text{B.43})$$

We can write the eigenvalue problem as

$$\det \left[\begin{pmatrix} \Gamma_1 & -\Gamma_2 \\ -\Gamma_2 & -\Gamma_1 \end{pmatrix} - \begin{pmatrix} \mathbb{I}\lambda_+ & 0 \\ 0 & \mathbb{I}\lambda_- \end{pmatrix} \right] = 0, \quad (\text{B.44})$$

for $\lambda_+ = \lambda$ and $\lambda_- = -\lambda$. Given that the determinant of this matrix is

$$\det(\Gamma_1 + \Gamma_2 \Gamma_1^{-1} \Gamma_2) \det \Gamma_1 = 1,$$

we have a hint that $\lambda = \pm 1$, and in fact

$$\det \left[\begin{pmatrix} \Gamma_1 & -\Gamma_2 \\ -\Gamma_2 & -\Gamma_1 \end{pmatrix} - \begin{pmatrix} \mathbb{I}\lambda & 0 \\ 0 & -\mathbb{I}\lambda \end{pmatrix} \right] = (1 - \lambda^2)^{N^2/2} = 0 \quad (\text{B.45})$$

As we can see, the eigenvalues of this matrix are $\lambda_j = \pm 1$. Then, the spectral norm of this pair of elements in the commutator is equal to one, for any lattice size. Since there are 4 total pairs in B , we will get that the norm can be estimated as $\|B\| \leq 8J^2$, taking into account the $2i$ factor from Pauli commutation relations, and the J^2 from the analog Hamiltonian. Given that the norm of A can be estimated in the same way, we have

$$\|[H_I, H_{II}]\| = \|-A + B\| \leq \|A\| + \|B\| \leq 16J^2, \quad (\text{B.46})$$

where we have eliminated the dependence on the system size.

Bibliography

- [1] P. Benioff, *The computer as a physical system: A microscopic quantum mechanical Hamiltonian model of computers as represented by Turing machines*. Journal of Statistical Physics, **22**, 563 (1980).
- [2] Y. I. Manin, *Vychislimoe i nevychislimoe (Computable and Noncomputable)* (Soviet Radio, Moscow, 1980).
- [3] R. P. Feynman, *Simulating physics with computers*. International journal of theoretical physics **21**, 467 (1982).
- [4] S. Lloyd, *Universal Quantum Simulators*. Science **273**, 1073 (1996).
- [5] D. Deutsch, *Quantum theory, the Church-Turing Principle and the universal quantum computer*. Proceedings of the Royal Society of London A **400**, 1818 (1985).
- [6] D. Deutsch, and R. Jozsa, *Rapid solution of problems by quantum computation*. Proceedings of the Royal Society of London A **439**, 1907 (1992).
- [7] P. W. Shor, *Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer*. SIAM Review **26**, 1484 (1996).
- [8] L. K. Grover, *Proceedings of the 28th Annual ACM Symposium on the Theory of Computing (STOC)*. ACM, New York (1996).

- [9] K. Bharti, A. Cervera-Lierta, T. H. Kyaw, T. Haug, S. Alperin-Lea, A. Anand, M. Degroote, H. Heimonen, J. S. Kottmann, T. Menke, W. Mok, S. Sim, L. Kwek, and A. Aspuru-Guzik, Noisy intermediate-scale quantum algorithms. *Reviews of Modern Physics* **94**, 015004 (2022).
- [10] J. I. Cirac, and P. Zoller, *Quantum computations with cold trapped ions*. *Physical Review Letters* **74**, 4091 (1995).
- [11] D. G. Cory, A. F. Fahmy, and T. F. Havel, *Ensemble quantum computing by NMR spectroscopy*. *Proceedings of the National Academy of Sciences USA* **94**, 1634 (1997).
- [12] N. A. Gershenfeld, and I. L. Chuang, *Bulk spin resonance quantum computation*. *Science* **275**, 350 (1997).
- [13] D. G. Cory, M. D. Price, and T. F. Havel, *Nuclear magnetic resonance spectroscopy: An experimentally accessible paradigm for quantum computing*. *Physica D* **120**, 82 (1998).
- [14] B. E. Kane, *A silicon-based nuclear spin quantum computer*. *Nature* **393**, 133 (1998).
- [15] D. Loss, and D. P. DiVincenzo, *Quantum computation with quantum dots*. *Physical Review A* **57**, 120 (1998).
- [16] Y. Nakamura, Y. A. Pashkin, and J. S. Tsai, *Coherent control of macroscopic quantum states in a single Cooper-pair box*. *Nature* **398**, 786 (1999).
- [17] D. P. DiVincenzo, *Topics in Quantum Computers*. In: L.L. Sohn, L.P. Kouwenhoven, and G. Schön, (eds) *Mesoscopic Electron Transport*. NATO ASI Series **3**, 45. Springer (1997).
- [18] J. A. Jones, and M. Mosca, *Implementation of a quantum algorithm on a nuclear magnetic resonance quantum computer*. *Journal of Chemical Physics* **109**, 5 (1998).
- [19] I. L. Chuang, N. Gershenfeld, and M. Kubinec, *Experimental Implementation of Fast Quantum Searching*. *Physical Review Letters* **80**, 3408 (1998).
- [20] N. C. Menicucci, and C. M. Caves, *Local realistic model for the dynamics of bulk-ensemble NMR information processing*. *Physical Review Letters* **88**, 167901 (2002).
- [21] I. Buluta, and F. Nori, *Quantum simulators*. *Science* **326**, 108 (2009).
- [22] I. M. Georgescu, S. Ashhab, and F. Nori, *Quantum simulation*. *Reviews of Modern Physics* **86**, 153 (2014).

-
- [23] D. Ballester, G. Romero, J. J. García-Ripoll, F. Deppe, and E. Solano, *Quantum Simulation of the Ultrastrong-Coupling Dynamics in Circuit Quantum Electrodynamics*. Physical Review X **2**, 021007 (2012).
- [24] A. Mezzacapo, U. Las Heras, J. S. Pedernales, L. DiCarlo, E. Solano, and L. Lamata, *Digital quantum Rabi and Dicke models in superconducting circuits*. Scientific Reports **4**, 7482 (2014).
- [25] J. S. Pedernales, I. Lizuain, S. Felicetti, G. Romero, L. Lamata, and E. Solano, *Quantum Rabi model with trapped ions*. Scientific Reports **5**, 15472 (2015).
- [26] J. Braumüller, M. Marthaler, A. Schneider, A. Stehli, H. Rotzinger, M. Weides, and A. V. Ustinov, *Analog quantum simulation of the Rabi model in the ultra-strong coupling regime*. Nature Communications **8**, 779 (2017).
- [27] D. Lv, S. An, Z. Liu, J.-N. Zhang, J. S. Pedernales, L. Lamata, E. Solano, and K. Kim, *Quantum Simulation of the Quantum Rabi Model in a Trapped Ion*. Physical Review X **8**, 021027 (2018).
- [28] S. Felicetti, M. Sanz, L. Lamata, G. Romero, G. Johansson, P. Delsing, and E. Solano, *Dynamical Casimir effect entangles artificial atoms*. Physical Review Letters **113**, 093602 (2014).
- [29] D. Z. Rossatto, S. Felicetti, H. Eneriz, E. Rico, M. Sanz, and E. Solano, *Entangling polaritons via dynamical Casimir effect in circuit quantum electrodynamics*. Physical Review B **93**, 094514 (2016).
- [30] M. Sanz, W. Wieczorek, S. Gröblacher, and E. Solano, *Electromechanical Casimir effect*. Quantum **2**, 91 (2018).
- [31] A. Mezzacapo, M. Sanz, L. Lamata, I. L. Egusquiza, S. Succi, and E. Solano, *Quantum simulator for transport phenomena in fluid flows*. Scientific Reports **5**, 13153 (2015).
- [32] J. Argüello-Luengo, A. Gonzalez-Tudela, T. Shi, P. Zoller, and J. I. Cirac, *Analogous quantum chemistry simulation*. Nature **574**, 215 (2019).
- [33] W. Hofstetter and T. Qin, *Quantum simulation of strongly correlated condensed matter systems*. Journal of Physics B: Atomic, Molecular and Optical Physics **51**, 082001 (2018).
- [34] F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, R. Biswas, S. Boixo, F. G. S. L. Brandao, D. A. Buell, B. Burkett, Y. Chen, Z. Chen, B. Chiaro, R. Collins, W. Courtney, A. Dunsworth, E. Farhi, B. Foxen, A. Fowler, C. Gidney, M. Giustina, R. Graff, K. Guerin, S. Habegger, M. P. Harrigan, M. J. Hartmann, A. Ho, M. Hoffmann, T. Huang, T. S. Humble, S. V. Isakov, E. Jeffrey, Z. Jiang, D. Kafri, K. Kechedzhi, J. Kelly, P. V. Klimov, S. Knysh, A. Korotkov, F. Kostritsa, D. Landhuis, M. Lindmark, E. Lucero, D. Lyakh, S.

- Mandrà, J. R. McClean, M. McEwen, A. Megrant, X. Mi, K. Michielsen, M. Mohseni, J. Mutus, O. Naaman, M. Neeley, C. Neill, M. Yuezhen Niu, E. Ostby, A. Petukhov, J. C. Platt, C. Quintana, E. G. Rieffel, P. Roushan, N. C. Rubin, D. Sank, K. J. Satzinger, V. Smelyanskiy, K. J. Sung, M. D. Trevithick, A. Vainsencher, B. Villalonga, T. White, Z. J. Yao, P. Yeh, A. Zalcman, H. Neven, and J. M. Martinis, *Quantum supremacy using a programmable superconducting processor*. *Nature* **574**, 505 (2019).
- [35] Y. Wu, W. S. Bao, S. Cao, F. Chen, M. C. Chen, X. Chen, T. H. Chung, H. Deng, Y. Du, D. Fan, M. Gong, C. Guo, C. Guo, S. Guo, L. Han, L. Hong, H. L. Huang, Y. H. Huo, L. Li, N. Li, S. Li, Y. Li, F. Liang, C. Lin, J. Lin, H. Qian, D. Qiao, H. Rong, H. Su, L. Sun, L. Wang, S. Wang, D. Wu, Y. Xu, K. Yan, W. Yang, Y. Yang, Y. Ye, J. Yin, C. Ying, J. Yu, C. Zha, C. Zhang, H. Zhang, K. Zhang, Y. Zhang, H. Zhao, Y. Zhao, L. Zhou, Q. Zhu, C. Y. Lu, C. Z. Peng, X. Zhu, and J. W. Pan, *Strong quantum computational advantage using a superconducting quantum processor*. *Physical Review Letters* **127**, 180501 (2021).
- [36] Q. Zhu, S. Cao, F. Chen, M. C. Chen, X. Chen, T. H. Chung, H. Deng, Y. Du, D. Fan, M. Gong, C. Guo, C. Guo, S. Guo, L. Han, L. Hong, H. L. Huang, Y. H. Huo, L. Li, N. Li, S. Li, Y. Li, F. Liang, C. Lin, J. Lin, H. Qian, D. Qiao, H. Rong, H. Su, L. Sun, L. Wang, S. Wang, D. Wu, Y. Wu, Y. Xu, K. Yan, W. Yang, Y. Yang, Y. Ye, J. Yin, C. Ying, J. Yu, C. Zha, C. Zhang, H. Zhang, K. Zhang, Y. Zhang, H. Zhao, Y. Zhao, L. Zhou, C. Y. Lu, C. Z. Peng, X. Zhu, and J. W. Pan, *Quantum computational advantage via 60-qubit 24 cycle random circuit sampling*. *Science Bulletin (Beijing)* **67**, 240 (2022).
- [37] H. S. Zhong, H. Wang, Y. H. Deng, M. C. Chen, L. C. Peng, Y. H. Luo, J. Qin, D. Wu, X. Ding, Y. Hu, P. Hu, X. Y. Yang, W. J. Zhang, H. Li, Y. Li, X. Jiang, L. Gan, G. Yang, L. You, Z. Wang, L. Li, N. L. Liu, C. Y. Lu, and J. W. Pan, *Quantum computational advantage using photons*. *Science* **370**, 1460 (2020).
- [38] H. S. Zhong, Y. H. Deng, J. Qin, H. Wang, M. C. Chen, L. C. Peng, Y. H. Luo, D. Wu, S. Q. Gong, H. Su, Y. Hu, P. Hu, X. Y. Yang, W. J. Zhang, H. Li, Y. Li, X. Jiang, L. Gan, G. Yang, L. You, Z. Wang, L. Li, N. L. Liu, J. J. Renema, C. Y. Lu, and J. W. Pan, *Phase-programmable Gaussian boson sampling using stimulated squeezed light*. *Physical Review Letters* **127**, 180502 (2021).
- [39] L. S. Madsen, F. Laudenbach, M. F. Askarani, F. Rortais, T. Vincent, J. F. F. Bulmer, F. M. Miatto, L. Neuhaus, L. G. Helt, M. J. Collins, A. E. Lita, T. Gerrits, S. W. Nam, V. D. Vaidya, M. Menotti, I. Dhand, Z. Vernon, N. Quesada, and J. Lavoie, *Quantum computational advantage with a programmable photonic processor*. *Nature* **606**, 75 (2022).
- [40] F. Albarrán-Arriagada, J. C. Retamal, E. Solano, and L. Lamata, *Measurement-based adaptation protocol with quantum reinforcement learning*. *Physical Review A* **98**, 042315 (2018).

-
- [41] J. Olivares-Sánchez, J. Casanova, E. Solano, and L. Lamata, *Measurement-based adaptation protocol with quantum reinforcement learning in a Rigetti quantum computer*. Quantum Reports **2**, 293 (2020).
- [42] A. Martin, B. Candelas, Á. Rodríguez-Rozas, J. D. Martín-Guerrero, X. Chen, L. Lamata, R. Orús, E. Solano, and M. Sanz, *Towards Pricing Financial Derivatives with an IBM Quantum Computer*. Physical Review Research **3**, 013167 (2021).
- [43] Y. Ding, J. Gonzalez-Conde, L. Lamata, J. D. Martín-Guerrero, E. Lizaso, S. Mugel, X. Chen, R. Orús, E. Solano, and M. Sanz, *Towards Prediction of Financial Crashes with a D-Wave Quantum Computer*. Entropy **25**, 323 (2023).
- [44] J. Gonzalez-Conde, Á. Rodríguez-Rozas, E. Solano, and M. Sanz, *Simulating option price dynamics with exponential quantum speedup*. arxiv: 2101.04023 (2021).
- [45] R. Sweke, M. Sanz, I. Sinayskiy, F. Petruccione, and E. Solano, *Digital quantum simulation of many-body non-Markovian dynamics*. Physical Review A **94**, 022317 (2016).
- [46] L. García-Álvarez, U. Las Heras, A. Mezzacapo, M. Sanz, E. Solano, and L. Lamata, *Quantum chemistry and charge transport in biomolecules with superconducting circuits*. Scientific Reports **6**, 27836 (2016).
- [47] N. K. Langford, R. Sagastizabal, M. Kounalakis, C. Dickel, A. Bruno, F. Luthi, D. J. Thoen, A. Endo, and L. DiCarlo, *Experimentally simulating the dynamics of quantum light and matter at deep-strong coupling*. Nature Communications **8**, 1715 (2017).
- [48] J. Preskill, *Quantum Computing in the NISQ era and beyond*. Quantum **2**, 79 (2018).
- [49] J. Preskill, *Quantum computing 40 years later*. arxiv: 2106.10522 (2021).
- [50] J. L. Dodd, M. A. Nielsen, M. J. Bremner, and R. T. Thew, *Universal quantum computation and simulation using any entangling Hamiltonian and local unitaries*. Physical Review A **65**, 040301(R) (2002).
- [51] A. Parra-Rodriguez, P. Lougovski, L. Lamata, E. Solano, and M. Sanz, *Digital-Analog Quantum Computation*. Physical Review A **101**, 022305 (2020).
- [52] A. Martin, L. Lamata, E. Solano, and M. Sanz, *Digital-analog quantum algorithm for the quantum Fourier transform*. Physical Review Research **2**, 013012 (2020).
- [53] P. García-Molina, A. Martin, M. Garcia de Andoin, and M. Sanz, *Noise in Digital and Digital-Analog Quantum Computation*. arxiv: 2107.12969 (2021).
- [54] A. Martin, R. Ibarrondo, and M. Sanz, *Digital-analog co-design of the Harrow-Hassidim-Lloyd algorithm*. arxiv: 2207.13528 (2022).

- [55] D. Headley, T. Müller, A. Martin, E. Solano, M. Sanz, and F. K. Wilhelm, *Approximating the Quantum Approximate Optimisation Algorithm*. Physical Review A **106**, 042446 (2022).
- [56] T. Gonzalez-Raya, R. Asensio-Perea, A. Martin, L. C. Céleri, M. Sanz, P. Lougovski, and E. F. Dumitrescu, *Digital-Analog Quantum Simulations Using The Cross-Resonance Effect*. PRX Quantum **2**, 020328 (2021).
- [57] D. V. Babukhin, A. A. Zhukov, and W. V. Pogosov, *Hybrid digital-analog simulation of many-body dynamics with superconducting qubits*. Physical Review A **101**, 052337 (2020).
- [58] M. G., H. Huang, S. Wang, C. Guo, S. Li, Y. Wu, Q. Zhu, Y. Zhao, S. Guo, H. Qian, Y. Ye, C. Zha, F. Chen, C. Ying, J. Yu, D. Fan, D. Wu, Hong Su, H. Deng, H. Rong, K. Zhang, S. Cao, J. Lin, Y. Xu, L. Sun, C. Guo, N. Li, F. Liang, A. Sakurai, K. Nemoto, W. J. Munro, Y. Huo, C. Lu, C. Peng, X. Zhu, and J. Pan, *Quantum Neuronal Sensing of Quantum Many-Body States on a 61-Qubit Programmable Superconducting Processor*. arxiv: 2201.05957 (2022).
- [59] A. Michel, S. Grijalva, L. Henriët, C. Domain, and A. Browaeys, *A blueprint for a Digital-Analog Variational Quantum Eigensolver using Rydberg atom arrays*. arxiv: 2301.06453 (2023).
- [60] S. Notarnicola, A. Elben, T. Lahaye, A. Browaeys, S. Montangero, and B. Vermersch, *A randomized measurement toolbox for Rydberg quantum technologies*. arxiv:2112.11046 (2021).
- [61] J. Yu, J. C. Retamal, M. Sanz, E. Solano, and F. Albarrán-Arriagada, *Superconducting Circuit Architecture for Digital-Analog Quantum Computing*. EPJ Quantum Technologies **9**, 9 (2022).
- [62] L. C. Céleri, D. Huerga, F. Albarrán-Arriagada, E. Solano, M. Garcia de Andoin, and M. Sanz, *Digital-analog quantum simulation of fermionic models*. arxiv: 2103.15689 (2021).
- [63] Ll. Masanes, G. Vidal and J. I. Latorre, *Time-optimal hamiltonian simulation and gate synthesis using homogeneous local unitaries*. Quantum Information and Computation **2**, 285 (2002).
- [64] C. H. Bennett, J. I. Cirac, M. S. Leifer, D. W. Leung, N. Linden, S. Popescu, and G. Vidal, *Optimal simulation of two-qubit Hamiltonians using general local operations*. Physical Review A **66**, 012305 (2002).
- [65] E. Jané, G. Vidal, W. Dür, P. Zoller, and J. I. Cirac, *Simulation of quantum dynamics with quantum optical systems*. Quantum Information and Computation **3**, 15 (2003).
- [66] D. Porras and J. I. Cirac, *Effective quantum spin systems with trapped ions*. Physical Review Letters **92**, 207901 (2004).

-
- [67] R. H. Lehmborg, *Radiation from an N-Atom System. I. General Formalism*. Physical Review A **2**, 883 (1970).
- [68] R. H. Lehmborg, *Radiation from an N-Atom System. II. Spontaneous Emission from a Pair of Atoms*. Physical Review A **2**, 889 (1970).
- [69] A. González-Tudela, C.-L. Hung, D. E. Chang, J. I. Cirac, and H. J. Kimble, *Subwavelength vacuum lattices and atom–atom interactions in two-dimensional photonic crystals*. Nature Photonics **9**, 320 (2015).
- [70] F. Solgun, D. P. DiVincenzo, J. M. Gambetta, *Simple Impedance Response Formulas for the Dispersive Interaction Rates in the Effective Hamiltonians of Low Anharmonicity Superconducting Qubits*. IEEE Transactions on Microwave Theory and Techniques **67**, 3 (2019).
- [71] M. Van den Nest, W. Dür, and H. J. Briegel, *Completeness of the Classical 2D Ising Model and Universal Quantum Computation*. Physical Review Letters **100**, 110501 (2008).
- [72] A. Galicia, B. Ramon, E. Solano, and M. Sanz, *Enhanced connectivity of quantum hardware with digital-analog control*. Physical Review Research **2**, 033103 (2020).
- [73] L. Lapidus and R. Luus, *Optimal Control of Engineering Processes*. (Blaisdell Publishing, Waltham, MA, 1967).
- [74] Y. Takahashi, M. J. Rabins, and D. M. Auslander, *Control and Dynamic Systems*. (Addison-Wesley, Reading, MA, 1970).
- [75] L. Viola and S. Lloyd, *Dynamical suppression of decoherence in two-state quantum systems*. Physical Review A **58**, 2733 (1998).
- [76] J. J. L. Morton, A. M. Tyryshkin, A. Ardavan, S. C. Benjamin, K. Porfyarakis, S. A. Lyon, and G. A. D. Briggs, *Bang–bang control of fullerene qubits using ultrafast phase gates*. Nature Physics **2**, 40 (2006).
- [77] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, Cambridge, 2000).
- [78] K. Kraus, *States, Effects and Operations: Fundamental Notions of Quantum Theory*. (Springer Verlag, 1983).
- [79] J. Preskill, *Lecture Notes for Ph219/CS219: Quantum Computation Chapter 3* (California Institute of Technology, 2018).
- [80] S. Endo, S. C. Benjamin, and Y. Li, *Practical Quantum Error Mitigation for Near-Future Applications*. Physical Review X **8**, 031027 (2018).

- [81] A. Kandala, K. Temme, A. D. Córcoles, A. Mezzacapo, J. M. Chow, and J. M. Gambetta, *Error mitigation extends the computational reach of noisy quantum processor*. Nature **567**, 491 (2019). [c](#)
- [82] S. Endo, Z. Cai, S. C. Benjamin, and X. Yuan, *Hybrid Quantum-Classical Algorithms and Quantum Error Mitigation*. Journal of the Physical Society of Japan **90**, 032001 (2021).
- [83] S. Endo, Q. Zhao, Y. Li, S. Benjamin, and X. Yuan, *Mitigating algorithmic errors in a Hamiltonian simulation*. Physical Review X **99**, 012334 Physical Review A (2019).
- [84] J. Sun, X. Yuan, T. Tsunoda, V. Vedral, S. C. Benjamin, and S. Endo, *Mitigating Realistic Noise in Practical Noisy Intermediate-Scale Quantum Devices*. Physical Review Applied **15**, 034026 (2021).
- [85] H. Hakoshima, Y. Matsuzaki, and S. Endo, *Relationship between costs for quantum error mitigation and non-Markovian measures*. Physical Review A **103**, 012611 (2021).
- [86] Y. Li and S. C. Benjamin, *Efficient Variational Quantum Simulator Incorporating Active Error Minimization*. Physical Review X **7**, 021050 (2017).
- [87] K. Temme, S. Bravyi, and J. M. Gambetta, *Error Mitigation for Short-Depth Quantum Circuits*. Physical Review Letters **119**, 180509 (2017).
- [88] P. Czarnik, A. Arrasmith, P. J. Coles, and L. Cincio, *Error mitigation with Clifford quantum-circuit data*. Quantum **5**, 592 (2021).
- [89] A. W. Harrow, A. Hassidim, and S. Lloyd, *Quantum Algorithm for Linear Systems of Equations*. Physical Review Letters **103**, 150502 (2009).
- [90] S. Aaronson, *Read the fine print*. Nature Physics **11**, 291–293 (2015).
- [91] L. Grover, and T. Rudolph, *Creating superpositions that correspond to efficiently integrable probability distributions*. arxiv: 0208112 (2002).
- [92] X. D. Cai, C. Weedbrook, Z. E. Su, M. C. Chen, M. Gu, M. J. Zhu, L. Li, N. L. Liu, C. Y. Lu, and J. W. Pan, *Experimental quantum computing to solve systems of linear equations*. Physical Review Letters **110**, 230501 (2013).
- [93] J. Pan, Y. Cao, X. Yao, Z. Li, Ch. Ju, H. Chen, X. Peng, S. Kais, and J. Du, *Experimental realization of quantum algorithm for solving linear system of equations*. Physical Review A **889**, 022313 (2014).
- [94] S. Barz, I. Kassal, M. Ringbauer, Y. O. Lipp, B. Dakić, A. Aspuru-Guzik, and P. Walther, *A two-qubit photonic quantum processor and its application to solving systems of linear equations*. Scientific Reports **4**, 6115 (2014).

-
- [95] Y. Zheng, C. Song, M. Chen, B. Xia, W. Liu, Q. Guo, L. Zhang, D. Xu, H. Deng, K. Huang, Y. Wu, Z. Yan, D. Zheng, L. Lu, J. Pan, H. Wang, C. Lu, and X. Zhu, *Solving Systems of Linear Equations with a Superconducting Quantum Processor*. Physical Review Letters **118**, 210504 (2017).
- [96] Y- Lee, J. Joo, and S. Lee, *Hybrid quantum linear equation algorithm and its experimental test on IBM Quantum Experience*. Scientific Reports **9** 4778 (2019).
- [97] M. Möttönen, J. J. Vartiainen, V. Bergholm, and M. M. Salomaa, *Quantum circuits for general multi-qubit gates*. Physical Review Letters **96**, 130502 (2004).
- [98] E. Farhi, J. Goldstone, and S. Gutmann, *A Quantum Approximate Optimization Algorithm*. arxiv: 1411.4028 (2014).
- [99] E. Farhi, J. Goldstone, and S. Gutmann, *A Quantum Approximate Optimization Algorithm Applied to a Bounded Occurrence Constraint Problem*. arxiv: 1412.6062 (2014).
- [100] B. Barak, A. Moitra, R. O'Donnell, P. Raghavendra, O. Regev, D. Steurer, L. Trevisan, A. Vijayaraghavan, D. Witmer, and J. Wright, *Beating the random assignment on constraint satisfaction problems of bounded degree*. arxiv: 1505.03424 (2015).
- [101] Z. Jiang, E. G. Rieffel, and Z. Wang, *Near-optimal quantum circuit for Grover's unstructured search using a transverse field*. Physical Review A **95**, 062317 (2017).
- [102] S. Hadfield, Z. Wang, B. O'Gorman, E. G. Rieffel, D. Venturelli, and R. Biswas, *From the Quantum Approximate Optimization Algorithm to a Quantum Alternating Operator Ansatz*. Algorithms **12**, 34 (2019).
- [103] S. Hadfield, *Quantum Algorithms for Scientific Computing and Approximate Optimization*. arxiv:1805.03265 (2018).
- [104] C. J. Ballance, T. P. Harty, N. M. Linke, M. A. Sepiol, and D. M. Lucas, *High-Fidelity Quantum Logic Gates Using Trapped-Ion Hyperfine Qubits*. Physical Review Letters **117**, 060504 (2016).
- [105] N. M. Linke, D. Maslov, M. Roetteler, S. Debnath, C. Figgatt, K. A. Landsman, K. Wright, and C. Monroe, *Experimental comparison of two quantum computing architectures*. Proceedings of the National Academy of Sciences USA **114**, 3305 (2017).
- [106] A. Pearson, A. Mishra, I. Hen, and D. A. Lidar, *Analog errors in quantum annealing: doom and hope*. Npj Quantum Information **5**, 107 (2019).
- [107] A. Callison, N. Chancellor, F. Mintert, and V. Kendon, *Finding spin glass ground states using quantum walks*. New Journal of Physics **21**, 123022 (2019).

- [108] J. G. Morley, N. Chancellor, S. Bose, and V. Kendon, *Quantum search with hybrid adiabatic–quantum-walk algorithms and realistic noise*. Physical Review A **99**, 022339 (2019).
- [109] S. Muthukrishnan, T. Albash, and D. A. Lidar, *When Diabatic Trumps Adiabatic in Quantum Optimization*. arxiv: 1505.01249 (2015)
- [110] J. R. McClean, J. Romero, R. Babbush, and A. Aspuru-Guzik, *The theory of variational hybrid quantum-classical algorithms*. New Journal of Physics **18**, 023023 (2016).
- [111] J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush, and H. Neven, *Barren plateaus in quantum neural network training landscapes*. Nature Communications **9**, 4812 (2018).
- [112] W. Dür and H. J. Briegel, *Stability of Macroscopic Entanglement under Decoherence*. Physical Review Letters **92**, 180403 (2004).
- [113] J. Koch, T. M. Yu, J. Gambetta, A. A. Houck, D. I. Schuster, J. Majer, A. Blais, M. H. Devoret, S. M. Girvin, and R. J. Schoelkopf, *Charge-insensitive qubit design derived from the Cooper pair box*. Physical Review A **76**, 042319 (2007).
- [114] D. C. McKay, C. J. Wood, S. Sheldon, J. M. Chow, and J. M. Gambetta, *Efficient Z gates for quantum computing*. Physical Review A **96**, 022330 (2017).
- [115] G. S. Paraoanu, *Microwave-induced coupling of superconducting qubits*. Physical Review B **74**, 140504 (2006).
- [116] C. Rigetti and M. Devoret, *Fully microwave-tunable universal gates in superconducting qubits with linear couplings and fixed transition frequencies*. Physical Review B **81**, 134507 (2010).
- [117] J. M. Chow, A. D. Córcoles, J. M. Gambetta, C. Rigetti, B. R. Johnson, J. A. Smolin, J. R. Rozen, G. A. Keefe, M. B. Rothwell, M. B. Ketchen, and M. Steffen, *Simple All-Microwave Entangling Gate for Fixed-Frequency Superconducting Qubits*. Physical Review Letters **107**, 080502 (2011).
- [118] A. Shabani, M. Mohseni, S. Lloyd, R. L. Kosut, and H. Rabitz, *Estimation of many-body quantum Hamiltonians via compressive sensing*. Physical Review A **84**, 012107 (2011).
- [119] M. P. da Silva, O. Landon-Cardinal, and D. Poulin, *Practical Characterization of Quantum Devices without Tomography*. Physical Review Letters **107**, 210404 (2011).
- [120] X.-L. Qi and D. Ranard, *Determining a local Hamiltonian from a single eigenstate*. Quantum **3**, 159 (2019).

- [121] E. Bairey, I. Arad, and N. H. Lindner, *Learning a Local Hamiltonian from Local Measurements*. Physical Review Letters **122**, 020504 (2019).
- [122] E. Bairey, C. Guo, D. Poletti, N. H. Lindner, and I. Arad, *Learning the dynamics of open quantum systems from their steady states*. New Journal of Physics **22**, 032001 (2020).
- [123] E. F. Dumitrescu and P. Lougovski, *Hamiltonian assignment for open quantum systems*. Physical Review Research **2**, 033251 (2020).
- [124] C. E. Granade, C. Ferrie, N. Wiebe, and D. G. Cory, *Robust online hamiltonian learning*. New Journal of Physics **14**, 103013 (2012).
- [125] S. Sheldon, E. Magesan, J. M. Chow, and J. M. Gambetta, *Procedure for systematically tuning up cross-talk in the cross-resonance gate*. Physical Review A **93**, 060302 (2016).
- [126] E. Magesan and J. M. Gambetta, *Effective Hamiltonian models of the cross-resonance gate*. Physical Review A **101**, 052308 (2020).
- [127] M. Malekakhlagh, E. Magesan, and D. C. McKay, *First-principles analysis of cross-resonance gate operation*. Physical Review A **102**, 042605 (2020).
- [128] J. Ku, X. Xu, M. Brink, D. C. McKay, J. B. Hertzberg, M. H. Ansari, and B. L. T. Plourde, *Suppression of Unwanted ZZ Interactions in a Hybrid Two-Qubit System*. Physical Review Letters **125**, 200504 (2020).
- [129] A. Kandala, K. X. Wei, S. Srinivasan, E. Magesan, S. Carnevale, G. A. Keefe, D. Klaus, O. Dial, and D. C. McKay, *Demonstration of a High-Fidelity CNOT for Fixed-Frequency Transmons with Engineered ZZ Suppression*. Physical Review Letters **127**, 130501 (2021).
- [130] M. C. Tran, S.-K. Chu, Y. Su, A. M. Childs, and A. V. Gorshkov, *Destructive Error Interference in Product-Formula Lattice Simulation*. Physical Review Letters **124**, 220502 (2020).
- [131] D. W. Berry, A. M. Childs, R. Cleve, R. Kothari, and R. D. Somma, *Simulating Hamiltonian Dynamics with a Truncated Taylor Series*. Physical Review Letters **114**, 090502 (2015).