Digital-Analog Counterdiabatic Quantum Optimization with Trapped Ions

Shubham Kumar [ID],[1] Narendra N. Hegade [ID],[1, *] Alejandro Gomez Cadavid [ID],[1, 2]
Murilo Henrique de Oliveira [ID],[1] Enrique Solano [ID],[1] and F. Albarrán-Arriagada [ID] [3]

[1]*Kipu Quantum, Greifswalderstrasse 226, 10405 Berlin, Germany*
[2]*Department of Physics, University of the Basque Country UPV/EHU, Barrio Sarriena, s/n, 48940 Leioa, Biscay, Spain*
[3]*Departamento de Física, CEDENNA, Universidad de Santiago de Chile (USACH), Avenida Víctor Jara 3493, 9170124, Santiago, Chile*

We introduce a hardware-specific, problem-dependent digital-analog quantum algorithm of a counterdiabatic quantum dynamics tailored for optimization problems. Specifically, we focus on trapped-ion architectures, taking advantage from global Mølmer-Sørensen gates as the analog interactions complemented by digital gates, both of which are available in the state-of-the-art technologies. We show an optimal configuration of analog blocks and digital steps leading to a substantial reduction in circuit depth compared to the purely digital approach. This implies that, using the proposed encoding, we can address larger optimization problem instances, requiring more qubits, while preserving the coherence time of current devices. Furthermore, we study the minimum gate fidelity required by the analog blocks to outperform the purely digital simulation, finding that it is below the best fidelity reported in the literature. To validate the performance of the digital-analog encoding, we tackle the maximum independent set problem, showing that it requires fewer resources compared to the digital case. This hybrid co-design approach paves the way towards quantum advantage for efficient solutions of quantum optimization problems.

I. INTRODUCTION

Optimization problems play a central role in several scientific, engineering, and economic applications. Among the vast list of optimization problems, the quadratic unconstrained binary optimization (QUBO) formulation is particularly relevant, not only because of the number of industrial applications that can be mapped to it [1], but also because most of the existing quantum hardware can encode such problems [2–5]. Quantum computing aims to address this class of optimization problems, where one can take advantage of the all-to-all connectivity presented by trapped-ion quantum processors [6–9].

In the last years, the use of control techniques to enhance quantum algorithms for optimization problems has emerged. Specifically, we highlight the digitized counterdiabatic quantum computing (DCQC) paradigm [10], where polynomial scaling enhancement has been reported in comparison to conventional methods [11]. DCQC has been used in several contexts from protein folding to factorization, showing the potential and flexibility of the technique in current digital quantum computers [12, 13].

However, with the current noisy intermediate-scale quantum (NISQ) computers, implementing scalable digital methods is not feasible within the available coherence times due to the substantial circuit depth required. Quantum algorithms using just single-qubit and two-qubit gates cause an overhead in the number of gates restricting the size of the problems that can be addressed on current quantum hardware. To counteract these limitations, there is a growing interest in leveraging multiqubit gate operations, or analog blocks to reduce the circuit depth by the simultaneous entanglement of multiple qubits through a single operation, offering a more efficient encoding. Complementing analog interactions with digital gates can reduce the circuit depth, and meanwhile address a variety of problems. It means that a suitable analog block reduces the circuit depth, increasing the success of the algorithms, maintaining a reasonable flexibility and allowing to address a variety of problems.

The digital-analog quantum computing (DAQC) paradigm [14] was proposed and implemented to enhance quantum algorithms, including the case of quantum simulations [15–26]. Along these lines, DAQC was able to show the potential, flexibility and feasibility offered by current NISQ processors. DAQC solutions require careful design because it is not only hardware dependent, since analog blocks are hardware specific, but also problem dependent, due to specific interactions in the target Hamiltonian.

In this work, we introduce a hardware-specific DAQC protocol to solve larger instances of QUBO problems in trapped ions with the aid of counterdiabatic quantum dynamics. We show the advantage of using Global Mølmer-Sørensen (GMS) gates [27–30] to obtain many-body interactions which are pivotal in different approximations of the counterdiabatic expansion. We complement it with digital steps in an optimal configuration to achieve a substantial reduction in algorithmic circuit depth. This leads to targeting larger instances of the optimization problem within the available coherence time of the quantum processor. Furthermore, it lays a strong foundation on how to engineer and deploy the digital-analog quantum algorithm on any current NISQ processor, along its specifications, to target larger problem instances.

II. COUNTERDIABATIC DRIVINGS

We focus on QUBO problems whose solution can be encoded in the ground state of the Hamiltonian

$$H_f = \sum_{i<j} J_{ij}\sigma_i^z\sigma_j^z + \sum_i h_i\sigma_i^z. \qquad (1)$$

Here, J_{ij}, is the coupling strength between the i-th and j-th qubit, and h_i is the local energy of the i-th qubit, σ_i^z are the Pauli Z operators for the i-th qubit. To reach the ground state, we can perform an adiabatic evolution governed by the time-dependent Hamiltonian

$$H_{ad}(t) = \lambda(t)\left(\sum_{i<j} J_{ij}\sigma_i^z\sigma_j^z + \sum_i h_i\sigma_i^z\right)$$
$$+ [1 - \lambda(t)]\sum_i \sigma_i^x, \qquad (2)$$

with initial state $\otimes_j |-_j\rangle$, where $|-_j\rangle = \frac{1}{\sqrt{2}}(|0_j\rangle - |1_j\rangle)$. The scheduling function $\lambda(t)$ is in principle an arbitrary function with the boundary conditions $\lambda(0) = 0$ and $\lambda(T) = 1$ with T, the total time of the adiabatic evolution.

To accelerate this optimization process, we can use shortcut-to-adiabaticity techniques, for example, to add counterdiabatic (CD) terms [10, 11, 31–34]. That is, to evolve the system under the following Hamiltonian

$$H(t) = H_{ad}(t) + \dot{\lambda}(t)A_\lambda(t), \qquad (3)$$

instead of the purely adiabatic method. Here, A_λ is called the adiabatic gauge potential (AGP). There exists several approximations to the AGP such as mean field approximation, variational approach [35], and nested commutator expansion [36]. Recently, the last one has gained relevance since it provides an approximation with different complexity in the many-body interaction terms, that is

$$A_\lambda(t) = i\sum_j^\infty \alpha_j(t)\underbrace{[H_{ad}, [H_{ad}[...[H_{ad}, \partial_\lambda H_{ad}]]]]}_{2j-1}, \qquad (4)$$

where α_j can be determined as proposed in Ref. [37]. In this approximation, the first order takes the form

$$A^{(1)}(t) = 2\alpha_1(t)\left[\sum_i h_i\sigma_i^y + \sum_{i<j} J_{ij}(\sigma_i^y\sigma_j^z + \sigma_i^z\sigma_j^y)\right]. \qquad (5)$$

The detailed calculation of the analytical expression of α_1 is given in Appendix VIII. Notice that the Hamiltonian of Eq. (5) involves only local and bilocal terms, matching with capabilities of current devices. In trapped ions, GMS interactions can realize $\sigma_i^z\sigma_j^z$, $\sigma_i^z\sigma_j^y$ and $\sigma_i^y\sigma_j^z$ simultaneously in a single gate operation. Below, we show a digital-analog encoding of the above problem which can be performed in trapped ions.

III. THE ALGORITHM

We introduce the digital-analog counterdiabatic quantum optimization (DACQO) algorithm. As the main analog block of the algorithm, we utilize the GMS gate as a resource, considering trapped-ion quantum processors as the target. This GMS gate can be generated by applying a

global entangling operation over a subset of qubits using a uniform microwave or laser field across a set of ions. This approach allows for the simultaneous manipulation of multiple qubits, facilitating the generation of large-scale entangled states [30, 38]. The GMS gate, complemented with single qubit rotations, has shown to be advantageous in reducing the gate count in quantum circuits for executing quantum algorithms [30]. In this work, we mix the circuit-depth reduction provided by the GMS and the power of digitized counterdiabatic quantum computing to solve binary optimizations, providing a significant step in the path to quantum advantage with digital-analog quantum computing protocols. A GMS gate is given by

$$U_{MS}^k(\theta, \phi) = \exp\left[-\frac{i\theta}{4}(\cos\phi S_x + \sin\phi S_y)^2\right], \qquad (6)$$

where, $S_{x,y} = \sum_{i=1}^k \sigma_i^{x,y}$ and k is the number of ions (qubits) to which the corresponding gate is applied. The GMS gate produces 2-local terms, as well as effective n-local term as was shown in Ref. [39], which can be use to implement high-order terms in the AGP approximation. To understand how to get effective n-local terms, see supplementary sec. VIII A and VIII B for a complete derivation.

Since the analog block (GMS gate) can only produce terms of the form $\sigma_i^x\sigma_j^x$, $\sigma_i^x\sigma_j^y$ and $\sigma_i^y\sigma_j^y$, we can write the Hamiltonian for our dynamics as

$$H'(\lambda) = \lambda(t)\left[\sum_{i<j=1}^N J_{ij}\sigma_i^x\sigma_j^x + \sum_{i=1}^N h_i\sigma_i^x\right]$$
$$+ (1 - \lambda(t))\left[\sum_{i=1}^N \sigma_i^z\right] + 2\dot{\lambda}\alpha_1(t)\left[\sum_{i=1}^N h_i\sigma_i^y\right.$$
$$\left. + \sum_{i<j=1}^N J_{ij}(\sigma_i^y\sigma_j^x + \sigma_i^x\sigma_j^y)\right]. \qquad (7)$$

Now, all the 2-local terms in Eq. (7) can be realized by the analog block in Eq. (6) without extra rotations. Next, we will digitized the evolution governed by the Hamiltonian (7) for two cases, first the homogenous case (best case) and then for the full inhomogenous case (worst case).

A. The homogeneous case

For the homogeneous case, we consider $J_{ij} = J, h_i = h$ in Eq. (7). We note that, from Eq. (6), we can obtain bilocal terms $\sigma_i^a\sigma_j^b$, where $a(b) = \{x, y\}$, with strength $\theta^{a,b}$ given by

$$\theta\cos^2(\phi)/2 = \theta^{x,x},$$
$$\theta\sin^2(\phi)/2 = \theta^{y,y},$$
$$\theta\cos(\phi)\sin(\phi)/4 = \theta^{x,y} = \theta^{y,x}. \qquad (8)$$

This maps the Hamiltonian coefficients to the angles of the GMS gate given in Eq. (6), obtaining $\theta^{x,x} =$
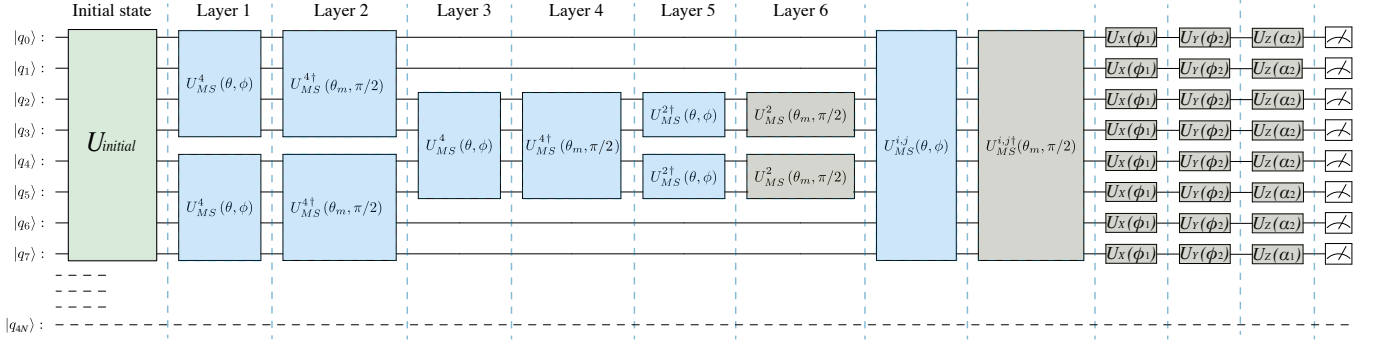
FIG. 1. Digital-analog quantum algorithm to perform CD-based optimization of homogeneous Ising Hamiltonian in trapped ions, a model widely utilized in quantum simulations of magnetic materials and combinatorial optimization problems. The circuit is modular and extendable, capable of accommodating an arbitrary number of qubits (N) to simulate larger systems. Key components of the circuit include analog blocks, highlighted in blue, which are responsible for the execution of continuous-time quantum operations that directly correspond to the $2 - local$ interactions in the Ising model and the digital blocks (highlighted in grey) are integrated within a digital framework that provides precise control and measurement capabilities. The modular nature of this design allows for scalable quantum computing applications, from fundamental physics research to practical problem-solving in industry-related topics.
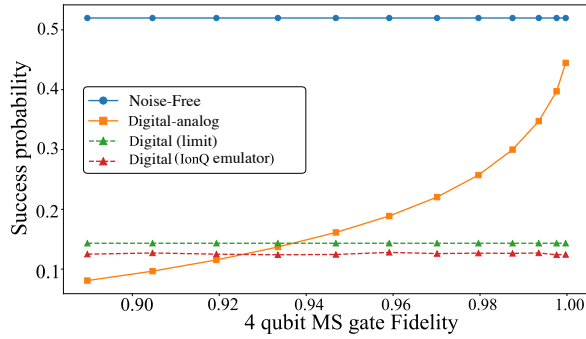

FIG. 2. DACQO applied to 4 qubit homogeneous Ising model. The figure illustrates the expected success probability as the fidelity of the 4-qubit GMS gate (analog block) improves for 10 trotter step and 1 unit of evolution time. The purpose of the figure is to show the region where digital-analog can surpass purely digital simulation.

$\lambda(t)Jt/n$ and $\theta^{x,y} = 2J\dot{\lambda}(t)\alpha_1(t)t/n$ with t the time and n the number of trotter steps (see supplementary information Sec. VIII C). Now, the local terms can be done by single-qubit rotations with angles $\theta^x = \lambda(t)ht/n$, $\theta^z = (1 - \lambda(t))t/n$, $\theta^y = 2\dot{\lambda}(t)\alpha_1(t)ht/n$. Finally, the GMS gate also produces an extra $\sigma^y\sigma^y$ terms, that can be eliminated by applying the conjugate GMS gate $U_{MS}^{k\dagger}(\theta_m, \pi/2)$, where $\theta_m = \theta\sin^2(\phi)$. Considering all these ingredients for the digital-analog proposal, the algorithms is as follow.

Algorithm description: The algorithm is depicted in Fig. 1 and described in **Algorithm 1**. We consider experimentally feasible GMS gates as analog block, then we restrict it only to the nearest-neighbour ions up to 4 qubits [29]. Despite this, our aim is to entangle as many ions as possible while still achieving a reasonable fidelity.

Layer 1 and **Layer 2** involve the application of the 4-qubit GMS gates. Layer 3 aims to entangle the remaining set of k nearest neighbor qubits as efficiently as possible

---

Algorithm 1: Digital-analog counterdiabatic quantum optimization for homogeneous problems

Layer 1: Initial entanglement;
for *each k set of nearest neighbour qubits* **do**
    Apply k-qubit GMS gates $U_{MS}^{k\dagger}(\theta, \phi)$;

Layer 2: Parasitic term elimination;
for *each k set of qubits affected in Layer 1* **do**
    Apply conjugate GMS gates $U_{MS}^{k\dagger}(\theta_m, \pi/2)$;

Layer 3: Supplementary entanglement;
for *remaining nearest neighbour k set of qubits* **do**
    Entangle using $U_{MS}^k(\theta, \phi)$;

Layer 4: Further parasitic term elimination;
For same set of qubits as layer 3, apply $U_{MS}^{k\dagger}(\theta_m, \pi/2)$;

Layer 5: Overlap correction;
for *overlapping GMS gates* **do**
    Apply 2-qubit GMS gates;

Layer 6: Final parasitic term elimination;
Target the last set of parasitic terms using $U_{MS}^{2\dagger}(\theta_m, \pi/2)$.;

Ion pair entanglement;
Entangle remaining ion pairs using 2-qubit GMS gates;
Remove the parasitic terms as in layer 2;

Apply single qubit gates;
for *all ions* **do**
    Apply single-qubit rotations;

Perform unitary time evolution;

---

while minimizing the creation of parasitic terms. **Layer 5** is designed to eliminate any parasitic terms that arise from the overlap between **Layer 3** and **Layer 4**, using conjugate GMS gates.

For the step 21, the number of ion pairs left to be entangled scales as $n_p = (N - k)(N - k + 1)/2$ and the gate requirement scales as $2n_p$ (including counjugate GMS gates). In trapped ions, ideally $N/2$ gates can be performed simultaneously, this leads to a circuit depth of
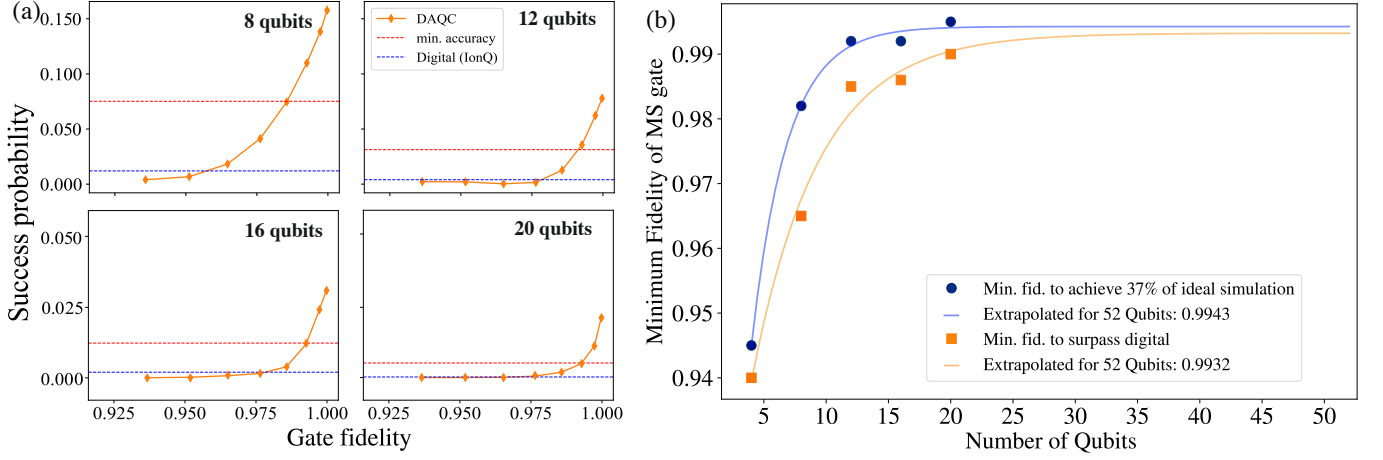
FIG. 3. Analysis of the success probability and gate Fidelity requirements to run digital-analog counterdiabatic quantum algorithms in trapped ions (2 trotter steps and 15 units of annealing time). (a) This panel illustrates the success probability achievable for varying problem sizes as the GMS gate fidelity increases. The results are compared against the peak performance achievable through purely digital quantum simulations (run on IonQ noisy emulator). Additionally, the figure includes a benchmark indicating the minimum accuracy threshold set at 37 % of the ideal success probability (current benchmark for circuit success probability by IonQ [40]), highlighting the efficacy of the algorithm in meeting or exceeding this benchmark across different fidelity levels. (b) In this panel, we extrapolate the minimum gate fidelity required for the algorithm to achieve the stated minimum accuracy and to surpass the capabilities of digital simulation, up to a hypothetical 52-qubit system. The extrapolation is based on the observed fidelity trends for problem sizes up to 20 qubits (multiple measurements with 1024 shots and averaged overall), providing insights into the scalability of the algorithm in handling larger quantum systems with stringent accuracy requirements. The extrapolation employs an exponential decay model towards a limit, mathematically represented as $L + (K - L)e^{\lambda N}$, where $L = 1$ denotes the limiting fidelity value, K represents the initial fidelity, $\lambda = 0.1$ is the decay constant, and N is the number of qubits.

$\frac{2(N-k)(N-k+1)}{N}$. The total circuit depth for a N qubit system can be expressed as the number of gate layers that can be executed sequentially, where each layer consists of gates that can be executed in parallel as follows

$$\text{Depth} = \text{layers}[U_{MS}^4(\theta)] + \text{layers}[U_{MS}^2(\theta)] + \text{layers}[U_{X,Y,Z}]$$
$$= 9 + \frac{2(N-k)(N-k+1)}{N}. \tag{9}$$

As a proof of concept, we solve the 4-qubit QUBO problem using a 4-qubit analog block to demonstrate the efficiency of the algorithm. We perform a noisy simulation by introducing noise using a random matrix noise as $U_{noisy} = c \cdot \mathcal{R}^4$, where c is the noise amplitude and $\mathcal{R}^4$ is a 4-qubit random matrix applied to the GMS gate unitary, Eq. (6). The purpose of this manual noise effect is that it allows us to manually control the noise (tuning c) to understand the performance of the algorithm based on the analog gate fidelity, which will give us an approximate idea of the minimal performance required by the analog building blocks to run the algorithm on a hardware. As a realistic noise source, we introduce depolarizing error of magnitude 0.02% (current state of the art of *IonQ* hardware) local to the qubits (performed using Qiskit's noise model [41]). The analysis presented in Fig. 2 illustrates the enhancement of success probability for achieving the ground state as the fidelity of the analog block increases (corresponding to a reduction in noise amplitude). As a benchmark, we consider the digital quantum simulation performed in *IonQ* noisy emulator. This platform provides a $\sigma_i^x \sigma_j^x$ gate with a fidelity ($\mathcal{F}$) of 99.5%, and with the corresponding success probability, we

---

Algorithm 2: Digital-analog counterdiabatic quantum optimization for inhomogeneous problems

Step 1: Initial entanglement with inhomogeneity;
for *each k set of nearest neighbour qubits* **do**
  Replace homogeneous $U_{MS}^{k\dagger}(\theta, \phi)$ with $k(k-1)/2$ blocks;
  Apply local rotations to introduce inhomogeneity;

Step 2: Parasitic term elimination;
for *each k set of qubits affected in Step 1* **do**
  Apply conjugate GMS gates $U_{MS}^{k\dagger}(\theta_m, \pi/2)$;

Step 3: Supplementary entanglement with inhomogeneity;
for *remaining nearest neighbour k set of qubits* **do**
  Entangle using $k(k-1)/2$ number of $U_{MS}^k(\theta, \phi)$ blocks;
  Apply local rotations for inhomogeneity;

Step 4: Further parasitic term elimination;
For same set of qubits as layer 3, apply $U_{MS}^{k\dagger}(\theta_m, \pi/2)$;
Step 5: Overlap correction;
for *overlapping GMS gates* **do**
  Apply 2-qubit GMS gates;

Step 6: Final parasitic term elimination;
Target the last set of parasitic terms using $U_{MS}^{2\dagger}(\theta_m, \pi/2)$;
Ion pair entanglement;
Entangle remaining ion pairs using 2-qubit GMS gates;
Remove the parasitic terms as in Step 2;
Apply single qubit gates;
for *all ions* **do**
  Apply single-qubit rotations;

Perform unitary time evolution;
Apply trotterization to simulate the time evolution;

can test the performance of the algorithm. Notably, the results demonstrate that DAQC has the potential to surpass digital simulation performance when the analog block fidelities exceed approximately 94%. This finding is particularly significant, underscoring the ability to achieve high-accuracy simulations even in the presence of noisy analog blocks, thereby highlighting the robustness and efficiency of digital-analog encoding in leveraging imperfect quantum hardware to realize practical quantum computation.

We have expanded our analysis to explore the scalability of the algorithm by evaluating the system capacities that can be effectively targeted. To this end, we utilize a 4-qubit analog block to perform DACQO across systems comprising $4N$ qubits (since we use a 4-qubit analog block). This approach enables us to assess the fidelity thresholds necessary for analog blocks to outperform digital simulations and achieve satisfactory accuracy levels. The results of this analysis are presented in Fig. 3. Additionally, we have extended our analysis to project the minimum fidelity thresholds necessary for surpassing the performance benchmarks set by digital simulations and meeting the established minimum accuracy standards. These projections, which estimate the fidelity requirements for systems scaling up to 52 qubits, leverage the observed trends in our current data to anticipate the evolving demands on quantum system performance as it expands. The findings indicate that for system sizes up to 20 qubits, GMS gate fidelities within the $98\% - 99\%$ range are adequate to surpass purely digital simulations and fidelities between $99\% - 99.5\%$ are required to achieve minimum accuracy benchmark. The extrapolated data suggests that to meet or exceed both benchmarks, fidelities around approximately 99.5% are required, which are achievable with current state-of-the-art capabilities [42]. This means that for achieving reasonable accuracy, the necessity of engineering very high fidelity 2-qubit gates [43] can be relaxed and instead one can focus on engineering analog gates with a reasonable fidelity by using similar techniques, and finally supplement it with the available digital gate sets to perform digital-analog encodings.

The inhomogeneous case:

Particularly interesting is the case when one would like to tackle an inhomogeneous Hamiltonian, Eq. (1) (different J_{ij}'s). The algorithm introduced for the homogeneous case can be extended to the inhomogeneous case. In principle, creating inhomogeneity using a homogeneous analog block will require more gates depending on the size of the analog blocks used. To solve a N qubit inhomogeneous case using a $k - qubit$ analog block, we can create inhomogeneity by applying local rotations to a sequence of analog blocks. A simple example to illustrate how to realize the $2 - local$ terms for a 3-qubit QUBO Hamiltonian, Eq. (7) using 3-qubit analog blocks is shown in the Supplementary VIII D. In general, to introduce inhomogeneity across any k-qubit interactions within a N-qubit system, it is necessary to employ $k(k - 1)/2$ analog blocks [14] along with local rotations. Consequently, a pragmatic strategy for addressing inhomogeneous QUBO problems involves the systematic realization of inhomogeneity for each subset of k qubits. This approach is critical for effectively managing the complexity

of interactions in any N-qubit system. This can be done by replacing only the analog blocks $(U_{MS}^{k\dagger}(\theta, \phi))$ in the **Algorithm 1** with $k(k - 1)/2$ homogeneous analog blocks supplemented with local rotations. the complete algorithm is explained in **Algorithm 2**. The number of local rotations required scales as $k(k - 1)$ for each k-qubit block. This suggests that bigger analog blocks will incur a large circuit depth. Along these lines, we study the scaling of the DACQO algorithm for homogeneous and inhomogeneous cases and by using different analog block sizes. This will reveal the size of the problem in terms of the number of qubits that can be solved using current trapped ion quantum computers. As a benchmark we will consider the current specifications of $IonQ$ (Forte device).

IV. SCALING

We study the scaling of the DACQO algorithm to see the effective system sizes it can solve within the coherence time allowed by current trapped ion hardwares. By accounting for both multi-qubit and single-qubit gate durations, we can calculate the overall circuit runtime in seconds. This quantity can be represented as the sum of the depths multiplied with their duration, with each depth's duration determined by the execution times of respective layers, as shown below:

$$\begin{aligned} \text{Circuit Runtime (sec)} \;=\; & t_M \cdot \text{depth}[U_{MS}(\theta, \phi)] \\ & + t_S \cdot \text{depth}[U_{X,Y,Z}], \end{aligned} \quad (10)$$

where t_M and t_S are the multiqubit and single-qubit gate times, respectively and $layers[U_{MS}(\theta)]$, $t_s(layers[U_{X(Y,Z)}])$ are the number of layers of multi-qubit and single qubit gates. We consider the gate times according to the current state of the art for 2-qubit GMS gates and single qubit gates that are available in current quantum processors (we consider the *IonQ* Forte system for our current analysis). For the 4-qubit GMS gates, we can consider their anticipated gate times achievable with the current technology [42] which is equivalent to the current 2-qubit gate times. In the current trapped ion systems, for example in the $IonQ\ Forte$ system, the gate times are $t_M = 930\mu s$ and $t_S = 130\mu s$. Based on these numbers achievable with the current technology, we study the circuit runtime of the DACQO algorithm with respect to the size of the optimization problem upto 100 qubits. The circuit runtime plots for both homogeneous and inhomogeneous fully-connected spin glass models are presented in Fig. 4 (a), utilizing 4-qubit GMS gates as the analog blocks. In this figure, the digital-analog encoding, embodying the DACQO paradigm, demonstrates a notably reduced runtime compared to the fully digital paradigm within the permissible coherence time of the current trapped ion processors indicating that employing DAQC can target larger problem instances (upto 55 qubits) requiring more qubits. To target larger instances of the problem, such as those involving 100 qubits, will require extending the processor's coherence time to approximately 1.8 seconds. This represents a significant improvement when
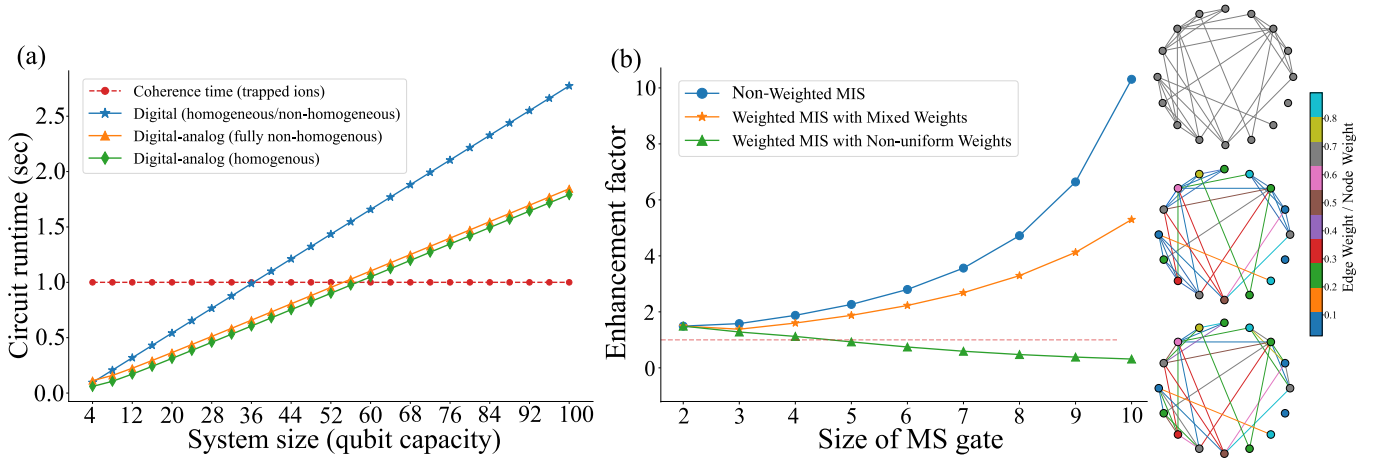
FIG. 4. Scaling of the algorithm using GMS gates as the analog blocks. (a) Circuit runtime using 4-qubit analog blocks, and (b) enhancement factor for 16 node MIS problem using analog blocks of varying qubit sizes.

compared to the purely digital approach, which requires about 2.8 seconds, thus achieving a 55% reduction in the runtime.

Furthermore, for demonstrating the efficiency of the DACQO algorithm, we consider different block sizes and study the enhancement factor ($\mathcal{R}_{Digital}/\mathcal{R}_{DAQC}$), where $\mathcal{R}$ is the circuit runtime for a fixed problem size. We select the maximum independent set (MIS) problem for this purpose with three different instances involving weighted and non-weighted graphs and 16 nodes. To tackle this problem efficiently, we can make use of the all-to-all connectivity of the trapped-ion quantum processors. We can represent the nodes with equal weights as the nearest neighbour qubits and apply the analog blocks only in $Layer\,1$, as shown the quantum circuit of Fig. 1. This is because the problem lacks all-to-all connectivity, necessitating alignment with both the hardware topology and the algorithm's structure, thus exemplifying a co-design approach. The result, shown in Fig. 4 (b) demonstrates that the algorithm shows about $1.6X-$fold improvement in reducing the circuit runtime even with the smallest analog block of 2 qubits. Moreover, further reduction can be achieved with increasing block sizes and the algorithm is most effective for a non-weighted graph followed by a graph with mixed weights and a fully-non uniform graph. For a fully-non uniform graph, increasing the size of the block (beyond 4 qubits) leads to a disadvantage, since the number of gates required for introducing non-homogeniety surpasses the number needed to realize all the many-body terms. Therefore, the algorithm is problem dependent and demonstrates non-trivially that for a fully non-homogeneous problem (non-uniform MIS) the highest reduction can be achieved with just a 2-qubit analog block [$U_{MS}(\theta, \phi)$].

V. FUTURE DIRECTION

Although, the hardware-specific implementation of DAQC shows the possibility of solving problems involving qubit capacity beyond 50, it is worthwhile to note that this was demonstrated by using a homogeneous multiqubit gate as the analog block applied on the nearest set of qubits. This implies that this predicted improvement is the worst case scenario and, in principle, one can design a programmable inhomogeneous analog block to tackle larger problem instances in trapped ions and the ability to entangle non-nearest neighbour blocks. Recent works on trapped ions have showcased the possibility of fast and programmable (capable of generating inhomogeneous or specific interactions within a set of ions) multiqubit XX gates [44–47] with an infidelity range of $\sim 10^{-5} - 10^{-4}$ upto 4 ions, which can be used for the further development of DAQC in trapped ions. These gates are also referred to as EASE gates [46] of which the XX type GMS gate is a subset. An estimate of the circuit runtime by employing $4-$qubit programmable XX gates can made. Since it is programmable, both homogeneous and inhomogeneous cases can be tackled with equal number of gates. Using DACQO, where we apply the analog blocks to the nearest neighbour ions, we need three layers of XX gates for each quadrupole of ions (to realize spin-spin and cross Pauli terms). The circuit depth will be

$$\text{Depth} = \text{layers}[XX^4(\theta)] + \text{layers}[U_{MS}^2(\theta)] + \text{layers}[U_{X,Y,Z}]$$
$$= 6 + \frac{2(N-4)(N-3)}{N}. \quad (11)$$

In this manner, one achieves an improved scaling for the non-homogeneous problems allowing to solve larger problem instances. Furthermore, the ability to generate analog interactions among non-nearest neighbour ions could lead to further reduction depending upon the number of non-nearest pairs that can be entangled. Suppose we apply 4-qubit analog blocks among non-nearest neighbours, that can generate entanglement among six pairs of ions simultaneously, so that applying M number of such blocks generates $6M$ pairs. Then, the reduced depth will be given by

$$\text{Depth} = 6 + \frac{2[(N-4)(N-3) - 6M]}{N}. \quad (12)$$

Thus, the system size that can be targeted within the coherence time of the device will depend on the number M. This is related to the ability of the hardware to entangle as many non-nearest neighbour ions as possible based on an analog block of a particular size.

Also, as we mentioned before, GMS gates applied over a set of ions beyond nearest-neighbours. This opens the door to produce effective k-local interactions, allowing the implementation of high-order approximations for the AGP, being closer to the exact one, and approaching to the quantum speed limit. Therefore this type of encoding give us a clear path to implement ultrafast algorithms, which are otherwise limited by the technological performance of the GMS interactions.

VI. CONCLUSIONS

We devised a hardware-specific application-dependent quantum algorithm for combinatorial optimization by implementing a digital-analog encoding of the corresponding digitized counterdiabatic quantum optimization algorithm. The aim is to be able to solve larger instances of optimization problems in trapped ions. The encoding is problem dependent, and we tailor the algorithm to solve combinatorial optimization problems such as QUBO. Utilizing the multi-qubit Mølmer-Sørensen interactions in trapped ions as the analog blocks, while supplementing it with single and two-qubit digital steps, we show how to achieve an optimal digital-analog configuration. We design an N-qubit quantum circuit and showcase that for QUBO problems. In the worst case scenario, where analog blocks are applied only to nearest-neighbour qubits or ions, we can target larger problem instances requiring more qubits (≥ 20) while respecting the available qubit coherence time. We developed the algorithm for a general all-to-all connected Ising spin glass model, and studied the minimum fidelity required by the analog blocks to solve them using the selected hardware. For smaller instances (up to 20 qubits), the algorithm can outperform digital quantum simulation with an average minimum analog block fidelity of $98\% - 99\%$ and the extrapolated data (up to 56 qubits) shows that a fidelity of $99\% - 99.5\%$ is sufficient. Moreover, we estimate the scaling of an industry relevant use case problem, the maximum independent set by generating a random instance of 16 nodes, and compare its circuit runtime, when the system is 1) non-weighted, 2) weighted with mixed weights, and 3) weighted with non-uniform weights, against the purely digital simulation. In consequence, we demonstrate that the algorithm reduces the circuit runtime by a factor $\sim 2X$ in all instances, and for the first two instances, further reduction can be achieved by employing a larger analog block. In the third instance, however, employing an analog block larger than 4-qubit does not lead to a better scaling. We also suggest potential advancements through the use of fast, programmable inhomogeneous analog blocks [44, 47]. In summary, our algorithm can reduce the circuit runtime by a minimum factor of $\sim 2X$, enabling the hardware to solve larger problem instances (≥ 20 qubits) within the available coherence time.

We believe that the DACQO paradigm paves the way to develop an intensive and interdisciplinary research in digital-analog quantum computing encoding of industry use cases, to bring quantum advantage to the present of academic and commercial quantum processors.

VII. SUPPLEMENTARY INFORMATION

This section provides additional analyses, and explanations to support the main findings of the manuscript. It is organized into the following subsections to facilitate navigation and comprehension.

VIII. ANALYTICAL CALCULATION OF CD COEFFICIENT FROM THE FIRST ORDER NESTED COMMUTATOR

The first order CD coefficient, α_t is given by

$$\alpha_1 = -\frac{1}{4} \frac{\sum_i h_i^2 + \sum_{i<j} J_{ij}^2}{(1-\lambda)^2 \left(\sum_i h_i^2 + 4 \sum_{i \neq j} J_{ij}^2 \right) + \lambda^2 \left[\sum_i h_i^4 + \sum_{i \neq j} J_{ij}^4 + 6 \sum_{i \neq j} h_i^2 J_{ij}^2 + 6 \sum_{i<j<k} \left(J_{ij}^2 J_{ik}^2 + J_{ij}^2 J_{jk}^2 + J_{ik}^2 J_{jk}^2 \right) \right]}. \tag{13}$$

The detailed calculation of α_t is as follows. In the nested commutator expansion, the approximated CD term is given by

$$H_{cd}^{(l)}(\lambda) = i\dot{\lambda} \sum_{k=1}^{l} \alpha_k(\lambda) \underbrace{[H_{\mathrm{ad}}, [H_{\mathrm{ad}}, \ldots [H_{\mathrm{ad}}, \partial_\lambda H_{\mathrm{ad}}]]]}_{2k-1}, \tag{14}$$

where ℓ is the order of the expansion and α_k are expansion coefficients to be found. The first order CD coefficient can be computed exactly for any Hamiltonian, provided the first two nested commutators $\mathcal{O}_1 = [H_{ad}, \partial_\lambda H_{ad}]$ and $\mathcal{O}_2 = [H_{ad}, \mathcal{O}_1]$, since $\alpha_1 = -\Gamma_1/\Gamma_2$, where $\Gamma_k = \|\mathcal{O}_k\|^2$ such that the CD term takes the form

$$H_{CD} = i\dot{\lambda}\alpha_1\mathcal{O}_1. \tag{15}$$

For the case of the Hamiltonian used in Eq. (2),

$$\mathcal{O}_1 = \left[\sum_i \sigma_i^x, \sum_k h_k \sigma_k^z + \sum_{kl} J_{k<l}\sigma_k^z\sigma_k^z \right]$$
$$= -2i\sum_i h_i\sigma_i^y - 2i\sum_{i<j} J_{ij}\left(\sigma_i^y\sigma_j^z + \sigma_i^z\sigma_j^y\right). \tag{16}$$

Therefore, Γ_1 is given by

$$\Gamma_1 = 4\sum_i h_i^2 + 4\sum_{i<j} J_{ij}^2. \tag{17}$$

Now, we need to calculate the denominator of the equation for α_1, which follows from $\mathcal{O}_2 = (1 - \lambda)\left[\sum_i \sigma_i^x, \mathcal{O}_1\right] + \lambda\left[\sum_i h_i\sigma_i^z + \sum_{i<j} J_{ij}\sigma_i^z\sigma_j^z, \mathcal{O}_1\right]$. Then, we need to evaluate two different commutators. The first one can be written as

$$\left[\sum_i \sigma_i^x, \mathcal{O}_1\right] = 4\sum_i h_i\sigma_i^z + 8\sum_{i<j} J_{ij}\sigma_i^z\sigma_j^z - 8\sum_{ikj} J_{ij}\sigma_i^y\sigma_j^y, \tag{18}$$

whereas the second one is given by the following equation as shown below

$$\left[\sum_i h_i\sigma_i^z + \sum_{i<j} J_{ij}\sigma_i^z\sigma_j^z, \mathcal{O}_1\right] = -4\sum_i\left(h_i^2 + \sum_j J_{ij}^2\right)\sigma_i^x - 8\sum_{i\neq j} h_i J_{ij}\sigma_i^x\sigma_j^z$$
$$- 8\sum_{i<j<k}\left(J_{ij}J_{ij}\sigma_i^x\sigma_j^z\sigma_k^z + J_{ij}J_{jk}\sigma_i^z\sigma_j^x\sigma_k^z + J_{ik}J_{jk}\sigma_i^z\sigma_j^z\sigma_k^x\right). \tag{19}$$

It follows that,

$$\Gamma_2 = 16(1-\lambda)^2\left(\sum_i h_i^2 + 4\sum_{i\neq j} J_{ij}^2\right) + 16\lambda^2\left(\sum_i h_i^4 + \sum_{i\neq j} J_{ij}^4 + 6\sum_{i\neq j}^1 h_i^2 J_{ij}^2 + 6\sum_{i<j<k}\left(J_{ij}^2 J_{ik}^2 + J_{ij}^2 J_{jk}^2 + J_{ik}^2 J_{jk}^2\right)\right). \tag{20}$$

Hence, the CD term is given by

$$H_{cd}^{(1)}(\lambda) = 2\dot{\lambda}\alpha_1(\lambda)\left(\sum_i h_i\sigma_i^y + \sum_{i<j} J_{ij}\left(\sigma_i^y\sigma_j^z + \sigma_i^z\sigma_j^y\right)\right), \tag{21}$$

where

$$\alpha_1 = -\frac{1}{4}\frac{\sum_i h_i^2 + \sum_{i<j} J_{ij}^2}{R(t)}, \tag{22}$$

and

$$R(t) = (1-\lambda)^2\left(\sum_i h_i^2 + 4\sum_{i\neq j} J_{ij}^2\right) + \lambda^2\left(\sum_i h_i^4 + \sum_{i\neq j} J_{ij}^4 + 6\sum_{i\neq j}^1 h_i^2 J_{ij}^2 + 6\sum_{i<j<k}\left(J_{ij}^2 J_{ik}^2 + J_{ij}^2 J_{jk}^2 + J_{ik}^2 J_{jk}^2\right)\right). \tag{23}$$

A. Realizing $n-local$ terms using GMS gates

Here, we show how to obtain $n-local$ terms using GMS gates and local rotations. For the GMS gate, Eq. (6), let $\phi = 0$, then we have $U_{MS}^k(\theta, 0) = e^{-i\theta S_x^2}$ where $S_x = \sum_{j,k}\sigma^x$ which produces the $2-local$ terms between qubits $j-k$. With some algebra, it can be shown that a local operation applied to $l-$th qubit supplemented by GMS gate can produce cross Pauli terms as

$$U_{j,k}(\theta) = e^{[-i\theta S_x^2]}\sigma_l^z e^{[i\theta S_x^2]}. \tag{24}$$

For $j = l$, we have

$$U_{j,k}(\theta)\sigma_l^z U_{j,k}^\dagger(\theta) = [\sigma_l^z\cos(\theta/2) - \sigma_l^y\sigma_k^x\sin(\theta/2)]\delta_{j,l}. \tag{25}$$

For a generalization to obtain $n - local$ terms, we can make use of the pattern of $\sigma_l^{z(y)}$ terms appearing for odd or even iterations given in Table. I: If we apply the transformation n times, the nth iteration will give us

$$I_l^{(n)}(\theta) = \prod_{i_1,i_2} U_{i_1,i_2}(\theta)\sigma_l^z U_{i_1,i_2}^\dagger(\theta) = \sigma_l^z \cos^{n-1}(\theta/2) - \sum_l \sigma_l^y \sigma_{i_2}^x \cos^{n-2}(\theta/2)\sin(\theta/2)$$
$$- \sum_{l \neq i_2} \sigma_l^z \sigma_{i_2}^x \sigma_{i_3}^x \cos^{n-3}(\theta/2)\sin^2(\theta/2) + \sum_{l \neq i_2 \neq i_3} \sigma_l^y \sigma_{i_2}^x \sigma_{i_3}^x \sigma_{i_4}^x \cos^{n-4}(\theta/2)\sin^3(\theta/2)$$
$$\vdots$$
$$+ \mathbf{1}_{\text{even}(n)} \sum_{i_{(j)}} (-1)^{j/2}\sigma_l^y \sigma_{i_2}^x \ldots \sigma_{i_{n-1}}^x \cos^{(n-j)}(\theta/2)\sin^{(j-1)}(\theta/2)$$
$$+ \mathbf{1}_{\text{odd}(n)} \sum_{i_{(j)}} (-1)^{(j-1)/2}\sigma_l^z \sigma_{i_2}^x \sigma_{i_3}^x \ldots \sigma_{i_n}^x \cos^{(n-j)}(\theta/2)\sin^{(j-1)}(\theta/2)$$
$$\vdots$$
$$(-)^{\lfloor n/2\rfloor} \sum_{l \neq i_2, i_2 \neq i_3 \neq \ldots \neq i_n} \sigma_l^y \sigma_{i_2}^x \ldots \sigma_{i_n}^x \sin^{n-1}(\theta/2). \tag{26}$$

Even	Odd	σ_l type	Sign
——	1	σ^z	+
2	——	σ^y	-
——	3	σ^z	-
4	——	σ^y	+
——	5	σ^z	+
6	——	σ^y	-
——	7	σ^z	-
8	——	σ^y	+
——	9	σ^z	+
10	——	σ^y	-

TABLE I. Patterns originating from even and odd local terms.

where $\lfloor n/2\rfloor = m$ and, $m \leq n < m+1$.

B. General building blocks for $n - local$ terms

We can generate different types of $n - local$ terms by performing a basis transformation. We can define a new basis $\hat{u}_j = \hat{i}\cos\phi + \hat{j}\sin\phi$, so that

$$\sum_j \sigma_j^u = \sum_j \sigma_j^x \cos\phi + \sum_j \sigma_j^y \sin\phi. \tag{27}$$

Substituting in Eq. (6) we obtain

$$U_{MS}^n(\theta,\phi) = \exp\left[-\frac{i\theta}{4}(S_j)^2\right], \tag{28}$$

where $S_j = \sum_j^n \sigma_j^u$. Now, we can express the gate as

$$U_{j,k} = \prod_{j,k} \exp\left[-\frac{i\theta}{4}\sigma_j^u \sigma_k^u\right], \tag{29}$$

The generalized operation of this GMS gate to any pauli operator σ_j^α can be given as

$$I(\theta) = \prod_{j,k} e^{[-i\theta\sigma_j^u \sigma_k^u]} \sigma_j^\alpha e^{[i\theta\sigma_j^u \sigma_k^u]}, \tag{30}$$

where $\alpha = x, y, z$. Following the method in the last section and using the following commutation relations

$$[\sigma_j^x, \sigma_j^u] = 2i\sigma_j^z \sin\phi, \quad [\sigma_j^y, \sigma_j^u] = -2i\sigma_j^z \cos\phi,$$
$$[\sigma_j^z, \sigma_j^u] = 2i\sigma_j^\perp, \quad \text{where } \sigma_j^\perp = \sigma_j^y \cos\phi - \sigma_j^x \sin\phi, \tag{31}$$

and

$$[\sigma_j^u, [\sigma_j^x, \sigma_j^u]] = -4\sigma_j^\perp \sin\phi, \quad [\sigma_j^y, \sigma_j^u] = 4\sigma_j^\perp \cos\phi,$$
$$[\sigma_j^u, [\sigma_j^z, \sigma_j^u]] = -4\sigma_j^z, \quad \text{where } \sigma_j^\perp = \sigma_j^y \cos\phi - \sigma_j^x \sin\phi, \tag{32}$$

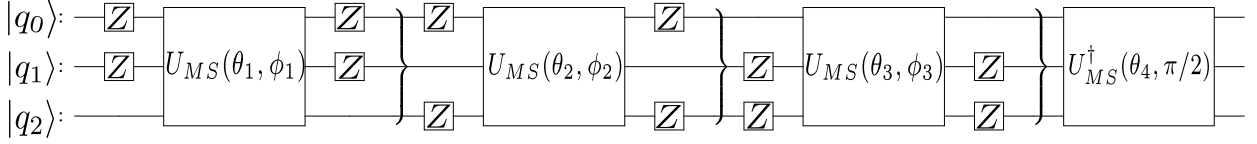we can obtain the generalization to n particles giving us

FIG. 5. Quantum circuit to realize in-homogenous $2-local$ terms in a simple example of a 3-qubit Hamiltonian, Eq. 7 utilizing 3-qubit MS gates as analog blocks and Pauli Z gates as local rotations.

$$\prod_{j,k} e^{\left[-i\theta\sigma_j^u\sigma_k^u\right]}\sigma_j^{\perp} e^{\left[i\theta\sigma_j^u\sigma_k^u\right]} = \quad +\mathbf{1}_{\text{even}(n)}\sum_{i_{(j)}}(-1)^{j/2}\sigma_l^z\sigma_{i_2}^u\ldots\sigma_{i_{n-1}}^u\cos^{(n-j)}(\theta/2)\sin^{(j-1)}(\theta/2)$$

$$+\mathbf{1}_{\text{odd}(n)}\sum_{i_{(j)}}(-1)^{(j-1)/2}\sigma_l^{\perp}\sigma_{i_2}^x\sigma_{i_3}^u\ldots\sigma_{i_n}^u\cos^{(n-j)}(\theta/2)\sin^{(j-1)}(\theta/2)$$

$$(33)$$

C. Mapping the target Hamiltonian coefficients to the angles of GMS gate

We perform time-evolution of the full CD Hamiltonian Eq. (7) using trotterization obtaining

$$\exp\{H'(\lambda)t/n\} = \left\{\exp\left(-i\theta^{x,x}\sum_{i<j=1}^{N}\sigma_i^x\sigma_j^x\right)\right.$$

$$\cdot\exp\left(-i\theta^x\sum_{i=1}^{N}\sigma_i^x\right)\cdot\exp\left(-i\theta^z\sum_{i=1}^{N}\sigma_i^z\right)$$

$$\cdot\exp\left(-i\theta^{x,y}\sum_{i,j}^{N}\sigma_y^i\sigma_x^j\right)$$

$$\left.\cdot\exp\left(-i\theta^{x,y}\sum_{i<j=1}^{N}\sigma_x^i\sigma_y^j\right)\cdot\exp\left(-i\theta^y\sum_{i=1}^{N}\sigma_y^i\right)\right\}^n \quad (34)$$

where $\theta^{x,x} = \lambda(t)Jt/n$, $\theta^x = \lambda(t)ht/n$, $\theta^z = (1-\lambda(t))t/n$, $\theta^{x,y} = \theta^{y,x} = 2J\dot{\lambda}\alpha_1(t)t/n$, $\theta^y = 2\dot{\lambda}\alpha_1(t)ht/n$, where t is the evolution time and n is the number of trotter steps. To map these Hamiltonian coefficients to the angles of the GMS gate we expand the GMS gate Hamiltonian Eq. (6) obtaining $\theta\cos^2(\phi)/2 = \theta^{x,x}$ and $\theta\cos(\phi)\sin(\phi)/4 = \theta^{x,y}$ using which we can realize the $2-local$ terms.

D. Inhomogeneous case

Inhomogeneous QUBO Hamiltonians can be generated by the Homogeneous analog blocks as explained in the main text. A simple example of realizing $2-local$ terms of a 3 qubit QUBO problem is shown in Fig. 5. To introduce inhomogeneity across any k-qubit interactions within a N-

qubit system, it is necessary to employ $k(k-1)/2$ analog blocks [14] along with local rotations. Therefore, in this example, we need 3 analog blocks. The unitary generated by the quantum circuit is

$$U(t_i) \approx exp\left[\frac{i}{2}\left(t_1\sigma_1^x\sigma_2^x + t_2\sigma_2^x\sigma_3^x + t_3\sigma_1^x\sigma_3^x\right.\right.$$

$$+t_4(\sigma_1^y\sigma_2^y + \sigma_2^y\sigma_3^y + \sigma_1^y\sigma_3^y) + t_5(\sigma_1^x\sigma_2^y + \sigma_2^x\sigma_1^y)$$

$$+t_6(\sigma_1^x\sigma_3^y + \sigma_3^x\sigma_1^y) + t_7(\sigma_2^x\sigma_3^y + \sigma_3^x\sigma_2^y))\right]$$

$$\left.\left.-t_8(\sigma_1^y\sigma_2^y + \sigma_2^y\sigma_3^y + \sigma_1^y\sigma_3^y)\right)\right]. \quad (35)$$

where

$t_1 = -\theta_1\cos^2\phi_1 + \theta_2\cos^2\phi_2 + \theta_3\cos^2\phi_3$,
$t_2 = -\theta_1\cos^2\phi_1 + \theta_2\cos^2\phi_2 - \theta_3\cos^2\phi_3$,
$t_3 = \theta_1\cos^2\phi_1 - \theta_2\cos^2\phi_2 + \theta_3\cos^2\phi_3$,
$t_4 = -\theta_1\sin^2\phi_1 - \theta_2\sin^2\phi_2 - \theta_3\sin^2\phi_3$,
$t_5 = -\theta_1\cos\phi_1\sin\phi_2 + \theta_2\cos\phi_1\sin\phi_2 + \theta_3\cos\phi_1\sin\phi_2$,
$t_6 = \theta_1\cos\phi_1\sin\phi_3 - \theta_2\cos\phi_1\sin\phi_3 + \theta_3\cos\phi_1\sin\phi_3$,
$t_7 = \theta_1\cos\phi_2\sin\phi_3 + \theta_2\cos\phi_2\sin\phi_3 - \theta_3\cos\phi_2\sin\phi_3$,
$t_8 = t_4$.

As the problem size grows, an increasing number of analog blocks are required per ion set to induce non-homogeneity, which suggests that enlarging the block size may not effectively reduce circuit depth. To determine the optimal block size for maximum circuit compression, we conducted an analysis, illustrated in Fig. 4. We discovered that the ideal size for the analog block in this specific problem is a 2-qubit GMS gate $U_{MS}^2(\theta,\phi)$. Expanding the block beyond four qubits tends to be counterproductive. Furthermore, if hardware allows for the generation of non-homogeneous interactions, enabling what are known as programmable quantum gates [44, 47], then the scalability improvements for

inhomogeneous problems could match those of homogeneous ones.

---

[1] L. Andrew, *Ising formulations of many NP problems*, Frontiers in Physics, **2**, (2014).

[2] A. D. King et al., *Quantum critical dynamics in a 5,000-qubit programmable spin glass*, Nature **617**, 61–66 (2023).

[3] L Henriet, L. Beguin, A. Signoles, T. Lahaye, A. Browaeys G.O. Reymond, and C. Jurczak, *Quantum computing with neutral atoms*, Quantum **4**, 327 (2020).

[4] A. Gomez Cadavid, I. Montalban, A. Dalal, E. Solano, and N. N. Hegade, *Efficient DCQO Algorithm within the Impulse Regime for Portfolio Optimization*, arXiv:2308.15475 **[quant-ph]** (2023).

[5] H. Guan, F. Zhou, F. Albarrán-Arriagada, X. Chen, E. Solano, N. N Hegade, H.-L. Huang, *Single-Layer Digitized-Counterdiabatic Quantum Optimization for p-spin Models*, arXiv:2311.06682 **[quant-ph]** (2023).

[6] N. Moll et al., *Quantum optimization using variational algorithms on near-term quantum devices*, Quantum Sci. Technol. **3**, 030503 (2018).

[7] N. Wiebe, D. Braun, and S. Lloyd, *Quantum Algorithm for Data Fitting*, Phys. Rev. Lett. **109**, 050505 (2012), arXiv:1204.5242.

[8] A. Montanaro, *Quantum algorithms: an overview*, npj Quantum Inf. **2**, 15023 (2016), arXiv:1511.04206.

[9] Niklas Pirnay, Vincent Ulitzsch, Frederik Wilde,Jens Eisert, and Jean-Pierre Seifert, *An in-principle super-polynomial quantum advantage for approximating combinatorial optimization problems via computational learning theory*, Sci. Adv. **10**, 11 (2024)

[10] N. N Hegade, K. Paul, Y. Ding, M. Sanz, F. Albarrán-Arriagada, E. Solano and Xi Chen, *Shortcuts to Adiabaticity in Digitized Adiabatic Quantum Computing*, Phys. Rev. Applied **15**, 024038 (2021).

[11] N. N. Hegade, X. Chen, and E. Solano, *Digitized counterdiabatic quantum optimization*, Phys. Rev. Res. **4**, L042030 (2023).

[12] P. Chandarana, N. N. Hegade, I. Montalban, E. Solano, and X. Chen, *Digitized Counterdiabatic Quantum Algorithm for Protein Folding*, Phys. Rev. Applied **20**, 014024 (2023).

[13] N. N. Hegade, and Enrique Solano, *Digitized-counterdiabatic quantum factorization*, arXiv:2301.11005v1 (2023).

[14] A. Parra-Rodriguez, P. Lougovski, L. Lamata, E. Solano, and M. Sanz, *Digital-analog quantum computation*, Phys. Rev. A **101**, 022305 (2020).

[15] Z. Davoudi, N. M. Linke, and G. Pagano, *Toward simulating quantum field theories with controlled phonon-ion dynamics: A hybrid analog-digital approach*, Phys. Rev. Research **3**, 043072 (2021).

[16] 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*, Sci. Rep. **6**, 27836 (2016).

[17] J. Yu, J. C. Retamal, M. Sanz, E. Solano, and F. Albarrán-Arriagada, *Superconducting circuit architecture for digital-analog quantum computing*, EPJ Quantum Technol. **9**, 9 (2022).

[18] M. Garcia-de-Andoin, Á. Saiz, P. Pérez-Fernández, L. Lamata, I. Oregi, and M. Sanz, *Digital-Analog Quantum Computation with Arbitrary Two-Body Hamiltonians*, arXiv:2307.00966 (2023).

[19] A. Martin, L. Lamata, E. Solano, and M. Sanz, *Digital-analog quantum algorithm for the quantum Fourier transform*, Phys. Rev. Research **2**, 013012 (2020).

[20] D. Headley, T. Müller, A. Martin, E. Solano, M. Sanz, and F. K. Wilhelm, *Approximating the quantum approximate optimization algorithm with digital-analog interactions*, Phys. Rev. A **106**, 042446 (2022).

[21] A. Martin, R. Ibarrondo, and M. Sanz, *Digital-Analog Co-Design of the Harrow-Hassidim-Lloyd Algorithm*, Phys. Rev. Applied **19**, 064056 (2023).

[22] 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).

[23] D. Hayes, S. T Flammia and M. J Biercuk *Programmable quantum simulation by dynamic Hamiltonian engineering*, New Journal of Physics **16**, 083027(2014).

[24] F. Rajabi, S. Motlakunta, Chung-You Shih, N. Kotibhaskar, Q. Quraishi, A. Ajoy, and R. Islam *Dynamical Hamiltonian engineering of 2D rectangular lattices in a one-dimensional ion chain*, npj Quantum Information **5**, 32 (2019).

[25] S. Kumar, N. N. Hegade, E. Solano, F. Albarrán-Arriagada, and G. A. Barrios, *Digital-analog quantum computing of fermion-boson models in superconducting circuits*, arXiv:2308.12040 **[quant-ph]** (2023).

[26] C. Chevallier, J. Vovrosh, J. de Hond, M. Dagrada, A. Dauphin, and V. E. Elfving, *Variational protocols for emulating digital gates using analog control with always-on interactions*, arXiv:2402.07653 **[quant-ph]** (2024).

[27] A. Sørensen and K. Mølmer, *Quantum Computation with Ions in Thermal Motion*, Phys. Rev. Lett. **82**, (1971).

[28] K. Mølmer and A. Sørensen, *Multiparticle Entanglement of Hot Trapped Ions*, Phys. Rev. Lett. **82**, 1835(1971).

[29] T. Monz et al, *14-Qubit Entanglement: Creation and Coherence*, Phys. Rev. Lett. **106**, 130506 (2011).

[30] D. Maslov and Y. Nam, *Use of global interactions in efficient quantum circuit constructions*, New Journal of Physics **20**, 033018 (2018).

[31] M. Demirplak, and S. A. Rice, *Adiabatic Population Transfer with Control Fields*, J. Phys. Chem. A **107**, 46, 9937–9945 (2003).

[32] M. V. Berry, *Transitionless quantum driving*, J. Phys. A: Math. Theor. **42** 365303 (2009).

[33] X. Chen, I. Lizuain, A. Ruschhaupt, D. Guéry-Odelin, and J. G. Muga, *Shortcut to Adiabatic Passage in Two- and Three-Level Atoms*, Phys. Rev. Lett. **105**, 123003 (2010).

[34] A. del Campo, *Shortcuts to Adiabaticity by Counterdiabatic Driving*, Phys. Rev. Lett. **111**, 100502 (2013).

[35] D. Sels and A. Polkovnikov, *Minimizing irreversible losses in quantum systems by local counterdiabatic driving*, PNAS **114**, E3909 (2017).

[36] P. W. Claeys, M. Pandey, D. Sels, and A. Polkovnikov, *Floquet-Engineering Counterdiabatic Protocols in Quantum Many-Body Systems*, Phys. Rev. Lett. **123**, 090602 (2019).

[37] Q. Xie, K. Seki, and S. Yunoki,*Variational counterdiabatic driving of the Hubbard model for ground-state preparation*, Phys. Rev. B **106**, 155153 (2022).

[38] Z. Cai, C.Y. Luan, L. Ou, H. Tu, Z. Yin, J.N. Zhang and K. Kim, *Entangling gates for trapped-ion quantum computation*

and quantum simulation, J. Korean Phys. Soc. **82**, 882–900 (2023).

[39] J. Casanova, A. Mezzacapo, L. Lamata, and E. Solano, *Quantum Simulation of Interacting Fermion Lattice Models in Trapped Ions*, Phys. Rev. Lett. **108**, 190502 (2012).

[40] *IonQ: Algorithmic Qubits*, https://ionq.com/resources/algorithmic-qubits-a-better-single-number-metric.

[41] *Qiskit Aer: Simulator Backends*, Qiskit Aer API Documentation.

[42] A. Bermudez, *et al.*, *Assessing the Progress of Trapped-Ion Processors Towards Fault-Tolerant Quantum Computation*, Phys. Rev. X **7**, 041061 (2023).

[43] R. Blümel, A. Maksymov, and M. Li, *Toward a Mølmer Sørensen Gate With .9999 Fidelity*, arXiv:2311.15958 **[quant-ph]** (2023).

[44] Y. Shapira, L. Peleg, D. Schwerdt, J. Nemirovsky, N. Akerman, A. Stern, A. B. Kish, and R. Ozeri, *Fast design and scaling of multi-qubit gates in large-scale trapped-ion quantum computers* arXiv:2307.09566 (2023).

[45] N. Grzesiak et al, *Efficient arbitrary simultaneously entangling gates on a trapped-ion quantum computer*, Nat Commun **11**, 2963 (2020).

[46] N. Grzesiak1, A. Maksymov1, P. Niroula, and Y. Nam *Efficient quantum programming using EASE gates on a trapped-ion quantum computer*, Quantum **6**, 634 (2022).

[47] Y. Shapira, R. Shaniv, T. Manovitz, N. Akerman, L. Peleg, L. Gazit, R. Ozeri, and A. Stern. *Theory of robust multiqubit nonadiabatic gates for trapped ions*, Phys. Rev. A 101, 032330

[48] Tom Manovitz, Yotam Shapira, Nitzan Akerman, Ady Stern, and Roee Ozeri, *Quantum Simulations with Complex Geometries and Synthetic Gauge Fields in a Trapped Ion Chain*, PRX Quantum **1**, 020303 (2020).