

Swap Network Augmented Ansätze on Arbitrary Connectivity

Teodor Parella-Dilmé¹, Jakob S. Kottmann^{2,3}, and Antonio Acín^{1,4}

¹ICFO - Institut de Ciències Fotoniques, The Barcelona Institute of Science and Technology, Av. Carl Friedrich Gauss 3, 08860 Castelldefels (Barcelona), Spain

²Institute for Computer Science, University of Augsburg, Germany

³Center for Advanced Analytics and Predictive Sciences, University of Augsburg, Germany

⁴ICREA-Institució Catalana de Recerca i Estudis Avançats, Lluís Companys 23, 08010 Barcelona, Spain

Efficient parametrizations of quantum states are essential for trainable hybrid classical-quantum algorithms. A key challenge in their design consists in adapting to the available qubit connectivity of the quantum processor, which limits the capacity to generate correlations between distant qubits in a resource-efficient and trainable manner. In this work we first introduce an algorithm that optimizes qubit routing for arbitrary connectivity graphs, resulting in a swap network that enables direct interactions between any pair of qubits. We then propose a co-design of circuit layers and qubit routing by embedding the derived swap networks within layered, connectivity-aware ansätze. This construction significantly improves the trainability of the ansatz, leading to enhanced performance with reduced resources. We showcase these improvements through ground-state simulations of strongly correlated systems, including spin-glass and molecular electronic structure models. Across exemplified connectivities, the swap-enhanced ansatz consistently achieves lower energy errors using fewer entangling gates, shallower circuits, and fewer parameters than standard layered-structured baselines. Our results indicate that swap network augmented ansätze provide enhanced trainability and resource-efficient design to capture complex correlations on devices with

constrained qubit connectivity.

1 Introduction

In variational quantum algorithms (VQAs) a parametrized quantum circuit (PQC) is optimized through a classical routine to solve optimization tasks [1, 2]. The structure and initialization of an ad-hoc PQC play a crucial role in the optimization and overall algorithm performance, as they influence both the accessible state space and the optimization landscape [3, 4]. Effective ansatz design is best guided by the problem’s structure (when available) and the limitations of the available hardware. For instance, physically-inspired ansätze exploit known symmetries to reduce the solution space [5], while hardware-efficient ansätze (HEA) leverage native gates and direct qubit connectivity to minimize resource requirements [6].

Regarding hardware limitations, constrained qubit connectivity represents a major challenge: two-qubit gates can only be applied between neighbouring qubits in the device’s connectivity. A common practice is to insert SWAP gates via routing strategies [7, 8, 9, 10] to bring qubits together whenever a two-qubit gate needs to be applied. However each SWAP (implemented in practice via multiple entangling gates) adds an overhead in noise and circuit depth [11]. In contrast, the aforementioned HEA are another workaround to the constrained connectivity problem by using ansätze that only entangle adjacent qubits in the connectivity using native gates. While this avoids SWAP gates, it severely limits the spread of correlations across the processor, therefore compromising trainability when

Teodor Parella-Dilmé: teodor.parella@icfo.eu

Jakob S. Kottmann: jakob.kottmann@uni-a.de

Antonio Acín: antonio.acin@icfo.eu

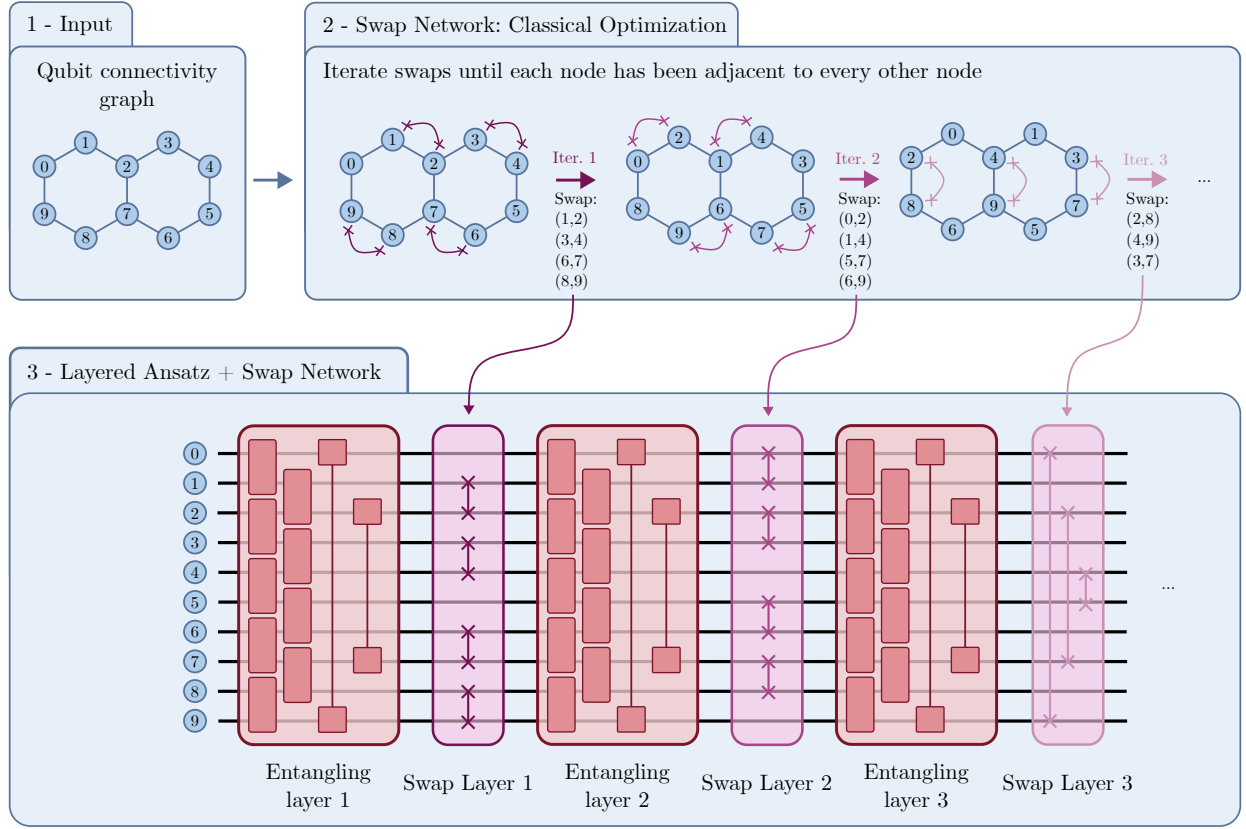


Figure 1: Diagram of the proposed algorithm for designing a layered ansatz embedded within a swap network with optimized qubit routing in arbitrary connectivity. **1)** The available qubit connectivity graph is taken as input. **2)** A classical optimization algorithm designs an optimized swap network for the given connectivity constraints. The optimized sequence of swaps ensures that by the end of the process each node has been adjacent to every other node at least once. **3)** The layered ansatz is embedded within the swap network, enabling direct information exchange between all qubits on the constrained connectivity hardware.

the target state has long-range entanglement [12].

An alternative strategy to overcome connectivity limitations is the use of swap networks in a linear qubit connectivity [13, 14, 15]. These are predetermined sequence of SWAP gates arranged in layers, designed so that after the sequence every qubit has effectively moved next to every other qubit at least once, enabling all-to-all connectivity. The network systematically permutes the qubits across the device so that any logical pair can eventually interact, without needing on-the-fly routing decisions. The fixed swap schedule deterministically guarantees all desired interactions occur, offering potential applicability in unstructured or repetitive circuits, such as Trotterized dynamics [16] or alternating-layer algorithms like the Quantum Approximate Optimization Algorithm (QAOA) [17, 18].

In this work, we first introduce an algorithm for generating optimized swap networks for arbitrary qubit connectivity, which is available at [19]. We then propose integrating the optimized swap networks with hardware-adapted layer-structured ansätze made of repeated blocks of local entanglers [Fig.1]. The net effect is a hardware-aware ansatz that effectively enables all-to-all connectivity with reduced SWAP overhead compared to dynamical compilers such as SABRE [7, 20]. Our procedure avoids the use of measurement-intensive Adaptive Derivative-Assembled Pseudo-Trotter ansatz (ADAPT) methods [21, 22, 23], therefore avoiding metaoptimization during the ansatz construction.

We demonstrate through variational ground-state simulations of highly correlated systems that the swap network enhanced ansatz significantly improves both trainability and performance, all

while maintaining low resource overhead. These gains are showcased across benchmark problems, including quantum spin-glass Hamiltonians and the electronic structure of the strongly correlated p-benzyne biradical. For a fixed resource budget of gate count, circuit depth, or number of parameters, the swap network enhanced ansatz consistently achieves lower energy errors and improved convergence compared to the HEA. This translates into higher noise resilience: at the same circuit depth or entangling gate count, and therefore comparable accumulated noise, it achieves lower errors and more reliable convergence. Across various connectivity graphs, we find that long-range correlations are captured more efficiently, underscoring the advantage of incorporating optimized swap networks in the ansatz design.

2 Preliminaries

2.1 Circuit Compilation

Circuit compilation is the process of transforming an abstract quantum circuit, defined as an ordered sequence of ideal gates,

$$U(\boldsymbol{\theta}) = \prod_{k=1}^K u_k(\theta_k), \quad \boldsymbol{\theta} \in \mathbb{R}^K,$$

into a hardware compatible circuit $U'(\boldsymbol{\theta})$ that can be executed on a specific quantum processor. Here u_k denotes the k th circuit gate, which may be a one- or two-qubit operation, and may be either fixed or parametrized. This transformation is necessary because quantum hardware supports only a limited set of native gates and allows two-qubit operations only between certain pairs of connected qubits. The compilation process typically comprises two main tasks (may include additional ones):

1. **Gate decomposition** translates or approximates all high-level operations into sequences of native gates, ensuring that every operation conforms to the hardware's supported gate set.
2. **Qubit routing** adapts the logical circuit to the physical qubit layout by initially mapping logical qubits onto physical qubits and then routing them by inserting swap gates

where necessary, so that two-qubit operations occur only between adjacent qubits in the connectivity.

Here, we consider gates in the abstract circuit U to be native, and take a closer look at the qubit routing step. Although essential, qubit routing imposes a significant burden on the resulting circuit by introducing numerous swap gates. These gates decompose into multiple error-prone two-qubit operations within the native gate set (Appendix A). The additional two-qubit operations not only increase circuit depth but also represent a primary source of noise, ultimately degrading computational fidelity on near-term devices. Therefore, minimizing swap overhead is essential for enhancing the efficiency of quantum simulations, reducing both noise and circuit depth to improve overall performance.

2.2 Hardware-Efficient Ansätze

To mitigate the detrimental effects of extensive swap overhead, a promising strategy is to design circuits that align with the hardware's native constraints. HEA embody this approach by directly using the available native gate set and restricting two-qubit operations to adjacent qubits in the hardware connectivity [6]. The HEA is typically implemented in an alternating-layer construction, where layers of gates are arranged in a structured, repeating pattern. By doing so, HEA eliminate the need for circuit compilation and the associated swap overhead, preventing additional swaps that would otherwise increase circuit depth and noise.

This improvement, however, comes at the cost of sacrificing all-to-all connectivity. Since two-qubit gates can only be applied to adjacent qubits, establishing correlations across distant regions of the processor requires a chain of intermediate interactions that relay information in the causal cone. Capturing long-range correlations in such architectures is particularly challenging because many parameters must simultaneously align to effectively correlate distant qubits, an effect that can lead to local minima or barren plateaus in the HEA cost landscape [12, 24]. This intricate interdependence complicates the optimization and can severely limit ansatz trainability in the presence of long-range correlations [11]. In

particular, Ref.[25] establishes that the HEA can remain trainable when the input states obey an area law of entanglement; conversely, for inputs with long-range correlations satisfying a volume law, the loss values concentrate exponentially around a trivial value, rendering the HEA effectively untrainable and inexpressible.

Despite these challenges, tailoring ansatz design to the underlying hardware remains a crucial strategy for minimizing circuit depth and mitigating noise, thereby enabling practical ansätze for quantum simulation.

2.3 Swap Networks

An alternative resource efficient approach that preserves effective all to all connectivity while minimizing compilation induced swap overhead is the use of swap networks [13, 14, 15]. Swap networks are predetermined sequences of swap gates that systematically reconfigure the qubits throughout the circuit execution. Unlike ad hoc routing strategies that insert swaps on a case-by-case basis in the compilation step whenever non-adjacent qubits need to interact, a swap network adheres to a fixed routing schedule that guarantees every required qubit interaction will eventually occur. This instance-agnostic strategy is especially valuable when the circuit exhibits an unstructured sequence of gates, as is common in Trotterized simulations [16] or the alternating layers of QAOA [14]. By optimally layering swap operations, swap networks completely avoid the circuit compilation step by providing a robust, hardware-adapted method to achieve the desired entanglement among qubits with minimal swap overhead.

In the following, we optimize the design of swap network protocols for arbitrary hardware graphs, enabling effective all-to-all connectivity ansätze on modern processors with any connectivity.

3 Swap Network Optimization For Arbitrary Qubit Connectivity

3.1 Overview

A diagram illustrating the proposed algorithm, featuring the construction of a hardware-friendly

layered ansatz embedded in an optimized swap network, is shown in Fig.1. The algorithm takes the qubit connectivity graph as input. A classical optimization process determines how to construct a circuit with a reduced number of swap layers that ensure that every qubit pair is connected at least once during the routing. The resulting swap layers from the classical optimization define the swap network, which is then interwoven between the layers of the hardware-friendly layered ansatz. Ultimately, a swap enhanced ansatz with optimized routing sequence is obtained, enabling direct information exchange between all pairs of qubits in the system. In the following, we detail each subroutine of the algorithm.

3.2 Qubit Display as a Labelled Graph

We define a labelled and undirected graph $\mathcal{G} = (V, E, \mathcal{L})$, where:

- $V = \{v_i\}_{i=1}^n$ is a set of n labelled vertices.
- $E \subseteq \{(x, y) \in V \times V \mid x \neq y\}$ is a subset of all the pairs of different vertices.
- $\mathcal{L} : V \rightarrow L$ denotes a bijective labelling function that assigns each vertex $v_i \in V$ a label $\mathcal{L}(v_i) \in L$, with the label set $L = \{1, 2, \dots, n\}$.

The set V represents the physical sites of the hardware, while the set E defines the connections between these sites, representing the possible entangling interactions. Furthermore, $\mathcal{L}(v_i)$ denotes the label assigned to the qubit located at each physical site v_i .

3.3 Qubit swapping as Graph Relabelling Functions

We now introduce a graph operation simulating the behaviour of a swap gate, corresponding to a label-swapping function $\mathcal{S}(\mathcal{G}, e_{i,j}) = \mathcal{G}'$, $e_{i,j} = (v_i, v_j) \in E$ which swaps the labels of the connected vertices v_i and v_j in a graph $\mathcal{G}$. The resulting graph $\mathcal{G}' = (V, E, \mathcal{L}')$ is characterized by the updated labelling:

- $\mathcal{L}'(v_i) = \mathcal{L}(v_j)$,
- $\mathcal{L}'(v_j) = \mathcal{L}(v_i)$,
- $\mathcal{L}'(v_k) = \mathcal{L}(v_k)$ for $k \neq i, j$.

To emulate a depth-1 layer of swap gate operations, we define $\mathcal{R}^1$ as a layer of swap-relabelling functions of depth 1. This function consists of a concatenation of commuting label-swapping functions $\{\mathcal{S}_i\}_{i=1}^m$, each acting on different vertices. Specifically:

$$\mathcal{R}^1 = \mathcal{S}_m \circ \mathcal{S}_{m-1} \circ \dots \circ \mathcal{S}_2 \circ \mathcal{S}_1,$$

Each $\mathcal{S}_i$ represents a distinct label-swapping operation, ensuring that two different operations do not act on the same vertex. This means each node is either swapped once with a neighbour or remains unchanged (only one swap per node).

Ultimately, a number k of different swap-relabelling layer functions of depth 1, can be concatenated to yield a swap-relabelling multilayer function of depth k :

$$\mathcal{R}^k = \mathcal{R}_k^1 \circ \mathcal{R}_{k-1}^1 \circ \dots \circ \mathcal{R}_2^1 \circ \mathcal{R}_1^1.$$

Such a multilayer function simulates an arbitrary swap layer of depth k within the available connectivity. The motivation behind $\mathcal{R}^k$ is that performing a single swap within the connectivity does not fully maximize the potential for creating new interactions before applying an entangling gate to neighbouring qubits. If the swap depth is limited to one, the swapped qubits will remain neighbours. Therefore, a depth of $k \geq 2$ is preferred, as it allows for a greater number of new qubit interactions to emerge.

3.4 The History Matrix

A key component of the algorithm is the *history matrix*, denoted as H . At each iteration step t , the history matrix H^t records the adjacency history of qubit interactions, tracking which pairs of qubits have already been nearest neighbours. Specifically, an entry $H_{i,j}^t = 0$ indicates that qubits i and j have already shared a direct connection at iteration $\leq t$, while $H_{i,j}^t = 1$ indicates they have not yet interacted directly. At the initial step, $H^{t=0}$ is complementary to the adjacency matrix of the labelled graph. As the algorithm progresses, successive qubit relabelling allows for new direct connections to form, and the history matrix is updated from H^t to H^{t+1} . This process continues until all entries are zero, signifying that every qubit pair has been directly connected at least once.

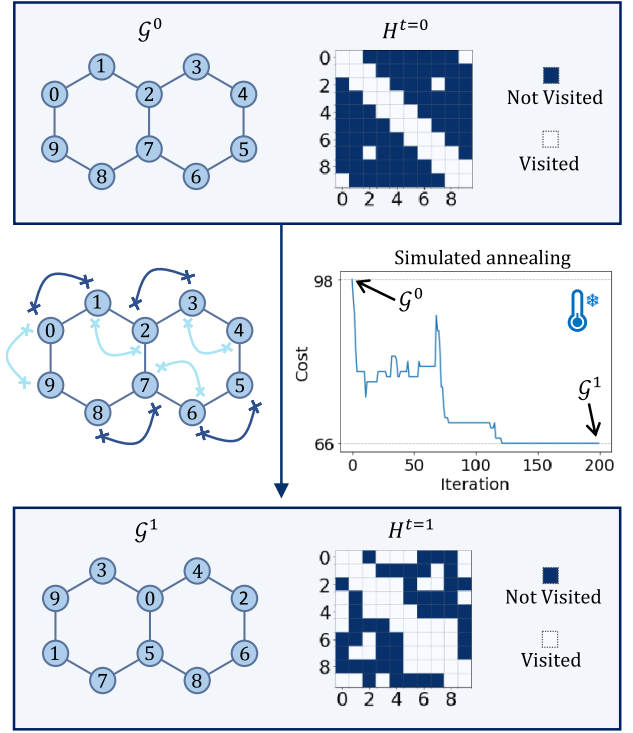


Figure 2: Exemplified algorithm step with $k = 2$, that optimizes an initial graph $\mathcal{G}_0$ into $\mathcal{G}_1$. The initial history matrix $H^{t=0}$ equals $\mathcal{G}_0$ adjacency matrix, where dark blue values ($H_{i,j}^{t=0} = 1$) indicate that indices i, j have yet to become adjacent, and white values ($H_{i,j}^{t=0} = 0$) indicate prior adjacency. A simulated annealing optimization modifies the graph by adding or removing swap gates, with up to $k = 2$ swap layers. The cost function minimization is shown at the middle, reducing from $C(\mathcal{G}_0) = 98$ to $C(\mathcal{G}_1) = 66$. The optimized $k = 2$ swap sequence morphing $\mathcal{G}_0$ to $\mathcal{G}_1$ (left) shows dark swaps in the first layer and light swaps in the second. The final graph $\mathcal{G}_1$ introduces new direct interactions, and its updated history matrix $H^{t=1}$ has fewer non visited entries, indicating reduced remaining interactions. To extract the complete swap network, this process can be repeated iteratively until all history matrix entries reach zero.

3.5 Cost Function

Given the iteration t , with associated graph $\mathcal{G}^t$ and history matrix H^t , we seek to quantify the suitability of a relabelled graph resulting from a swap relabelling function of depth k , $\mathcal{G}^{t+1} = \mathcal{R}^k(\mathcal{G}^t)$ using a cost function $\mathcal{C}$. The cost function should satisfy the two following objectives:

1. Maximize the number of new direct connections between labels.
2. Minimize the distance between labels that have yet to connect in forthcoming steps.

Let d_{ij} denote the shortest path distance between labels i and j in the graph $\mathcal{G}^t$. We can then define the following cost function on the graph $\mathcal{G}^t$ with history matrix H^t :

$$\mathcal{C}(\mathcal{G}^t, H^t) = \sum_{i,j} H_{ij}^t d_{ij}^\alpha, \quad (1)$$

where the α exponent is an hyperparameter. The minimization of this cost function addresses the two key requirements. Firstly, it maximizes the number of new direct connections by summing the terms $H_{ij} d_{ij}^\alpha$, which favours configurations where H_{ij} values shift from 1 to 0. This aspect of the function encourages the minimization of qubit pairs that have yet to interact. Secondly, the term d_{ij}^α penalizes qubit pairs that remain unconnected and are positioned far apart, thereby promoting closer arrangements within the connectivity graph to facilitate direct connections in subsequent steps.

Note that α is a tunable hyperparameter. Larger values (e.g., $\alpha = 2$) favor clustering distant qubits together rather than introducing new interactions, whereas smaller values (e.g., $\alpha = 1/2$) promote the creation of new interactions. The value of α influences the convergence of the algorithm, and its optimal choice depends on the given qubit connectivity graph. Note that α , and therefore the cost function, can be changed during the optimization. Small values help early by promoting local interactions, while larger values help later by bringing distant qubits together and speeding up convergence. In Appendix A we demonstrate how the algorithm's convergence depends on this exponent for graph instances relevant to practical electronic structure simulations.

3.6 Graph Updating

In order to propose a candidate $\mathcal{G}'$ deriving from $\mathcal{G}$, we require an updating function. In particular, $\mathcal{G}'$ should result from a given relabelling function $\mathcal{M}_t^k$ of depth k . To encode such relabelling function, we create a list of length k , where the i 'th entry of the list will encode the swapping information of the i 'th layer of the swapping relabelling function. Such information, is a subset of the edges of $\mathcal{G}$, where no label is repeated.

The graph updating chooses a random layer from 1 to k . Then with probability p adds a random edge that is not appearing in that particular layer, or with probability $1 - p$ removes a random edge from that layer.

3.7 The Algorithm

The procedure starts with the initial graph $\mathcal{G}^{t=0}$ and the corresponding history matrix $H^{t=0}$. The algorithm relabels the graph with a swap depth k , yielding candidate graphs $\mathcal{G}_{candidate}^{t+1} = \mathcal{M}_t^k(\mathcal{G}^t)$, with associated candidate history matrices $H_{candidate}^{t+1}$. The candidates are optimized through a simulating annealing protocol in order to minimize the cost function $\mathcal{C}$ in Equation 1. Multiple annealing schedules may be run in parallel to increase the optimization quality. The best candidate found along the annealing is promoted as $\mathcal{G}^{t+1}$, as well as its associated history matrix H^{t+1} . The algorithm terminates when every label has been a neighbour of every other label, i.e., when $H_{i,j} = 0$ for all i, j . The algorithm is detailed in Alg.1. A visual example of a first optimization step for $k = 2$ is shown in Fig.2.

Since the algorithm iterates until the history matrix is fully cleared, the required number of iterations scales with the graph's average degree, i.e., the mean number of edges per node:

$$\hat{D}(G) = \frac{2|E|}{|V|}. \quad (2)$$

Each qubit creates $\mathcal{O}(\hat{D}(G))$ new adjacencies per iteration on average. As each must connect to the other $N - 1$ qubits, to clear the history matrix the total number of iterations scales as

$$\mathcal{O}\left(\frac{N}{\hat{D}(G)}\right). \quad (3)$$

In practical hardware of interest, the average degree $\hat{D}(G)$ is typically $\mathcal{O}(1)$. Thus, the algorithm does not change the asymptotic scaling of SWAP-routing depth with system size, which remains linear in N . Its purpose is instead to exploit the fact that experimental devices often have connectivity larger than a simple line, $\hat{D}(G) > 2$, to reduce the constant prefactor: higher degree creates more new adjacencies per iteration on average, thereby reducing the overall

SWAP-network depth compared to the strictly linear case.

The simulated-annealing hyperparameters control the trade-off between runtime and solution quality. Since the routing/ordering task is a hard combinatorial problem, we typically seek good (not necessarily optimal) schedules. Larger computational budgets usually improve the result, while smaller budgets still provide adequate schedules with efficient scaling.

For the small quantum circuits considered in this work, the algorithm is always able to terminate after a modest computational effort and swap network depth, yielding the complete swap network where all the qubits connect. It is however expected that for much larger circuits it is harder to attain a situation where $H_{i,j} = 0$ for all i, j , and that the required swap network depth exceeds the noise-limited circuit depth that can be implemented in hardware. In those cases, a stopping criterion may be applied after a given number of iterations or connections in the history matrix (I_{max}), yielding a partial swap network.

Algorithm 1 Swap network optimization under arbitrary constrained qubit connectivity

Require: $\mathcal{G} = (V, E, \mathcal{L}) \triangleright$ Labelled connectivity graph

Require: $k \triangleright$ Swap depth

Require: $I_{max} \triangleright$ Max iterations (optional)

```

1:  $H_{i,j} \equiv \begin{cases} 0 & \text{if } (v_i, v_j) \in E \text{ or } i = j, \\ 1 & \text{otherwise.} \end{cases}$ 
2: SwapProtocol=[]
3: while  $\exists(i, j)$  such that  $H_{i,j} \neq 0 \wedge |SwapProtocol| < I_{max}$  do
4:   Optimize  $\mathcal{R}^k(\mathcal{G}), H'$ , minimizing  $\mathcal{C}(\mathcal{R}^k(\mathcal{G}), H')$ 
5:    $\mathcal{G} \leftarrow \mathcal{R}^k(\mathcal{G})$ 
6:    $H \leftarrow H'$ 
7:   Append  $\mathcal{R}^k$  to SwapProtocol
return SwapProtocol

```

3.8 Ansatz Design

Once an optimized swap network is found for a given qubit connectivity using the proposed algorithm, embedding a layered ansatz within it

becomes straightforward. The entangling layers are interleaved with the swap layers from the optimized swap network, as displayed in Fig.1.

4 Numerical Results

To evaluate how optimized swap networks enhance optimization in systems with long-range correlations that do not align with hardware connectivity, in the following we conduct a VQE simulation on highly correlated systems. All simulations were implemented in TEQUILA [26], using QULACS simulator [27].

4.1 Spin Glass Simulations

In this section, we simulate the quantum ground state of random spin-glass models with all-to-all interactions. These models generate highly correlated ground states, where every subparty shares strong correlations with all others. The Hamiltonian is given by:

$$\hat{H} = \sum_{i < j} J_{ij} \hat{\sigma}_i^x \hat{\sigma}_j^x + \sum_i h_i \hat{\sigma}_i^z$$

With, J_{ij} interaction coefficients, h_i on-site magnetic fields, and $\hat{\sigma}_i^x, \hat{\sigma}_i^z$ the Pauli-X and Pauli-Z operators on spin i . Uniformly distributed random values between -1 and 1 are assigned for both J_{ij} and h_i .

To perform simulation statistics, we generate 100 different random spin-glass instances, each with $N = 7$ spins. For each case, we perform a VQE optimization on a vanilla setting for the HEA, which we refer to as CRy-HEA. In this setup, each two-qubit entangling gate consists of an Ry gate applied to each qubit, followed by a CRy gate, with each gate having its own parameter. Details on the implementation of CRy-HEA for different connectivities are found on Appendix A. The resulting energy statistics are compared to the exact ground-state energy found with exact diagonalization, across different depths of the CRy-HEA, both with and without the swap protocol, and for various qubit connectivity configurations, see Fig. 3. For both the swapped and non-swapped versions of the HEA, CRy and swap gates are decomposed into native gates, consisting of single-qubit gates and CNOTs. Additionally, the VQE

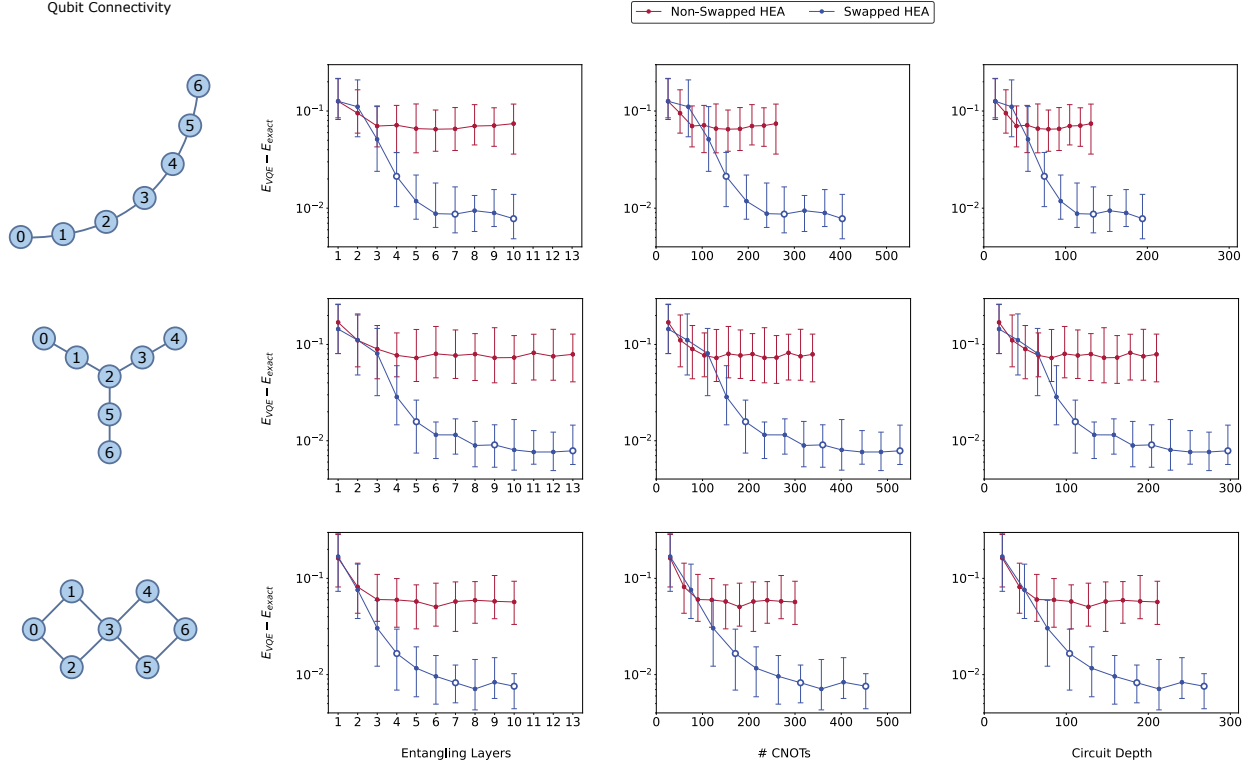


Figure 3: VQE energy error statistics for 100 random 7-spin glass instances, evaluated across varying qubit connectivities and relevant parameters. We compare the standard HEA to the version with optimized swap layers added between entangling layers across three connectivity configurations: linear (top row), heavy-hex (middle row), and square-lattice (bottom row). The energy error is shown as a function of entangling layers, CNOT count and circuit depth. CNOT count and circuit depth include the decomposition of swap gates as 3 CNOTs, and CRY with 2 CNOTs. Median values across the 100 instances are represented by dots, with error bars indicating interquartile ranges. Highlighted blue circles denote the complete application of the optimized swap network once, twice, and three times sequentially.

energy precision is shown as a function of the total number of CNOTs (major source of noise) and the overall circuit depth (qubit decoherence).

Ground-state optimization of random spin glasses is challenging due to the complex energy landscape, which features many local minima that can trap optimizers. Here, we used the gradient-free COBYLA optimizer, which mitigates the effect of getting trapped in shallow local minima. The optimization was set up to 10000 iterations.

Although highly expressive at deep circuits, we observe that the Ry-HEA optimization without swaps tends to saturate at a certain energy. This occurs because the Ry-HEA cannot capture the necessary correlations between long-range qubits, leading to an insufficient representation of the ground state. However, when the Ry-HEA is embedded into an optimized swap network

tailored to the given connectivity, it can more effectively capture correlations between all qubits in the system. This leads to a better representation of the ground state and, consequently, a reduction in the error of the ground state energy.

When comparing energy errors as a function of the parameters, the HEA embedded within the swap network yields better results for a fixed number of CNOTs, fixed circuit depth, or number of optimization parameters. Therefore, although the additional swap layers introduce an overhead in terms of CNOTs, circuit depth and parameters, their use proves beneficial to the optimization accuracy. Note that this translates into higher noise efficiency: for a fixed circuit depth or CNOT count (therefore comparable accumulated noise), it achieves lower errors and more reliable convergence.

4.2 Electronic Structure Simulations

We now turn to problems of real-world relevance, consisting in the electronic structure problems. These are central to quantum chemistry and represent the foundation of transversal theoretical simulations in fields such as drug discovery [28, 29], computational catalysis [30], or materials science [31]. VQAs provide a promising way to address those physically structured problems, both in near-term and long-term quantum hardware [32]. The proposed method is readily extendable to these physically structured systems, as demonstrated in this section.

The eigenstates of the electronic-structure Hamiltonian are the key objects underpinning most analysis in quantum chemistry. In brief, a Hamiltonian expressed in the Fermionic basis can be translated into an equivalent qubit operator by applying a suitable Fermion-to-qubit encoding. Among many of these available encodings [33, 34, 35, 36, 37, 38, 39, 40, 41], the Jordan-Wigner transformation [42] represents the paradigmatic one, encoding the occupation of spin-orbitals locally in the qubits: the i th qubit is in state $|0\rangle$ if the i th spin-orbital is unoccupied, and $|1\rangle$ if it is occupied.

A typical reference state for variational optimization consists in the Hartree-Fock state, which is the single determinant with minimum energy. The Unitary Coupled Cluster (UCC) framework parametrizes the Hartree Fock reference state via single and double excitations, restricting the parametrization to be particle-preserving [43], see Appendix B. An effective ansatz variant consists in the Unitary paired Coupled Cluster of generalized Singles and Dou-

bles (UpCCGSD), which restricts excitations to occur between spin-orbital pairs belonging to the same molecular orbital [44, 45]. This significantly reduces the resource complexity by limiting operations to molecular orbital interactions instead of arbitrary spin-orbital excitations. Here, we use approximations of these single and double paired excitations as the entangling operations (integrated in TEQUILA [26]), and embed them within a Fermionic swap network that routes the molecular orbital basis through the processor’s connectivity. Details on the implementation of the approximate single and double excitations in arbitrary qubit connectivity, can be found in Appendix B.

The qubits representing the two spin-orbitals from the same molecular orbital are positioned adjacently in the initial qubit connectivity graph $\mathcal{G}$ (see [Fig.4]). Such qubit pairs can be coarsened to yield a connectivity graph $\mathcal{G}'$ in the molecular orbital picture, indicating which pairs of molecular orbitals can be directly correlated in the processor. It is important to note that the graphs $\mathcal{G}$ and $\mathcal{G}'$ discussed here are unrelated to valence bond graphs used in quantum chemistry and other ansatz design approaches [46]. While valence bond graphs represent electron pairings and bonding topologies in a molecule, the connectivity graphs in this context describe the physical or logical arrangement of qubits and how molecular orbitals are mapped onto them for quantum simulation purposes. Within this picture, the approximate implementation of single and double excitation operations serve as building blocks for entangling layers (analogous to the HEA case), which correlate adjacent molecular orbitals in $\mathcal{G}'$. Such entangling layers are embedded within an optimized Fermionic swap network on the molecular orbital connectivity $\mathcal{G}'$ using Alg.1. Implementation details can be found in Appendix A and in the source code [19].

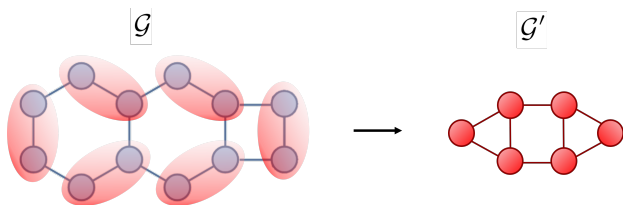


Figure 4: Example of connectivity graph coarsening from the spin-orbital (qubit) level $\mathcal{G}$ to the molecular orbital level $\mathcal{G}'$. Each blue node corresponds to a qubit holding the occupancy information of a single spin orbital. Each red node corresponds to a pair of qubits holding the occupancy information of a molecular orbital.

We simulate the ground state of the π system of p-benzyne (see [Fig.19]) using the cc-pVDZ basis set (12 spin-orbitals/qubits), detailed in Appendix A. The system exhibits an open-shell singlet biradical ground state, corresponding to a challenging strongly correlated state. We use the L-BFGS-B optimizer [47] up to a maximum

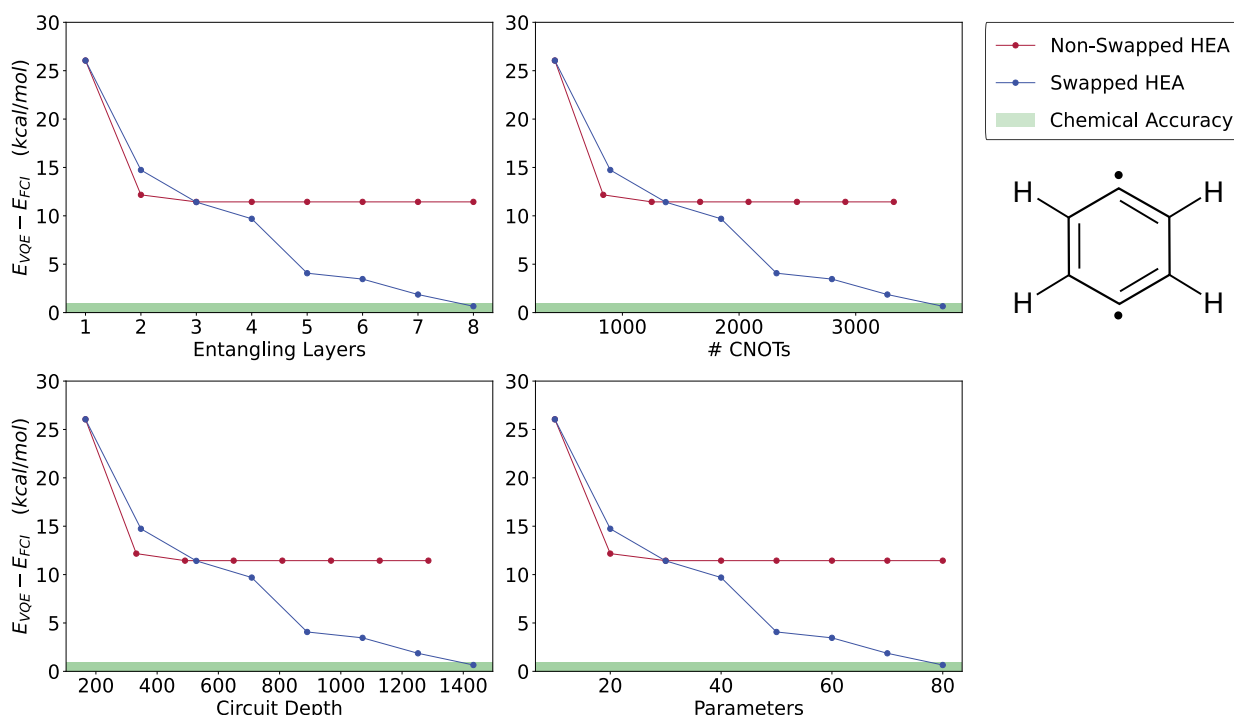


Figure 5: VQE energy error for the π orbital system of p-benzyne birradical (Appendix.A) on the qubit connectivity of [Fig. 4]. The ansatz optimization with all initialization angles close to zero (in red), is compared to the case where the swap network is embedded (in blue). The energy error is shown as a function of entangling layers, CNOT count, circuit depth, and optimization parameters.

of 1000 iterations. In the example connectivity from Fig.4, we compare the performance of the non-swapped ansatz version, with the swapped one including the Fermionic swap layers in between. As for a given amount of entangling layers the swapped version has additional resources due to the additional Fermionic swaps. We decompose the Fermionic swap operations and count the resulting elementary CNOT gates, circuit depth, and variational parameters for each simulated circuit, enabling a fair comparison between the ansätze. The converged VQE energies are also displayed as a function of the relevant parameters. All the optimizations have been performed with optimization angles close to zero [48, 49], motivated by the swap network initialization. Alternatively, the swap network can be incorporated in certain cases into the initial parameters of the optimization, as single and double excitations with angles of π effectively act as Fermionic swaps [15]. We leave the exploration of this integration for arbitrary connectivity to future work.

Numerical results are displayed in Fig.5. We observe that the non-swapped version of the

ansatz struggles to achieve chemical precision. Moreover, increasing the circuit depth up to the eight entangling layers tested, does not lead to improved accuracy. In contrast, as seen in the previous section, the swapped version of the ansatz guarantees efficient trainability, converging rapidly to the solution as simulation resources are increased. When analyzing the resulting circuits in terms of CNOT count, circuit depth, and number of optimization parameters, we find that the swapped ansatz consistently outperforms its counterpart in all of the variables. As for spin glasses, this again shows higher resilience to noise. With the same depth or number of entangling gates, and therefore equivalent noise levels, it provides improved convergence. Ultimately, this highlights the advantages of incorporating swap networks into the ansatz for practical noisy simulations.

4.3 Scaling Analysis

We now evaluate the scaling of the algorithm in the context of practical electronic-structure simulations on realistic quantum processors. Modern quantum processors typically feature two-dimensional square-grid connectivity, which

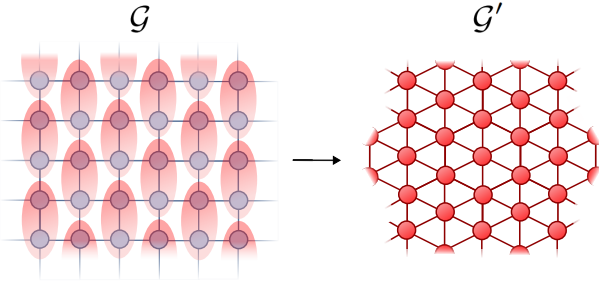


Figure 6: Selected square-grid connectivity graph coarsening from the spin-orbital (qubit) level $\mathcal{G}$ to the molecular-orbital level $\mathcal{G}'$. The resulting effective graph $\mathcal{G}'$ exhibits hexagonal symmetry with additional nodes at the centers of the hexagonal plaquettes.

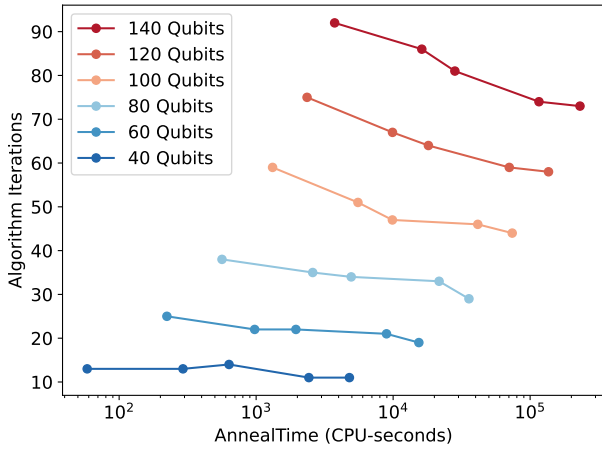


Figure 7: Scaling of algorithm iterations with qubit count and annealing time. Graph instances are from the family in Fig. 6, generated with the function `create_hex_triangular_grid`.

we will now adapt to electronic-structure problems. Following the procedure introduced in the previous section, we apply the graph coarsening illustrated in Fig. 6, which maps the native qubit-level connectivity $\mathcal{G}$ onto an effective orbital-level interaction graph $\mathcal{G}'$. The resulting graph $\mathcal{G}'$ exhibits hexagonal symmetry with additional nodes at the centers of the hexagonal plaquettes. This connectivity is particularly advantageous, as it features enhanced coordination numbers: bulk nodes are connected to six neighbors, while boundary nodes exhibit between two and five connections. In the source code [19], we provide the function `create_hex_triangular_grid` that constructs graphs from this family for a prescribed number of qubits (see examples in Appendix A).

We evaluate the scaling of the classical op-

timization algorithm on these instances as a function of the graph size in Fig. 7. The number of algorithm iterations increases with larger instances. This is expected, as more complex swap protocols and deeper swap networks are required.

As already discussed, the scaling remains practically efficient because annealing time sets a clear trade off between runtime and solution quality. Short anneals produce schedules quickly and scale to large instances, but they typically give lower quality solutions and deeper swap networks. Longer anneals cost more, yet they more reliably find higher quality schedules, often yielding shallower and more efficient swap networks. Therefore, even for large system sizes, one can choose fast approximate solutions or slower higher quality solutions depending on the available runtime budget.

4.4 Comparison to Dynamical Routing

Finally, we compare the swap network augmented ansatz with a dynamically routed all-to-all ansatz generated by a round-robin schedule and compiled with SABRE [7, 20]. For both approaches, and for different graph sizes within the family shown in Fig. 6, we decompose the circuits into native gates and report the resulting total CNOT depth in Fig. 8. We also report the total number of fSWAP gates inserted by each method, namely SABRE routing and the swap network augmented construction.

The swap network augmented ansatz requires fewer fSWAP gates in the final circuit, and since each fSWAP decomposes into multiple CNOT gates, this directly reduces the entangling gate overhead and yields a significantly shallower circuit in CNOT depth. This improvement stems from how the ansatz is constructed. It is designed to exploit the native device connectivity from the start, with a swap network that optimizes the swap protocol while the entangling pattern is generated directly from the available couplings, so much of the routing is handled by the ansatz design itself. In contrast, an all to all ansatz is defined at the logical level and must later be embedded into the hardware graph, typically requiring longer and more complex swap protocols. SABRE is highly effective at compiling such abstract circuits to a given

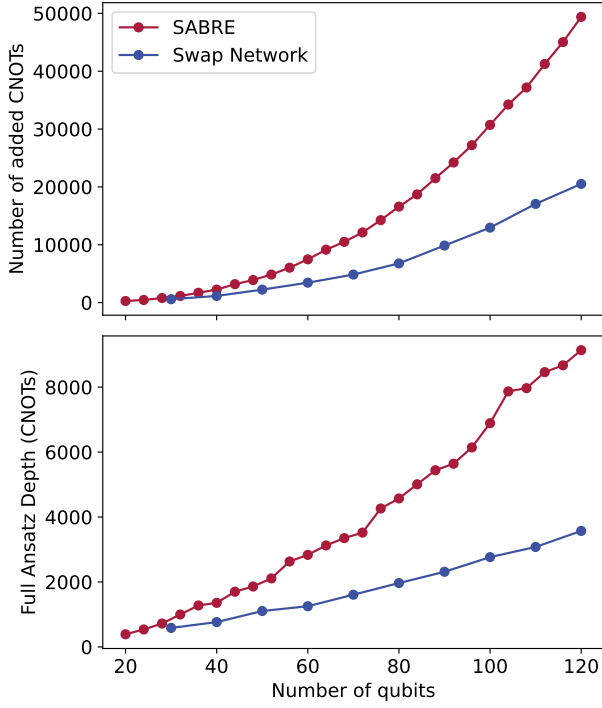


Figure 8: Comparison of circuit depth and added FSWAP count to implement a full connectivity ansatz for SABRE and the proposed swap network augmented ansatz. The comparison is conducted across instances of different graph sizes generated by the function `create_hex_triangular_grid`.

connectivity, but the swap network approach reduces the need for heavy routing by avoiding an all-to-all target and instead constructing the required connectivity directly on the hardware.

5 Outlook

Efficient design of parametrized circuit ansätze is key to the performance of variational quantum optimization. Suitable initialization techniques and parametrizations that allow an effective navigation to the solution state are a major concern. While ansätze adapted to the connectivity (eg. HEA, naive UpCCGSD) can significantly reduce resources, here we have observed that their trainability is severely compromised when using an unstructured and agnostic flow of information within the ansatz, getting stuck in a local minima.

To overcome these problems, we have proposed to include swap networks in the optimisation, which allow overcoming such trainability barriers in non-local correlated systems, as arbitrary

pairs of qubits can be directly correlated in the ansatz. Although additional resources are required for building the swap network, for the same amount of total resources the overall performance of the optimization algorithm is improved, showcasing the advantage of a synergistic hardware-friendly ansatz with a swap network. Key in our construction is the algorithm that allows the classical optimization for the swap network given an arbitrary qubit connectivity, making the swap-augmented ansatz readily available for variational optimization in the devices. Importantly, our algorithm is architecture-agnostic: it adapts to any qubit layout, and can skip faulty qubits by excluding them from the connectivity graph. Ultimately, we set optimized swap networks as a tool to be exploited in ansatz design, bridging the gap between hardware constraints and algorithmic performance.

Future work may explore swap network augmented ansätze in other contexts, such as in quantum machine learning, where circuit trainability and efficient parameterization are equally critical. Beyond simulation, it would be valuable to validate these results on real quantum hardware. More broadly, our approach has implications for beneficial circuit compilation under constrained connectivity, suggesting routing/compilation strategies that trade additional structure for improved effective expressivity. Future work may explore these directions in practical compilation and experimental demonstrations. Finally, our results open new avenues in the understanding of the tradeoff between trainability and expressivity of variational quantum circuits that deserve further investigation. In this sense, note that, when comparing circuits of equal depth, there is a choice of gate parameters in the circuit without SWAP networks that may allow recovering the circuit with the SWAP networks. That is, the ansatz without these networks is in principle more expressive than the one including them. Yet, the trainability is much harder and, in particular, a very subtle fine tuning of the parameters is required to obtain the ansatz with the SWAP networks. This implies that the parameter exploration in the optimisation is not able to find the solution with the SWAP networks and the circuit performance

is in practice much worse, despite being more expressive in theory.

ACKNOWLEDGEMENTS

We thank Júlia Barberà-Rodríguez for discussions. This work is supported by the ERC AdG CERQUTE, the Government of Spain (FUNQIP, NextGenerationEU PRTRC17.I1 and Quantum in Spain, Severo Ochoa CEX2019-000910-S), Fundació Cellex, Fundació Mir-Puig, Generalitat de Catalunya (CERCA programme), the AXA Chair in Quantum Information Science, EU project PASQUANS2, QSNP 101114043, NeQST 101080086 and COMPUTE PCI2024-153430. Ayuda PREP2022-000452 financiada por MCIN/AEI/10.13039/501100011033 y por el FSE+.

References

- [1] M. Cerezo, Andrew Arrasmith, Ryan Babbush, Simon C. Benjamin, Suguru Endo, Keisuke Fujii, Jarrod R. McClean, Kosuke Mitarai, Xiao Yuan, Lukasz Cincio, and et al. “Variational quantum algorithms”. *Nature Reviews Physics* **3**, 625–644 (2021).
- [2] Jarrod R McClean, Jonathan Romero, Ryan Babbush, and Alán Aspuru-Guzik. “The theory of variational hybrid quantum-classical algorithms”. *New Journal of Physics* **18**, 023023 (2016).
- [3] Jules Tilly, Hongxiang Chen, Shuxiang Cao, Dario Picozzi, Kanav Setia, Ying Li, Edward Grant, Leonard Wossnig, Ivan Rungger, George H. Booth, and et al. “The variational quantum eigensolver: A review of methods and best practices”. *Physics Reports* **986**, 1–128 (2022).
- [4] Sukin Sim, Peter D. Johnson, and Alán Aspuru-Guzik. “Expressibility and entangling capability of parameterized quantum circuits for hybrid quantum-classical algorithms”. *Advanced Quantum Technologies* **2** (2019).
- [5] Jonathan Romero, Ryan Babbush, Jarrod R. McClean, Cornelius Hempel, Peter Love, and Alán Aspuru-Guzik. “Strategies for quantum computing molecular energies using the unitary coupled cluster ansatz” (2018). [arXiv:1701.02691](#).
- [6] Abhinav Kandala, Antonio Mezzacapo, Kristan Temme, Maika Takita, Markus Brink, Jerry M. Chow, and Jay M. Gambetta. “Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets”. *Nature* **549**, 242–246 (2017).
- [7] Gushu Li, Yufei Ding, and Yuan Xie. “Tackling the qubit mapping problem for nisq-era quantum devices” (2019). [arXiv:1809.02573](#).
- [8] Alexander Cowtan, Silas Dilkes, Ross Duncan, Alexandre Krajenbrink, Will Simmons, and Seyon Sivarajah. “On the qubit routing problem” (2019).
- [9] Prakash Murali, Jonathan M. Baker, Ali Javadi Abhari, Frederic T. Chong, and Margaret Martonosi. “Noise-adaptive compiler mappings for noisy intermediate-scale quantum computers” (2019). [arXiv:1901.11054](#).
- [10] Chenghong Zhu, Xian Wu, Zhaohui Yang, Jingbo Wang, Anbang Wu, Shenggen Zheng, and Xin Wang. “Quantum compiler design for qubit mapping and routing: A cross-architectural survey of superconducting, trapped-ion, and neutral atom systems” (2025). [arXiv:2505.16891](#).
- [11] Adam Holmes, Sonika Johri, Gian Giacomo Guerreschi, James S Clarke, and A Y Matsuura. “Impact of qubit connectivity on quantum algorithm performance”. *Quantum Science and Technology* **5**, 025009 (2020).
- [12] Jarrod R. McClean, Sergio Boixo, Vadim N. Smelyanskiy, Ryan Babbush, and Hartmut Neven. “Barren plateaus in quantum neural network training landscapes”. *Nature Communications* **9** (2018).
- [13] Ian D. Kivlichan, Jarrod McClean, Nathan Wiebe, Craig Gidney, Alán Aspuru-Guzik, Garnet Kin-Lic Chan, and Ryan Babbush. “Quantum simulation of electronic structure with linear depth and connectivity”. *Phys. Rev. Lett.* **120**, 110501 (2018).
- [14] Bryan O’Gorman, William J. Huggins, Eleanor G. Rieffel, and K. Birgitta Whaley. “Generalized swap networks for

- near-term quantum computing” (2019). [arXiv:1905.05118](#).
- [15] Gian-Luca R. Anselmetti, David Wierichs, Christian Gogolin, and Robert M Parrish. “Local, expressive, quantum-number-preserving vqe ansätze for fermionic systems”. *New Journal of Physics* **23**, 113010 (2021).
- [16] Tobias Hagge. “Optimal fermionic swap networks for hubbard models” (2022). [arXiv:2001.08324](#).
- [17] Akel Hashim, Rich Rines, Victory Omole, Ravi K. Naik, John Mark Kreikebaum, David I. Santiago, Frederic T. Chong, Ifan Siddiqi, and Pranav Gokhale. “Optimized swap networks with equivalent circuit averaging for qaoa”. *Phys. Rev. Res.* **4**, 033028 (2022).
- [18] Júlia Barberà-Rodríguez, Nicolas Gama, Anand Kumar Narayanan, and David Joseph. “Finding dense sublattices as low energy states of a hamiltonian”. *Phys. Rev. Res.* **6**, 043279 (2024).
- [19] Teodor Parella-Dilmé. “Autoswap”. <https://github.com/teoparella/AutoSwap> (2026).
- [20] Henry Zou, Matthew Treinish, Kevin Hartman, Alexander Ivrii, and Jake Lishman. “Lightsabre: A lightweight and enhanced sabre algorithm” (2024). [arXiv:2409.08368](#).
- [21] Harper R. Grimsley, Sophia E. Economou, Edwin Barnes, and Nicholas J. Mayhall. “An adaptive variational algorithm for exact molecular simulations on a quantum computer” (2019).
- [22] Ho Lun Tang, V.O. Shkolnikov, George S. Barron, Harper R. Grimsley, Nicholas J. Mayhall, Edwin Barnes, and Sophia E. Economou. “Qubit-adapt-vqe: An adaptive algorithm for constructing hardware-efficient ansätze on a quantum processor”. *PRX Quantum* **2**, 020310 (2021).
- [23] Yordan S. Yordanov, V. Armaos, Crispin H. Barnes, and David R. Arvidsson-Shukur. “Qubit-excitation-based adaptive variational quantum eigensolver”. *Communications Physics* **4** (2021).
- [24] Martín Larocca, Supanut Thanasilp, Samson Wang, Kunal Sharma, Jacob Biamonte, Patrick J. Coles, Lukasz Cincio, Jarrod R. McClean, Zoë Holmes, and M. Cerezo. “Barren plateaus in variational quantum computing”. *Nature Reviews Physics* **7**, 174–189 (2025).
- [25] Lorenzo Leone, Salvatore F.E. Oliviero, Lukasz Cincio, and M. Cerezo. “On the practical usefulness of the Hardware Efficient Ansatz”. *Quantum* **8**, 1395 (2024).
- [26] Jakob S Kottmann, Sumner Alperin-Lea, Teresa Tamayo-Mendoza, Alba Cervera-Lierta, Cyrille Lavigne, Tzu-Ching Yen, Vladyslav Verteletskyi, Philipp Schleich, Abhinav Anand, Matthias Degroote, et al. “Tequila: A platform for rapid development of quantum algorithms”. *Quantum Science and Technology* **6**, 024009 (2021).
- [27] Yasunari Suzuki, Yoshiaki Kawase, Yuya Masumura, Yuria Hiraga, Masahiro Nakadai, Jiabao Chen, Ken M Nakanishi, Kosuke Mitarai, Ryosuke Imai, Shiro Tamiya, et al. “Qulacs: a fast and versatile quantum circuit simulator for research purpose”. *Quantum* **5**, 559 (2021).
- [28] Nick S. Blunt, Joan Camps, Ophelia Crawford, Róbert Izsák, Sebastian Leontica, Arjun Mirani, Alexandra E. Moylett, Sam A. Scivier, Christoph Sünderhauf, Patrick Schopf, Jacob M. Taylor, and Nicole Holzmann. “Perspective on the current state-of-the-art of quantum computing for drug discovery applications”. *Journal of Chemical Theory and Computation* **18**, 7001–7023 (2022).
- [29] Y. Cao, J. Romero, and A. Aspuru-Guzik. “Potential of quantum computing for drug discovery”. *IBM Journal of Research and Development* **62**, 6:1–6:20 (2018).
- [30] Vera von Burg, Guang Hao Low, Thomas Häner, Damian S. Steiger, Markus Reiher, Martin Roetteler, and Matthias Troyer. “Quantum computing enhanced computational catalysis”. *Physical Review Research* **3** (2021). url: <https://10.1103/physrevresearch.3.033055>.
- [31] Vincenzo Lordi and John M. Nichol. “Advances and opportunities in materials science

- for scalable quantum computing”. *MRS Bulletin* **46**, 589–595 (2021).
- [32] Zoltán Zimborás, Bálint Koczor, Zoë Holmes, Elsi-Mari Borrelli, András Gilyén, Hsin-Yuan Huang, Zhenyu Cai, Antonio Acín, Leandro Aolita, Leonardo Banchi, Fernando G. S. L. Brandão, Daniel Cavalcanti, Toby Cubitt, Sergey N. Filippov, Guillermo García-Pérez, John Goold, Orsolya Kálmán, Elica Kyoseva, Matteo A. C. Rossi, Boris Sokolov, Ivano Tavernelli, and Sabrina Maniscalco. “Myths around quantum computation before full fault tolerance: What no-go theorems rule out and what they don’t” (2025). [arXiv:2501.05694](#).
- [33] Sergey B. Bravyi and Alexei Yu. Kitaev. “Fermionic quantum computation”. *Annals of Physics* **298**, 210–226 (2002).
- [34] Teodor Parella-Dilmé, Korbinian Kottmann, Leonardo Zambrano, Luke Mortimer, Jakob S. Kottmann, and Antonio Acín. “Reducing entanglement with physically inspired fermion-to-qubit mappings”. *PRX Quantum* **5**, 030333 (2024).
- [35] Manuel G. Algaba, Miha Papič, Inés de Vega, Alessio Calzona, and Fedor Šimkovic IV. “Fermion-to-qubit encodings with arbitrary code distance” (2025). [arXiv:2505.02916](#).
- [36] Alexander Yurievich Vlasov. “Clifford algebras, spin groups and qubit trees”. *Quanta* **11**, 97–114 (2022).
- [37] Aaron Miller, Zoltán Zimborás, Stefan Knecht, Sabrina Maniscalco, and Guillermo García-Pérez. “Bonsai algorithm: Grow your own fermion-to-qubit mappings”. *PRX Quantum* **4**, 030314 (2023).
- [38] Aaron Miller, Adam Glos, and Zoltán Zimborás. “Treespilation: Architecture and state-optimised fermion-to-qubit mappings” (2024). [arXiv:2403.03992](#).
- [39] Mitchell Chiew, Cameron Ibrahim, Ilya Safro, and Sergii Strelchuk. “Optimal fermion-qubit mappings via quadratic assignment” (2025). [arXiv:2504.21636](#).
- [40] Mitchell Chiew, Brent Harrison, and Sergii Strelchuk. “Ternary tree transformations are equivalent to linear encodings of the fock basis” (2024). [arXiv:2412.07578](#).
- [41] Brent Harrison, Mitchell Chiew, Jason Necaie, Andrew Projansky, Sergii Strelchuk, and James D. Whitfield. “A sierpinski triangle fermion-to-qubit transform” (2024). [arXiv:2409.04348](#).
- [42] Pascual Jordan and Eugene Wigner. “Über das paulische äquivalenzverbot”. *Zeitschrift für Physik* **47**, 631 (1928).
- [43] F. Coester and H. Kümmel. “Short-range correlations in nuclear wave functions”. *Nuclear Physics* **17**, 477–485 (1960).
- [44] Joonho Lee, William J. Huggins, Martin Head-Gordon, and K. Birgitta Whaley. “Generalized unitary coupled cluster wave functions for quantum computation”. *Journal of Chemical Theory and Computation* **15**, 311–324 (2018).
- [45] Abhinav Anand, Philipp Schleich, Sumner Alperin-Lea, Phillip W. Jensen, Sukin Sim, Manuel Díaz-Tinoco, Jakob S. Kottmann, Matthias Degroote, Artur F. Izmaylov, and Alán Aspuru-Guzik. “A quantum computing view on unitary coupled cluster theory”. *Chemical Society Reviews* **51**, 1659–1684 (2022).
- [46] Jakob S Kottmann. “Molecular quantum circuit design: A graph-based approach”. *Quantum* **7**, 1073 (2023).
- [47] Richard H. Byrd, Peihuang Lu, Jorge Nocedal, and Ciyu Zhu. “A limited memory algorithm for bound constrained optimization”. *SIAM Journal on Scientific Computing* **16**, 1190–1208 (1995). [arXiv:https://doi.org/10.1137/0916069](#).
- [48] Mohamed Hibat-Allah, Marta Mauri, Juan Carrasquilla, and Alejandro Perdomo-Ortiz. “A framework for demonstrating practical quantum advantage: Racing quantum against classical generative models” (2023). [arXiv:2303.15626](#).
- [49] Edward Grant, Leonard Wossnig, Mateusz Ostaszewski, and Marcello Benedetti. “An initialization strategy for addressing barren plateaus in parametrized quantum circuits”. *Quantum* **3**, 214 (2019).
- [50] Andrew Tranter, Peter J. Love, Florian Mintert, Nathan Wiebe, and Peter V. Coveney. “Ordering of trotterization: Impact

on errors in quantum simulation of electronic structure”. *Entropy* **21**, 1218 (2019).

[51] Yordan S. Yordanov, David R. M.

Arvidsson-Shukur, and Crispin H. W. Barnes. “Efficient quantum circuits for quantum computational chemistry”. *Phys. Rev. A* **102**, 062612 (2020).

A Implementation Details

A.1 Entangling Gates Implementation

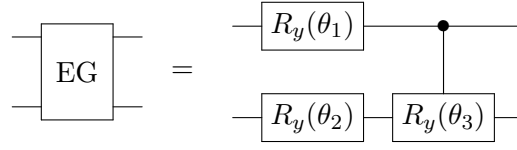


Figure 9: Entangling gate (EG) from the CRy-HEA ansatz used in the simulation of spin systems. Decomposition into single-qubit rotations $R_y(\theta_1)$ and $R_y(\theta_2)$, followed by a controlled- $R_y(\theta_3)$ gate.

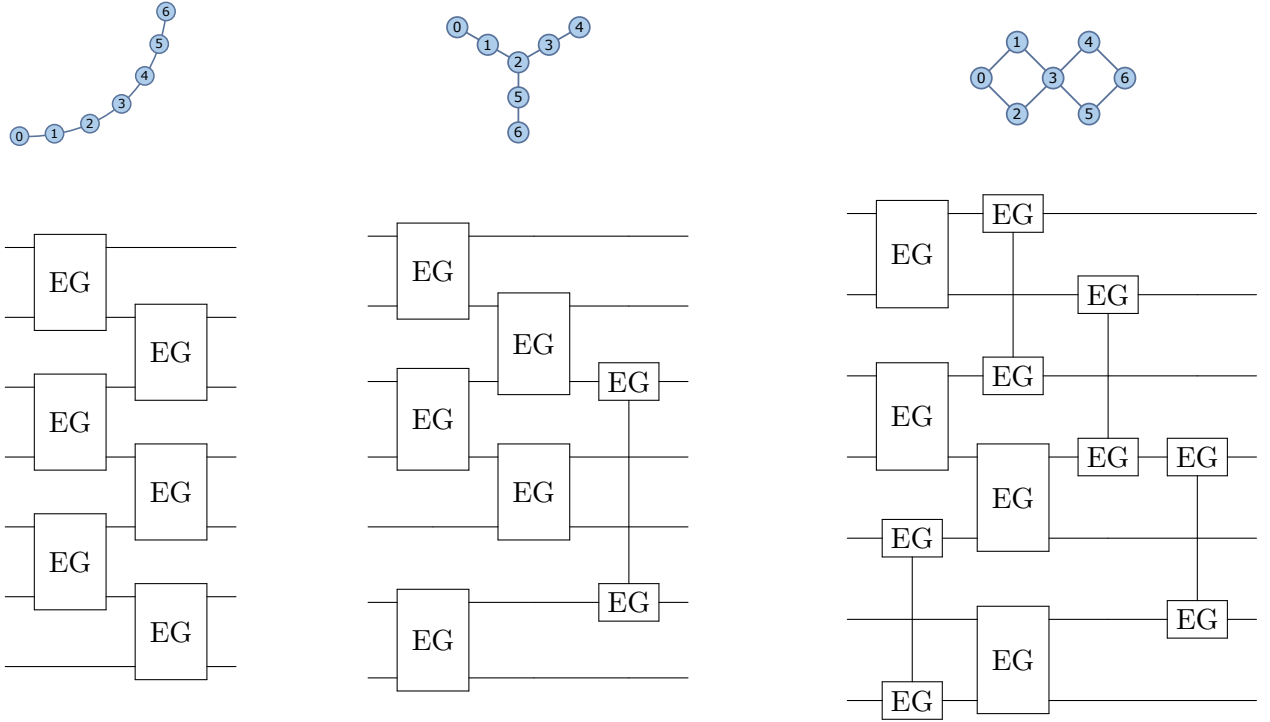


Figure 10: Entangling layers in the CRy-HEA ansatz on the exemplified 7-qubit connectivities for the simulation of spin systems. Each EG gate has its own different parameters.

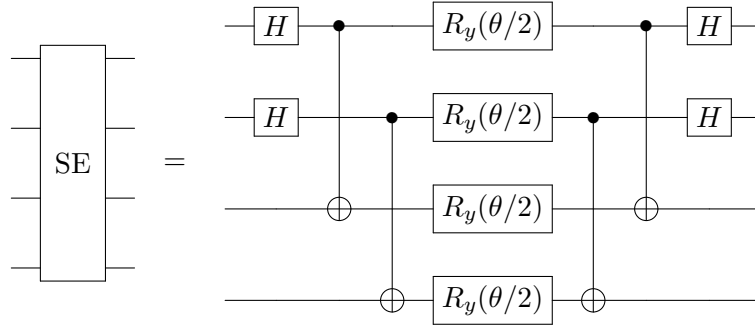


Figure 11: Decomposition of single excitation gate (SE) used in excitation-based circuits for electronic structure simulation, acting on four qubits and parametrized by a single parameter θ .

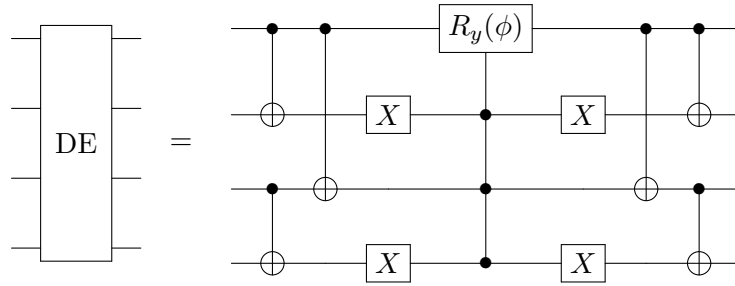


Figure 12: Decomposition of a double excitation gate (DE) used in excitation-based circuits for electronic structure simulation, acting on four qubits and parametrized by a single parameter ϕ .

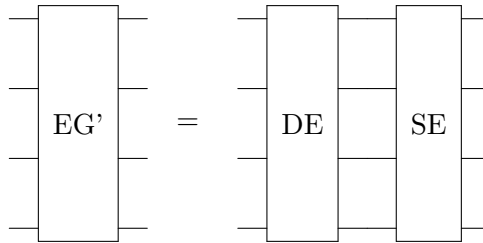


Figure 13: Decomposition of an entangling gate (EG') used in excitation-based circuits for electronic structure simulation. It acts on four qubits, and consists in a double excitation followed by a single excitation, parametrized by two parameters ϕ (DE) and θ (SE).

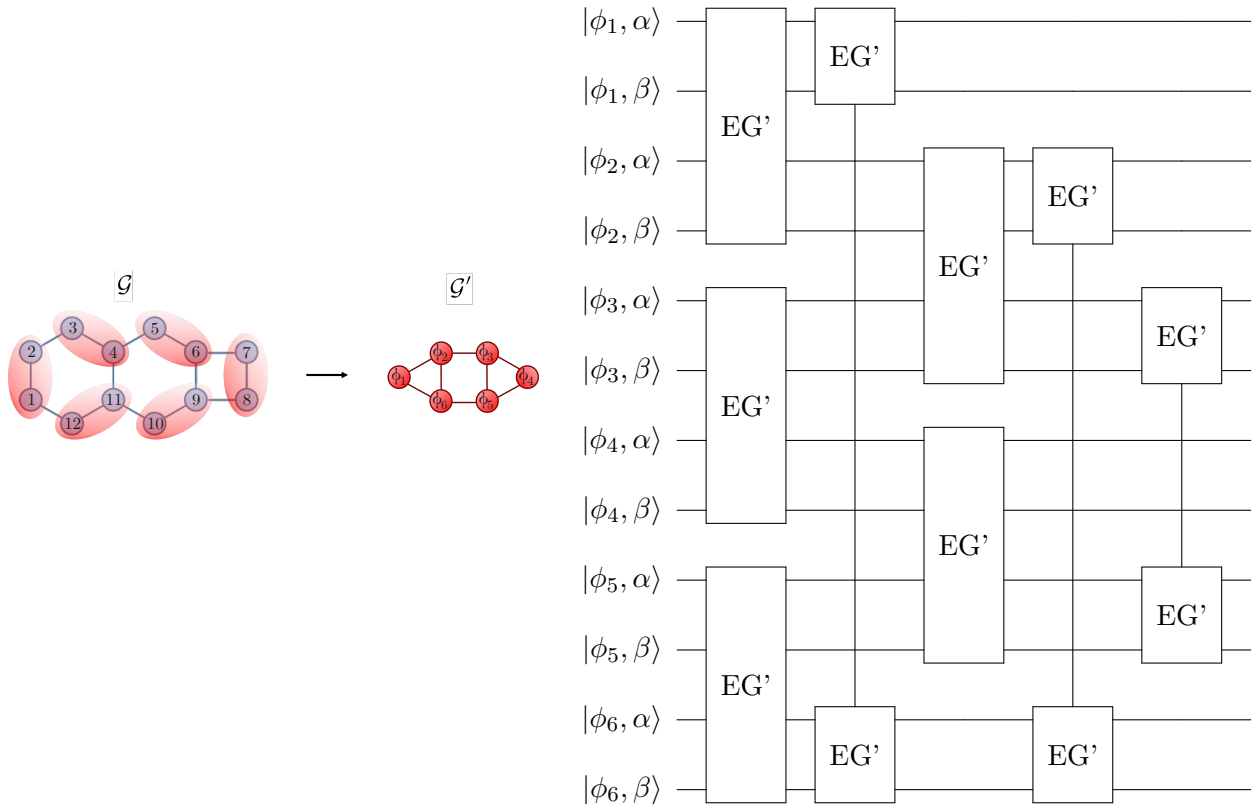


Figure 14: Entangling layer in the excitation-based ansatz used for electronic structure simulation in the exemplified connectivity. The qubit connectivity $\mathcal{G}$ (blue) is coarsened pairwise to get a connectivity in the molecular orbital level $\mathcal{G}'$ (red). Each EG gate has its own two different parameters. The depth in EG gates is equal to the maximum connectivity of a single node in $\mathcal{G}'$ (3 in the example).

A.2 Swap Networks Implementation

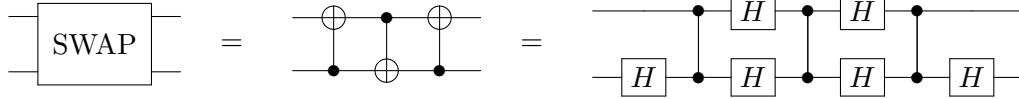


Figure 15: SWAP gate and its equivalent decompositions into CNOT gates, and CZ + Hadamard gates.

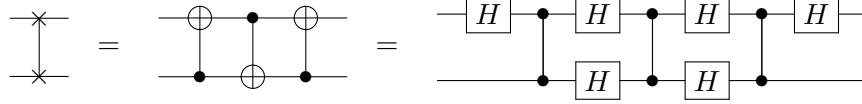


Figure 16: SWAP gate decomposition into three CNOT gates, or 3 CZ + 6 Hadamard gates.

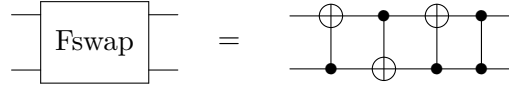


Figure 17: Fermionic Swap gate (Fswap) and its standard decomposition.

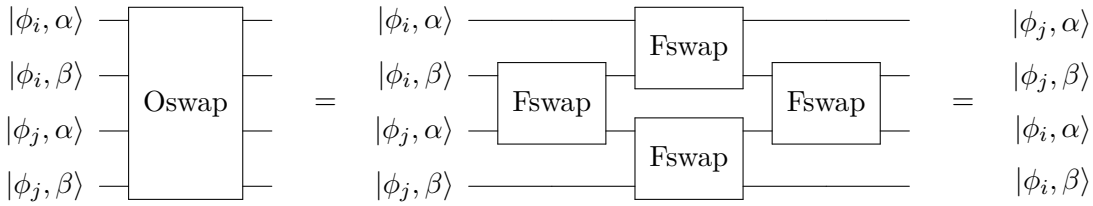


Figure 18: Decomposition of the Orbital Swap (Oswap) gate using Fswap gates. The Oswap gate exchanges the orbitals ϕ_i and ϕ_j in qubit representation.

A.3 Electronic Structure Initialization

Molecule	Basis	$E^{corr.}(kcal/mol)$	Cartesian (Å)	Geometry
p-benzyne	<i>cc-pVDZ</i>	76.7520	C -0.7396 -1.1953 0.0000 C 0.7396 -1.1953 0.0000 C 1.3620 0.0000 0.0000 C 0.7396 1.1953 0.0000 C -0.7396 1.1953 0.0000 C -1.3620 0.0000 0.0000 H 1.1999 -2.1824 0.0000 H -1.1999 2.1824 0.0000 H 1.1999 2.1824 0.0000 H -1.1999 -2.1824 0.0000	

Table 1: Initialization details for the π active space system of p-benzyne birradical.

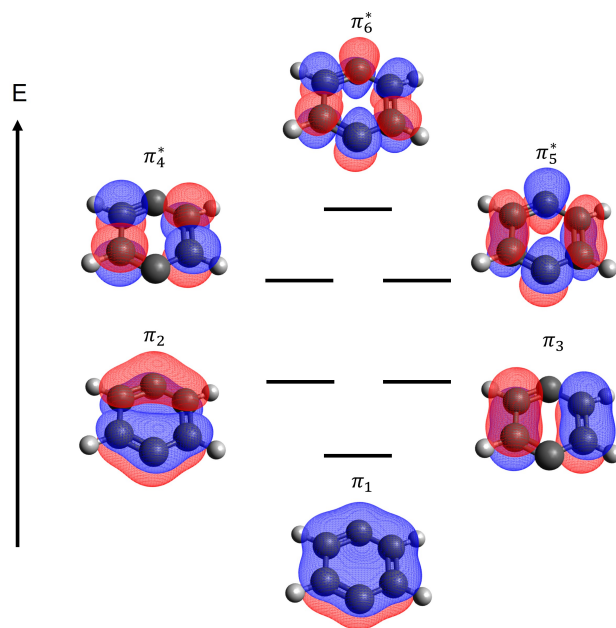


Figure 19: Active space of π molecular orbitals of p-benzyne, in the basis of canonical orbitals.

A.4 Graph examples

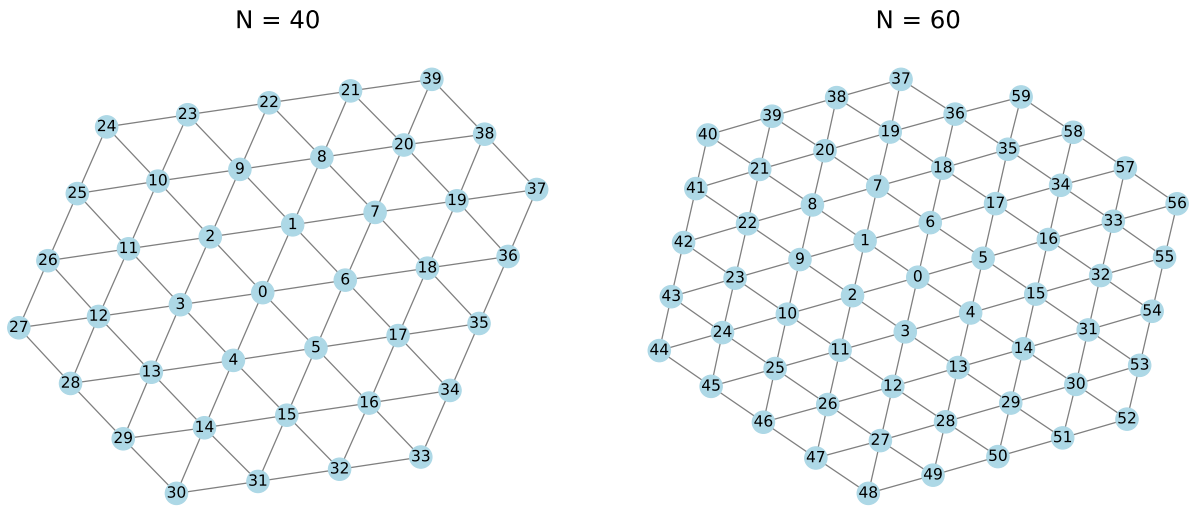


Figure 20: Finite graph examples from the graph family in Fig.6, generated with the function `create_hex_triangular_grid`.

A.5 Optimization Parameters

System	T_0	k	α	Annealing Steps	Annealings
spin glasses	10	2	2	1000	10
p-benzyne	10	2	2	1000	10
Fig.7	1	1	1	1000	10,50,100,500,1000
Fig.8	1	1	1	1000	100
Fig.21	1	1	0.1 - 3	1000	100

Table 2: Parameters used in the algorithm optimizations.

A.6 Exponent Dependence

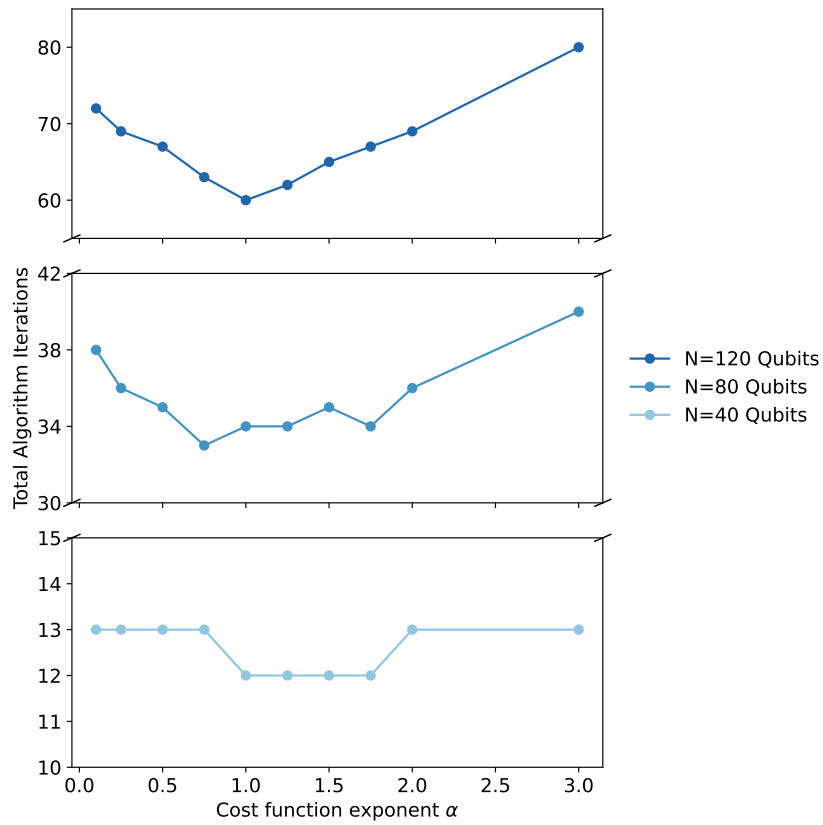


Figure 21: Analysis of the total algorithm iterations dependence on the exponent α in Eq. 1. Optimization results for three graph instances from the hexagonal family shown in Figs. 6 and 20. The total annealing time is the same for all simulations. The dependence on the exponent becomes more pronounced for larger systems, as seen for $N = 12$. Since $\alpha = 1$ consistently yields among the best performance in our simulations, we adopt this value for the large graph scaling studies, as shown in Tab.2.

B Approximate Implementation of Fermionic Excitations

In the UCC theory [43], a reference state is parametrized in the form:

$$|\psi_{UCC}\rangle = e^{T-T^\dagger} |\psi_{ref.}\rangle \quad (4)$$

with the cluster operator:

$$T = \sum_{i,j} t_i^j \hat{a}_i^\dagger \hat{a}_j + \sum_{i,j} t_{i,j}^{k,l} \hat{a}_i^\dagger \hat{a}_j^\dagger \hat{a}_k \hat{a}_l + \dots \quad (5)$$

with $t_i^j, t_{i,j}^{k,l}, \dots$ coefficients. Typically, the UCC can be restricted to single and double excitations (explicit terms in the previous equation), yielding the UCC of singles and doubles (UCCSD). Higher order excitations can be added in the description to yield higher order UCC forms, such as triples, quadruples... In the following, we consider the UCCSD form. The operation $e^{T-T^\dagger}$ is typically implemented via Suzuki–Trotter decomposition as q Trotter steps, in the form of individual single and double excitations:

$$|\psi_{UCCSD}\rangle \simeq \left[\left(\prod_{i<j} e^{\frac{t_i^j}{q} (\hat{a}_i^\dagger \hat{a}_j - \hat{a}_i \hat{a}_j^\dagger)} \right) \left(\prod_{i<j<n<m} e^{\frac{t_{i,j}^{n,m}}{q} (\hat{a}_i^\dagger \hat{a}_j^\dagger \hat{a}_n \hat{a}_m - \hat{a}_i \hat{a}_j \hat{a}_n^\dagger \hat{a}_m^\dagger)} \right) \right]^q |\psi_{ref.}\rangle \quad (6)$$

being an equality at $\lim q \rightarrow \infty$. In practice, however, one usually sets $q = 1$ to minimize implementation resources. The standard UCCSD ansatz exhibits steep circuit depth and a rapidly growing parameter space, as the amount of excitations is $\mathcal{O}(N^4)$. Moreover, the non-commutativity between the excitations makes its ordering influence the simulation [50].

Additional improvements can be performed on the UCCSD, such as restricting excitations only between Fermionic modes associated to two spatial orbitals. Repeating such a complete circuit k times, yields the k -unitary paired coupled cluster of generalized singles and doubles (k -UpCCGSD), which provides a more compact, scalable, and hardware-friendly alternative. In k -UpCCGSD, double excitations are restricted to the four spin-orbitals $\phi_a^\alpha, \phi_a^\beta, \phi_b^\alpha, \phi_b^\beta$ associated to two different molecular orbitals ϕ_a, ϕ_b , with spin α / β . That is, the only non-zero coefficients can be $t_{\phi_a^\alpha, \phi_a^\beta}^{\phi_b^\alpha, \phi_b^\beta}$. Additionally, in the closed-shell version, single excitations are symmetric between spin-orbitals of equal molecular orbitals $t_{\phi_a^\alpha}^{\phi_b^\alpha} = t_{\phi_a^\beta}^{\phi_b^\beta}$.

We now consider the spin-orbital ordering $\{\phi_1^\alpha, \phi_2^\alpha, \dots, \phi_N^\alpha, \phi_1^\beta, \phi_2^\beta, \dots, \phi_N^\beta\} \equiv \{1, 2, \dots, N, N+1, N+2, \dots, 2N\}$ under the JW transformation, where ladder operators are transformed as:

$$a_j^\dagger = \left(\prod_{k=0}^{j-1} Z_k \right) \cdot \frac{X_j - iY_j}{2}, \quad a_j = \left(\prod_{k=0}^{j-1} Z_k \right) \cdot \frac{X_j + iY_j}{2} \quad (7)$$

The implementation of single (double) excitations between adjacent molecular orbitals ϕ_i and ϕ_{i+1} , result in 2 (4) qubit operations due to the cancellation of long Z Pauli strings from the ladder operators. In this direction, UpCCGSD in a ladder qubit connectivity can be implemented using 4-qubit entanglers, where single excitations between two molecular orbitals can be implemented with 4 CNOTs, while double excitations can be implemented with 13 CNOT gates [51, 26].

In our setting with an arbitrary qubit connectivity graph, we may wish to implement excitation operators between modes associated with molecular orbitals ψ_i and ψ_j that are adjacent on the hardware graph but non-contiguous in the Fermionic ordering (i.e., $i < j$ and $j \neq i+1$). In this case, the exact implementation of single Fermionic excitations act on $1+j-i$ qubits (or $2(1+j-i)$ qubits in the case of double excitations), increasing significantly the resources.

Drawing inspiration from qubit-excitation-based excitations [23] or qubit excitations [22] from ADAPT-VQE methods [21], we implement an approximation of the Fermionic excitations in exchange to improved hardware-friendliness of the ansatz. To do so, we truncate the parity qubit sets on the pauli strings, that is, the long Z strings part. Therefore, single excitations are implemented between arbitrary orbitals ϕ_p, ϕ_q as:

$$U_{\phi_p, \phi_q}^{single}(t_p^q) \simeq \exp \left[i \frac{t_p^q}{4} \left(X_p Y_q - Y_p X_q + X_{N+p} Y_{N+q} - Y_{N+p} X_{N+q} \right) \right] \quad (8)$$

And double excitations as:

$$U_{\phi_j, \phi_k}^{double}(t_{p, N+p}^{q, N+q}) \simeq \exp \left[i \frac{t_{p, N+p}^{q, N+q}}{8} \left(Y_p X_{N+p} X_q X_{N+q} + X_p Y_{N+p} X_q X_{N+q} - X_p X_{N+p} Y_q X_{N+q} \right. \right. \\ \left. \left. - X_p X_{N+p} X_q Y_{N+q} + Y_p Y_{N+p} X_q Y_{N+q} + Y_p Y_{N+p} Y_q X_{N+q} \right. \right. \\ \left. \left. - X_p Y_{N+p} Y_q Y_{N+q} - Y_p X_{N+p} Y_q Y_{N+q} \right) \right] \quad (9)$$

being equivalent as in the case of two adjacent qubits in the Fermionic enumeration. That is, any single (double) excitation gate in the arbitrary connectivity setting, is implemented with 4 (13) CNOT gates.