Characterization and compilation techniques for CR gates

Ashish Kakkar
Q-CTRL, Los Angeles, CA USA
(Dated: 15 Jul 2025)

In quantum computing, parametrized entangling gates have recently gathered significant interest. Parameterized gates can be shorter in duration (compared to a maximally entangling gate) and can facilitate shallower circuit syntheses. Fast and efficient calibration of a discrete set of shorter, high fidelity fractionally entangling gates and the construction of optimal circuit synthesis schemes that leverage the richer gate set are open problems that are addressed in this work. We develop a novel gate calibration technique that is highly resource efficient in terms of both QPU and CPU time. It relies on building a faithful model for the gate Hamiltonian using a set of offline characterization experiments. Our specialized compiler features a unitary synthesis algorithm that can use such gate Hamiltonian models to synthesize circuits with shorter depths and higher fidelity. We demonstrate the power of this fully automated technique experimentally on cross resonance gates on IBM Quantum devices by augmenting the existing entangling gate set with shorter duration *efficient* gates. Our approach introduces overhead of only 3 min of QPU time and 30s of classical post-processing time. Algorithmic benchmarking of these techniques shows over 3 times higher fidelity of the quantum Fourier transform algorithm for up to 26 qubits and up to 9 times lower mean-square-error (MSE) in the estimation of expectation values in a trotterized Hamiltonian simulation of the 1-D transverse field Ising model on 25 qubits.

I. INTRODUCTION

An essential component in quantum information processing is a layer of abstraction that maps primitive instructions determined by the physical system (pulses) to a set of easily programmable abstract operations (logical gates). This mapping relies on the *native gate set*: the set of all logical operations natively available on a hardware device. A gate set is universal when it can be used to implement any unitary to arbitrary precision [1–3].

However, not all universal gate-sets are equivalent in practice; the compilation and execution of a quantum algorithm can vary significantly for different gate sets. This is especially important when one considers the time duration of these gates and their characteristic errors. On contemporary quantum computing platforms, gate sets typically include a number of single-qubit rotations and a single entangling two-qubit gate. There has been remarkable progress in designing fast entangling two-qubit gates, while suppressing parasitic coherent errors, using a wide range of entanglement generating interactions [4–10].

There is mounting evidence that more expressive gate sets (including multiple entangling gates, beyond the minimum required for universality) enables the execution of near-term algorithms with higher fidelity. This has led to the design of novel entangling gates and calibration techniques, notably in the development of parametric entangling gates [11–14]. These families of parametric entangling gates generally have shorter durations, which affects the execution fidelity of the circuit in two ways: Firstly, a reduction in the overall circuit duration that arises from implementing shorter gates should increase circuit fidelity, especially for deep circuits that approach the finite qubit coherence limit. Secondly, a favorable choice of entangling gates in the gate set may allow algorithms to be implemented via shorter gate sequences (i.e. fewer gates), which leads to a reduction in gate errors accumulated throughout the circuit.

However, several technical challenges make it difficult to construct a richer entangling gate set, given the constraints imposed by a physical device. Calibrating a two-qubit interaction to a high-fidelity target unitary is resource-intensive [15–19]. Typically, several time-varying control parameters that characterize the pulse profile(s) need to be carefully tuned to achieve the target unitary. Even when calibration protocols yield high-fidelity gates, it is crucial that the calibration time budget remains minimal [20]. The temporal stability of the calibration under device parameter drift imposes a finite useful lifetime for any calibrated gate [21]. Ideally, calibration routines should consume a minimal portion of the calibration lifetime, since it is desirable that device time be mostly dedicated to execution of algorithmic jobs. For continuous-parameter entangling gates, there is a trade-off between the fidelity gains due to shorter gate sequences and the error incurred due to inaccurate interpolation of control parameters.

From the perspective of quantum compiler design, there is a need to build techniques that can leverage the advantages that are offered by a richer gate set. Most contemporary compilers [22–24] perform the synthesis of generic quantum circuits using standard entangling gates. Initial steps towards compiling around scaled entangling gates were taken recently in [25, 26]. There has also been recent work in outlining sets of circuit synthesis schemes that can leverage such gates [27, 28].

In this work, we introduce a novel approach to generating and deploying an additional discrete set of fractionally entangling gates. Our approach to diversifying the entangling gate set is based on a shift of perspective: rather than *calibration*, we focus on compiler-

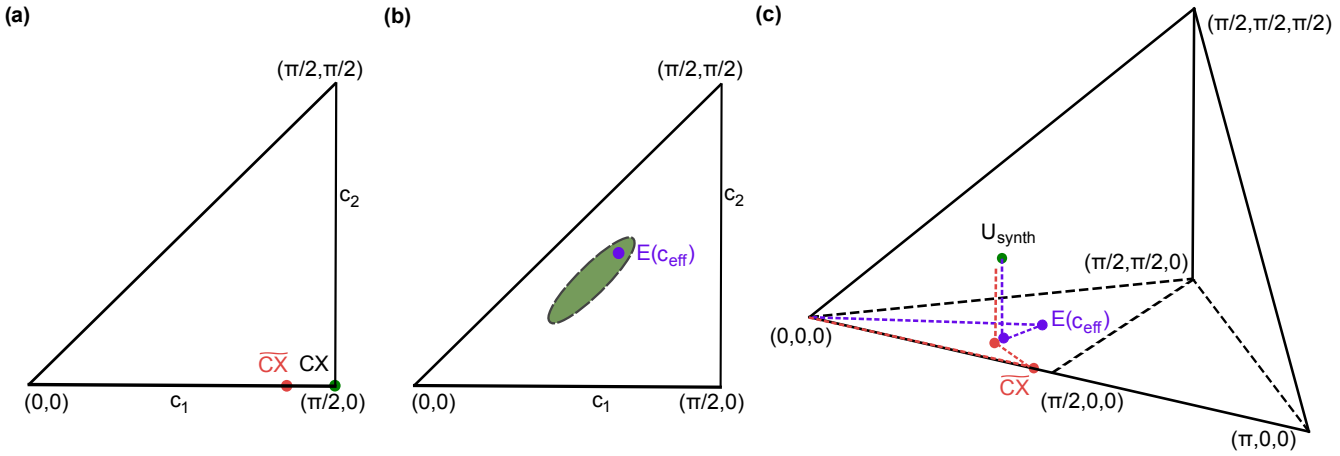

FIG. 1: **(a)-(b)** Two different schemes for gate design depicted on a section of the $SU(4)$ Weyl chamber. **(a)** A calibration based scheme, the target unitary (green point) in this case being a (maximally entangling) CX gate. A pulse optimization routine (closed or open loop) is performed to yield an approximate (noisy) realization of the unitary (red point). Errors and inaccuracies in the procedure eventually can lead to a noisy gate definition being constructed. **(b)** A characterization based scheme for fractionally entangling gates, where a pulse is loosely designed to generate a unitary within a bounded region (green), which represents the class of models being considered for its generating Hamiltonian. Off-line characterization of the model allows a precise determination (purple point) of its co-ordinates c_{eff}, and using local rotations the pulse is converted into a gate definition $E\left(c_{\text{eff}}\right)$ which can be used for unitary synthesis. **(c)** Synthesis of arbitrary 2-qubit unitary blocks in the Weyl chamber: a 2-qubit unitary U_{synth} (green point) needs to be synthesized via a set of basis gates. Steering of these trajectories is achieved by interleaving single-qubit gates between up to three repetitions of a 2-qubit gate. In the red trajectory, a sequence of three CX gates are used to reach the target U_{synth} while in purple a sequence of 3 $E\left(c_{\text{eff}}\right)$ gates (which may be different) are steered to the target U_{synth}.

understandable *characterization*. The development of gate characterization techniques such as Gate Set Tomography is an active area of research [29], and in this work we build on existing techniques that fall broadly under Hamiltonian tomography, which allow us to build faithful models of the generating Hamiltonian corresponding to the unitary entangling gate [8, 30–32].

Rather than calibrating a pulse to produce a particular desired target unitary, we instead send an approximately calibrated pulse to the quantum device, which is expected to realize the generating Hamiltonian of the unitary within a known bounded range. We then perform an efficient characterization procedure to precisely identify the corresponding unitary operation that is realized. Such off-line characterization, as opposed to online closed-loop calibration, is highly resource efficient in terms of both QPU time and classical postprocessing time. The result of this procedure is generalized gate definition, i.e. a map from the pulse to the unitary which includes the contribution of terms that may typically be called coherent error terms. However, we make no attempt to calibrate these errors towards zero; instead, we precisely measure the relevant Hamiltonian terms. For deployment of these gates in quantum algorithms, we rely on compilation using these gate definitions in addition to existing available gates. In particular, our compiler features a unitary synthesis algorithm that is able to use such gate definitions efficiently to construct circuits.

In Figure 1, we depict this procedure within the mathematical framework of locally equivalent gates. Each 2-qubit unitary can be assigned to a point with canonical coordinates (c_1, c_2, c_3) in the Weyl chamber, where a point represents an equivalence class of locally equivalent gates. Maximally entangling gates, such as CX gates and SWAP gates, are special edge points in this space with coordinates $(\pi/2, 0, 0)$ and $(\pi/2, \pi/2, \pi/2)$, respectively. A traditional procedure for fine-tuning the calibration of a gate manipulates pulse level instructions to realize an ideal target unitary with measured deviations from the target unitary coordinates from *non-local coherent gate errors*. Quantum control techniques such as and closed-loop optimization attempt to decrease these deviations with each iteration, as in Figure 1(a). However, for fractionally entangling gates, one only needs to determine the coordinates corresponding to the coarsely calibrated pulse schedule. These coordinates have a bounded range of expected values that can be determined from a unitary model of the pulse schedule with a bounded set of model coefficients, as in Figure 1(b). Our approach allows for the synthesis of target unitaries using a minimal discrete sequence of these characterized gates, interleaved with single-qubit gates. These characterized gates definitions are useful once the set of their coordinates, c_{eff}, is determined. To realize a unitary target within a quantum algebra, a set of *steering* single-qubit gates is interleaved with up to three repetitions of the characterized gate, as

in Figure 1(c).

Benchmarking of these gates is performed via an evaluation on algorithmic tasks, since the performance improvements our non-standard gate definitions offer are intimately tied to circuit contexts that appear in the algorithm and these are not standard Clifford gates. Gate fidelity measures such as such as interleaved Randomized Benchmarking (iRB) [33] are not used, as they tend to overlook the contribution of coherent errors terms such as crosstalk, which are precisely the kind of errors that are suppressed by our protocol and are individualistic rather than holistic measures.

The manuscript is organized as follows: in Section II, we describe the construction of control pulses used to create the gates used in this work, followed by Hamiltonian model building techniques. In Section III we describe the procedure to build a gate definition that can be utilized by our compiler. We also describe a set of synthesis motifs that can use such gate definitions to compile shorter and higher-fidelity circuits. Finally in section IV, we demonstrate the power of this fully automated technique. In particular, we present algorithmic benchmarking on IBM Quantum devices, with enriched gate-sets. We observe a >3X higher success probability of the quantum Fourier transform algorithm for up to 26 qubits (compared to only using default available calibrated gates) and up to 9 times lower mean square error (MSE) in the estimations of expectation values in a trotterized Hamiltonian simulation of the 1D transverse field Ising model on 25 qubits.

II. PULSE CONSTRUCTION AND EFFECTIVE HAMILTONIAN LEARNING

An important challenge in improving 2-qubit gate fidelities across superconducting qubit architectures is suppressing coherent noise. The cross-resonance (CR) interaction [34] used by the IBM Quantum devices we use for our experimental work is realized in fixed frequency transmons using microwaves to drive the qubit 0 (control) resonantly at the frequency of the target qubit 1. In addition, qubit 1 is resonantly driven with a cancelation tone to cancel coherent single-qubit errors on the target. In the regime of operation, the transmons are weakly anharmonic and weakly driven; as a result, in addition to the desired interaction in the 2-qubit subspace, interactions between the higher energy levels of the transmon necessarily take place. An effective Hamiltonian description incorporates these interactions by projecting them onto the relevant 2-qubit subspace. To arrive at such a description, one can use Schriefer Wolfe Perturbation theory [35] in the common drive frame of the target and control to get:

$$H_{\text{eff}}(\omega_{ab}) = \frac{1}{2}(\omega_{zx}ZX + \omega_{zy}ZY + \omega_{zz}ZZ + \omega_{ix}IX + \omega_{iy}IY + \omega_{iz}IZ + \omega_{zi}ZI) \quad (1)$$

The coefficients in (1) typically depend non-linearly on the control parameters (such as the drive amplitude). The Echoed Cross Resonance (ECR) gate implements the unitary generated by (1) while exploiting an echo scheme on the control qubit to cancel unwanted coefficients [8]:

$$\text{ECR}_\theta = e^{-it_\theta/2H_{\text{eff}}(\omega_{ab})}(XI)\, e^{it_\theta/2H_{\text{eff}}(\omega_{ab})}(XI) \quad (2)$$

We remind the reader that to implement a CX gate, the ideal gate unitary that can be used is $\text{ECR}_{\pi/2} = e^{-i\pi/4ZX}$. In order to calibrate to this target unitary, control parameters in (2) need to be tuned such that $t_{\pi/2}\ \omega_{zx} = \pi/2$ and all other coefficients are zero.

This is a single point in a family of pulse schedules parametrized by θ, obtained by scaling the area of the calibrated echoed cross-resonance pulse provided by the backend [25, 26]. Scaling according to $t_\theta = \frac{2\theta}{\pi}t_{\pi/2}$ gives rise to the ideal gate unitary $e^{-i\theta/2ZX}$, under the underlying assumptions that there are no nonlinearities (ω_{ab} stay the same while scaling the pulse) and the initial pulse was calibrated with no errors.

With the goal of taking into account coherent errors in the gate definition itself, we demonstrate our method by directly scaling the CR pulse provided by the backend as well as eliminate the echo structure to get the Scaled-CR gate (SCR):

$$\text{SCR}_\theta = e^{-it_\theta H_{\text{eff}}(\omega_{ab})} = e^{-iH_{\text{eff}}(\nu_{ab})} \quad (3)$$

For convenience, we switch to dimensionless units where $t_\theta\omega_{ab} = \nu_{ab}$. Since this pulse is constructed by deforming a calibrated backend ECR pulse with $t_{\pi/2}$, this calibration data is implicitly used in the definition (3), via $t_\theta = \frac{2\theta}{\pi}t_{\pi/2}$.

SCR_θ has been directly defined in terms of its generating Hamiltonian $H_{\text{eff}}(\nu_{ab})$. The ideal unitary implemented by SCR_θ, is at this point, unknown since the pulse it is generated by differs from its original calibration. The unitary will only be determined after the parameters ν_{ab} have been learned via a set of characterization experiments.

The design principle we use is that characterization is easier than calibration, in terms of both QPU and classical resources. The non-linear dependence of the realized unitary on control parameters is no longer problematic, since we do not need to tune them via closed-loop optimization to reach a particular target value of ν_{zx} (while tuning the remaining ν_{ab} to zero). Instead, our protocol considers the set ν_{ab} to be learnable parameters that are experimentally reconstructed from a predefined set of Hamiltonian characterization experiments. An effective Hamiltonian analysis provides the set of terms that should be included in the model $H_{\text{eff}}(\nu_{ab})$ to describe the unitary evolution of the gate as well as a good set of initial points and bounds on the parameters ν_{ab}. The only requirement is that: the *measured* effective Hamiltonian should only be in the 2-qubit subspace defined by our ansatz (1), which is an experimental consistency check.

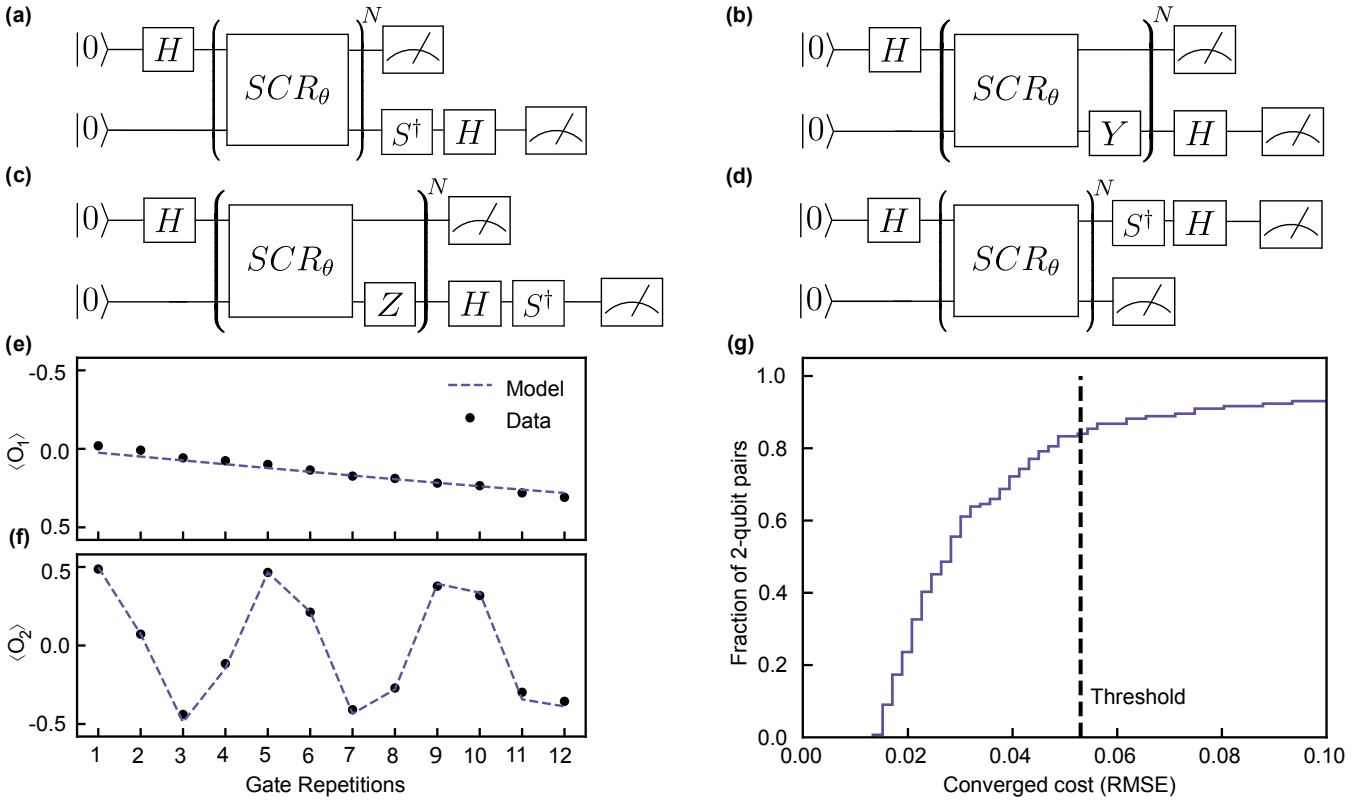

FIG. 2: (**a** - **d**) The set of Hamiltonian tomography sequences used to determine the learnable parameters ν_{ab} that describe the scaled cross resonance (SCR) pulse/gate defined in (3). (**e**) Fitting the model described by Eq (1) to the data generated by circuit (**a**) with RMSE 0.02. (**f**) The distribution of final converged costs (RMSE) of fit between data and model (including all circuits (**a** - **d**) for all 144 edges between coupled qubits of the 127 qubit device ibm brisbane. A majority (85%) of the SCR gates model optimization costs converge to an RMSE below an empirically set threshold. The converged parameters along with the SCR pulse are now to be used to construct a new basis gate $XX\,(\pi/4+\epsilon)$ using Algorithm 2

Conventionally, many of the coefficients such as ν_{zz} would be identified as coherent error terms (cross-talk). However, in our protocol, these terms are coefficients of a model of the generating Hamiltonian for the new unitary gate. Characterization, rather than calibration, is sufficient only when the compiler can synthesize operations around such non-standard basis gates. The procedure to use non-standard unitary gate definitions (such as (3) with arbitrary coefficients ν_{ab}) to synthesize quantum circuits is described in detail in III. While we focus on the particular entangling interaction described by (1) which was used for the experiments in this work, the same principle is applicable for a different entangling interaction starting from a different effective Hamiltonian description as a starting point.

To perform model learning of an ansatz for H_{eff}, we tailor a set of error amplification sequences. Fig 2(a-d) depicts the circuits that are used to measure the Hamiltonian coefficients ν_{ab} for the pulse defined in (3). The sequences are constructed by repeating the gate N times and measuring Rabi-like oscillations on both the target and control qubit. Error amplifying echoes are interleaved into the repetition sequence to selectively amplify coefficients. Measurement of each circuit provides a set of two probe observables $\langle O_i \rangle$. In Appendix D we describe the conventions used to map the set of probe observables $\langle O_i \rangle$ to experimental measurements. The properties of the set of sequences that make them a favorable choice are that they are minimal and complete to reconstruct the full set of Hamiltonian coefficients ν_{ab}. These sequences are highly sensitive for the parameters that are

Algorithm 1 Model Learning

1: Construct pulse SCR_θ by deforming a pre-calibrated gate with a model ansatz (3).
2: Execute pre-determined set of characterization sequences 2 (**a** – **d**)
3: Compute expectation values of observables $\mathbf{O^E_{exp}}$ from experimental data.
4: Determine ν_{ab} minimizing $C = |\mathbf{O^E_{exp}} - \mathbf{O^E_{model}}(\nu_{\mathbf{ab}})|^{\mathbf{2}}$.
5: If $C \leq$ threshold, use ν_{ab} to start Algorithm 2.

expected to be small, ie. $(\nu_{zy}, \nu_{iy}, \nu_{zz}, \nu_{iz})$. The amplification in the corresponding O_j increases with N as $\langle O_j(N)\rangle \sim |N|\nu_{ab}$.

These circuit sequences are executed on all qubit-pairs on the device, with each pulse to be added to the gate-set being parameterized by a target angle θ. The raw experimental probability distributions are obtained given a set of experimental observable expectation values O_j^{exp} that are calculated offline. Optimization of the model parameters ν_{ab} is done to minimize the cost function $C = \sum_j |O_j^{exp} - O_j^{model}(\nu_{ab})|^2$. If this cost converges to a value below an empirically set threshold, a new gate is added to the gate-set. This procedure is fast and highly parallelized, with the optimization being performed via the CMA-ES algorithm implemented in Q-CTRL Boulder Opal [36]. To minimize the total number of experiments needed to characterize all 2-qubit pairs on the device, characterization experiments for multiple 2-qubit pairs is performed simultaneously. Viewing each experiment that characterizes a 2-qubit pair as an edge-coloring of the connectivity graph of the device, the minimum number of independent experiments that are needed to cover all connected qubit-pairs is the minimum edge coloring of the device topology (3 for the heavy-hex connectivity). In Appendix E we describe in detail the breakdown of the QPU resource usage which allows this procedure to add a new gate to the full device in under 3min of QPU time. In Figure 2(e) we show experimental data and fits of a successful instance of the model learning procedure and in Figure 2(f) we show the results of a device-wide model learning procedure that successfully characterizes a new pulse $\text{SCR}_{\pi/4}$ for 85% of the 144 2-qubit pairs on the device IBM Brisbane. The model learning procedure described in this section is summarized in the form of Algorithm 1 which is the first step that is to be executed for each new gate to be added to the gate set.

III. COMPILER ARCHITECTURE DESIGN FOR EFFICIENT GATES

So far we have focused on techniques that allow us to construct pulses that give rise to non-standard fractionally entangling gates, and characterization routines that return parameters ν_{ab} that describe the generating Hamiltonian of the corresponding unitary. We now shift our focus to the question of *using* the gate characterization data to efficiently synthesize a generic circuit. This is done in two parts, first the unitary specified by several parameters ν_{ab} generated by the non-standard pulse is *transformed by acting with only single qubit gates before and after the* pulse, into a simpler unitary defined by a set of 3 parameters c_1, c_2, c_3, which we call a new basis gate. This is done without making any changes to the pulse waveform that was characterized at the time of circuit compilation, similar to a stored gate calibration. In the second part we have the following objective: given such a discrete set of 2-qubit, how does one synthesize a desired unitary using short sequences of such gates, when it is feasible to do so. From a practical standpoint, in addition to being easier to calibrate and compile, single qubit gates typically have an order of magnitude higher fidelities and shorter durations compared to 2-qubit gates. To leverage this effectively, we use the appropriate mathematical structure that captures the map between local control parameters and non-local control parameters when considering sequences of 2-qubit unitaries interleaved with 1-qubit unitaries. The control theory of two-qubit unitaries and an analysis of local and non-local properties of 2-qubit gates was developed in [37],[38], whose essential components we briefly review below.

We introduce the following notation for a useful elementary 2-qubit unitary

$$\mathbf{Can}(c_1, c_2, c_3) = e^{-i/2(c_1 XX + c_2 YY + c_3 ZZ)} \tag{4}$$

On any 2-qubit unitary U, one can perform a Cartan or KAK decomposition which decomposes it into a non-local unitary and 4 local single-qubit unitaries $k_{L/R}, \tilde{k}_{L/R}$

$$U = \left(k_L \otimes \tilde{k}_L\right) \mathbf{Can}(c_1, c_2, c_3) \left(k_R \otimes \tilde{k}_R\right) \tag{5}$$

The notation described in (4) has been used and (c_1, c_2, c_3) are called the Cartan or canonical coordinates of U. When a subroutine is called to perform a KAK decomposition on the unitary U, it returns the decomposed form written in (5), with the set of single qubit gates $k_{L/R}, \tilde{k}_{L/R}$ being determined and a unique set of c_i that follow the ordering $\pi/2 \geq c_1 \geq c_2 \geq c_3 \geq 0$ or $\pi - c_1 \geq c_2 \geq c_3 \geq 0$.

Manipulating trade-offs between local and non-local components of 2-qubit unitaries becomes easier when one defines a notion of local equivalence. It captures the idea that a 2-qubit unitary U_1 can be transformed into a particular class of unitaries U by left or right multiplication of arbitrary 1-qubit gates. All the unitaries U it can be transformed (locally rotated) into in this way are *equivalent* to it. More formally, U is locally equivalent to U_1 if there exist single qubit gates k such that

$$U = \left(k_L \otimes \tilde{k}_L\right) U_1 \left(k_R \otimes \tilde{k}_R\right) \tag{6}$$

A local invariant is a property of a unitary that does not change when it is transformed via local rotations. This property is unchanged when U_1 is transformed into any of the different unitaries U that one gets by picking any set of k in (6). [38] provides a formal definition of a set of local invariants I_p as follows: U_1 is locally equivalent to U if and only if $I_p(U) = I_p(U_1)$ for all I_p in the set. The two local invariants I_1, I_2 (which give 3 real numbers) for 2-qubit unitaries are stated in (A2).

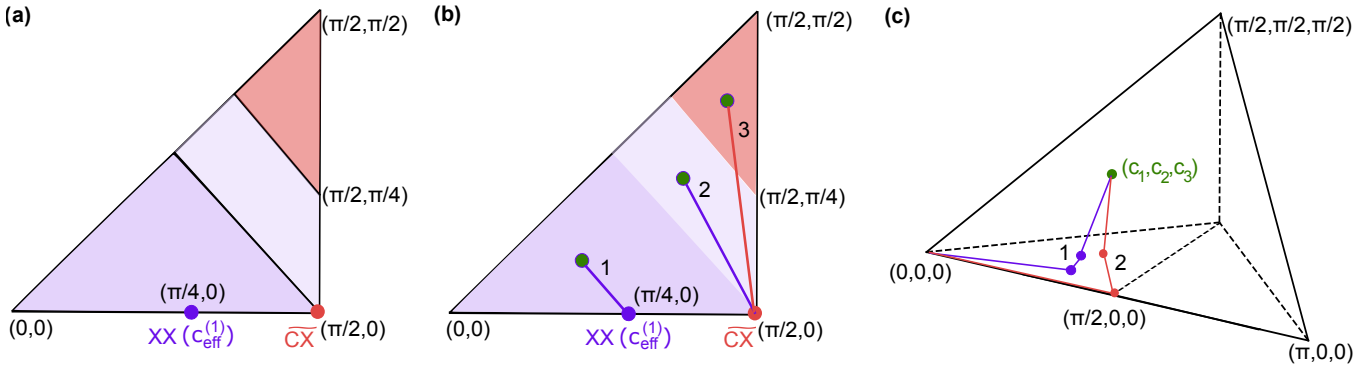

FIG. 3: **(a)** The canonical coordinates of a constructed basis gate set consisting of $XX(c_{\text{eff}}^{(1)})$ and CX gates, depicted as purple and red points respectively on the base of the Weyl tetrahedron. **(b)** The feasible region of coordinates that can be synthesized using 2 applications of $XX(c_{\text{eff}}^{(1)})$ (dark purple). The feasible region of coordinates that can be synthesized using 1 application of $XX(c_{\text{eff}}^{(1)})$ and 1 application of a CX gate (light purple). The feasible region of coordinates that can only be synthesized using 2 application of a CX gate (red). **(c)** A target unitary with coordinates (c_1, c_2, c_3) is synthesized using (1) a sequence of three $XX(c_{\text{eff}}^{(1)})$ gates, or (2) a sequence of three CX gates (2).

A. Constructing a gate definition using characterization data

Before describing the procedure to construct a new basis gate for the pulse SCR_θ using it's model, we first describe the general method to construct a useful gate definition for a generic entangling pulse from it's unitary model $e^{-i/2\ H_{\text{eff}}(\nu_{ab})}$. The utility of this method lies in being agnostic to the details of the entanglement generating mechanism used and allowing creation of fractional gates by deformation of existing control parameters (such as the pulse scaling rule in (3)). The underlying assumption is that a unitary model constructed from a set of characterization experiments, does faithfully capture how the fractional gate acts within the 2-qubit subspace. While the choice of model ansatz ie. the set of terms ν_{ab} that are allowed in the generating Hamiltonian of the unitary, are constrained by the underlying device physics, we let U be any 4-dimensional unitary matrix. The model is now parameterized by the coefficients of H_{eff} expanded in the full 2-qubit Pauli basis, with σ_0 denoting the denoting the Identity and σ_i denoting the Pauli matrices,

$$H_{\text{eff}}\left(\nu_{ab}\right) = \sum_{a,b=0}^{3} \sigma_a \otimes \sigma_b\ \nu_{ab} \tag{7}$$

There can be at most 16 parameters ν_{ab} contributing to this model, however only certain combinations of these parameters capture the non-local properties of the gate. Only those combinations of model parameters are essential, if one needs to synthesise other non-local target unitaries using repeated applications of the gate described by the model. Performing a KAK decomposition (5) on the model unitary allows the expression of these parameters in terms of the canonical coordintaes c_1, c_2, c_3, which uniquely capture the non-locality of the gate.

$$e^{-i/2H_{\text{eff}}(\nu_{ab})} = \left(k_L \otimes \tilde{k}_L\right) \mathbf{Can}(c_1, c_2, c_3) \left(k_R \otimes \tilde{k}_R\right) \tag{8}$$

The task of explicitly calculating all quantities on the right-hand side can be done numerically [39] to compute the 4 single qubit unitary matrices k and the 3 real parameters (c_1, c_2, c_3). Acting with the inverses of k on both sides above leads to the following gate definition:

$$\mathbf{Can}(c_1, c_2, c_3) := \left[k_L^{-1} \otimes \tilde{k}_L^{-1}\right] \left[e^{-i/2\tilde{H}_{\text{eff}}}\right] \left[k_R^{-1} \otimes \tilde{k}_R^{-1}\right] \tag{9}$$

This basis gate definition is to be interpreted in the following manner: the unitary matrix corresponding to $\mathbf{Can}(c_1, c_2, c_3)$ is computed in (8) using the learned model $H_{\text{eff}}\left(\nu_{ab}\right)$ of the unitary generated by a non-standard 2-qubit entangling pulse (such as a scaled pulse SCR_θ). The physical pulse waveform that generates the unitary is stored as the pulse waveform object $\left[e^{-i/2\tilde{H}_{\text{eff}}}\right]$ which is sandwiched between the single qubit gate objects $\left[k_R^{-1} \otimes \tilde{k}_R^{-1}\right]$ and $\left[k_L^{-1} \otimes \tilde{k}_L^{-1}\right]$, with the single qubit rotation angles calculated using (8). Importantly, the compiler interprets the appearance of $\mathbf{Can}(c_1, c_2, c_3)$ during different transpilation stages as follows: (i) At the stage of circuit synthesis, it is interpreted as a unitary matrix computed in (8) that is part of an available basis gate set that can be used for circuit synthesis. (ii) At the stage of pulse scheduling, it is interpreted as a composite gate object, with the replacement rule (9) being executed. Finally, circuit depth is optimized by letting all single qubit gates surrounding the pulse waveform $\left[e^{-i/2\tilde{H}_{\text{eff}}}\right]$, merge with neighboring single qubit gates, followed by resynthesis of only the merged single qubit gates.

Algorithm 2 Synthesis using efficient gates

Require: (Model, Pulse): ($H_{\text{eff}}(\nu_{ab})$, SCR_θ)
Require: Target unitary: U_{block}
1: Get c_{eff} and $k_{L/R}$,$\tilde{k}_{L/R}$ by appplying (8) to $H_{\text{eff}}(\nu_{ab})$
2: Define basis gate $XX(c_{\text{eff}})$ by unrolling SCR_θ according to (10).
3: Repeat for next pair ($H_{\text{eff}}(\nu_{ab})$, SCR_θ)
4: Apply (12) to U_{block} to get c_1, c_2, c_3.
5: Synthesize c_1, c_2, c_3 using minimum number of $XX(c_{\text{eff}}^{(j)})$ with minimum $c_{\text{eff}}^{(j)}$.
6: Consolidate and re-synthesize single qubit basis gates.

We now describe in more detail the case of experimental interest from the previous section, ie. the pulse $\text{SCR}_\theta(\nu_{ab})$. The operation (8) is performed on its model $e^{-i/2H_{\text{eff}}(\nu_{ab})}$ to calculate the set of single qubit matrices k and $(c_1, c_2, c_3) = (c_{\text{eff}}, 0, 0)$. These determined values are now used to construct a gate definition according to (9):

$$XX(c_{\text{eff}}) = \left(k_L \otimes \tilde{k}_L\right)^{-1} \text{SCR}_\theta(\nu_{ab}) \left(k_R \otimes \tilde{k}_R\right)^{-1} \tag{10}$$

The left-hand side is precisely $\mathbf{Can}(c_{\text{eff}}, 0, 0)$, but has been denoted with the more identifiable 2-qubit rotation $XX(c_{\text{eff}})$. Importantly, any ansatz that has the form (1) necessarily has $c_2 = c_3 = 0$ and c_{eff} related to its model parameters via

$$c_1 = c_{\text{eff}} = \sqrt{\nu_{zx}^2 + \nu_{zy}^2 + \nu_{zz}^2} \tag{11}$$

These relations are obtained readily by observing that $XX(c_{\text{eff}})$ is locally equivalent to the unitary model $e^{-i/2H_{\text{eff}}(\nu_{ab})}$ since both sides are only related by single qubit rotations. As a result they must have the same values of the local invariants (A2), and acting with the local invariant I_1 on both sides gives these equalities.

B. Synthesizing circuits using efficient gates

With a discrete set of newly constructed basis gates added to the gate-set, the objective is to synthesize generic circuits using them. We assume we are given as input an abstract circuit which is broken into a sequence of two-qubit unitary blocks. Each U_{block} is taken to be a sequence of uninterrupted operations on the same pair of qubits and is an element of $SU(4)$, ignoring a global phase that can be tracked separately. Typically, a circuit can comprise of products of various different U_{block} on different qubits at different circuit depths (such as the Inverse QFT algorithm) or repetitions of the same U_{block} repeated on different qubits (such as in trotterized Hamiltonian simulations).

The problem of synthesis of a generic U_{block} is first reduced to the problem of synthesizing only its non-local components (t_1, t_2, t_3), since the required single qubit gates k can be determined by applying a KAK decomposition on the block:

$$U_{\text{block}} = \left(k_L \otimes \tilde{k}_L\right) \mathbf{Can}(t_1, t_2, t_3) \left(k_R \otimes \tilde{k}_R\right) \tag{12}$$

This is an important first step because $\mathbf{Can}(t_1, t_2, t_3)$ is an easier target unitary for synthesis by virtue of being described completely by only 3 parameters.

Synthesizing $\mathbf{Can}(t_1, t_2, t_3)$ using a sequence of non-standard pulses with only models of their corresponding unitaries can be difficult, especially when each unitary is described by a large number of parameters (products of 3 such arbitrary gates can have up-to 48 parameters). Single qubit gates need to be interleaved into a template circuit, and their gate parameters optimized to make the template circuit unitarily equivalent to the target unitary. This contributes heavily to the complexity of the task.

Logical and numerical simplifications are made for the task of unitary synthesis of $\mathbf{Can}(t_1, t_2, t_3)$, by ensuring the entangling basis gate set only has canonical gates. For the efficient gates that have been constructed from model definitions, the procedure of constructing of constructing a gate definition already adds to the gate-set canonical gates, as described in (8) and (9). Additionally, existing pre-calibrated gates that are required to complete the gate-set, such as a CX gate are also added to the basis gate-set in the canonical form, ie. $\mathbf{Can}(\pi/2, 0, 0)$ using single qubit gates calculated via the same replacement rules. This forms the over-complete, discrete entangling basis gate-set with fixed parameters, that is available for any 2-qubit pair on the device. The single qubit basis gates are allowed to implement arbitrary single unitaries using 3 continuous parameters. In practice, after U_{block} has been synthesized, these are further decomposed into discrete single qubit rotations during re-synthesis of only single qubit gates, to ensure virtual Z gates are used.

To reduce the complexity of this task further, an important observation is that one only needs to to synthesize a locally equivalent circuit to $\mathbf{Can}(t_1, t_2, t_3)$. This is sufficient because any unitary A that is locally equivalent to the target unitary $\mathbf{Can}(t_1, t_2, t_3)$ can be KAK decomposed to yield the necessary single qubit gates that make it unitarily equivalent to the target. In other words, if any unitary A with the same canonical coordinates as the target is found, performing a KAK decomposition $A = (u_L \otimes \bar{u}_L)\ \mathbf{Can}(t_1, t_2, t_3)\ (u_R \otimes \bar{u}_R)$ provides 4 single single qubit gates $u_{L/R}, \tilde{u}_{L/R}$. Acting with the inverse gates $(u_L^{-1} \otimes \bar{u}_L^{-1})$ on the left of both sides and similarly with $(u_R^{-1} \otimes \bar{u}_R^{-1})$ on the right, one arrives at a circuit that is unitarily equivalent to the target.

To analyse the canonical coordinates of products of canonical basis gates such as $\mathbf{Can}(a_1, a_2, a_3)$, it is helpful to geometrically interpret each one as a 3-vector. Each single qubit gate that needs to be inserted is to be thought of as a control parameter that rotates or scales

such a vector. The problem of synthesis of the target unitary $\mathbf{Can}(t_1, t_2, t_3)$ can then be framed as determining the appropriate rotations and or scalings that need to be implemented on each 3-vector using single qubit gates, such that the product of the sequence of unitaries finally has the target coordinates (t_1, t_2, t_3).

The important base case to consider, is using 2 entangling gates to synthesize a generic target unitary. Once such a the sum rule or motif has been determined, it can be repeatedly used if needed. This can be stated as finding the set of single qubit gates $s, \tilde{s}$ inserted between two basis gates such that the interleaved product is locally equivalent to the target:

$$\mathbf{Can}(t_1, t_2, t_3) \equiv \mathbf{Can}(a_1, a_2, a_3)(s \otimes \tilde{s})\mathbf{Can}(b_1, b_2, b_3) \tag{13}$$

We have used $\equiv$ to emphasize that both sides only need to be locally equivalent and not unitarily equivalent, and we use this notation where only this weaker equivalence is needed.

The goal is to express the set of locally invariant quantities $\{I_k\}$ of a sequence of basis gates that is to be used for synthesis, parametrically in terms of the rotation angles of the interleaved single qubit gates. This parametrized set $\{I_k(s, \tilde{s})\}$ must be equal to the set evaluated solely on the target coordinates (t_1, t_2, t_3), which gives a set of equalities. Any single qubit gate angles that satisfy the complete set of equalities, are valid solutions and can be used for synthesis. The target coordinates for which it is possible to find any (real) solutions for the single qubit gate angles, are referred to as feasible coordinates. These are the set of target coordinates that can be synthesized using a chosen sequence or motif of entangling gates.

To use the set of local invariants parametrically, one needs an efficient construction of these invariants for sequences of canonical gates with interleaved local gates. The magic matrix Q defined in (A1) is useful for constructing these invariant quantities as it maps local unitary transformations to real orthogonal transformations. The generators of local transformations acting on the 2-qubit subspace are the generators of a pair of single qubit rotations generated by the Pauli matrices, ie. $i\sigma_i/2 \otimes I$ and $I \otimes i\sigma_i/2$ for $i \in \{1, 2, 3\}$. These are mapped to the generators $Q^\dagger(i\sigma_i/2 \otimes I)\ Q = L_i$ and $Q^\dagger(I \otimes i\sigma_i/2)\ Q = \tilde{L}_i$ which satisfy the commutation relations $[L_i, L_j] = \epsilon_{ijk}\, L_k$, $[\tilde{L}_i, \tilde{L}_j] = \epsilon_{ijk}\, \tilde{L}_k$ and $[L_i, \tilde{L}_j] = 0$. The 6 generators $L_i, \tilde{L}_i$ are purely real with purely real structure constants (ϵ_{ijk}) and generate 4 dimensional real orthogonal transformations. The underlying mathematical fact that has been important in studying local invariants [37] that is being used here is that there exists a decomposition of the 2-qubit $su(4)$ algebra into $so(4) + i\ p$, where p is real and symmetric while $so(4)$ is real and antisymmetric.

The second transformation which simplifies the task maps all 6 generators $L_i, \tilde{L}_i$ in the 4 dimensional defining representation to the adjoint (or regular) representation of this algebra. This 6-dimensional representation has as basis vectors the generators $L_i, \tilde{L}_i$ themselves and is equipped with the Cartan-Killing inner product $\langle L_i L_j \rangle = -Tr(L_i L_j) = \delta_{ij}$ and $\langle \tilde{L}_i \tilde{L}_j \rangle = -Tr(\tilde{L}_i \tilde{L}_j) = \delta_{ij}$.

The adjoint representation is useful because local rotations are realized in the form of a 6 dimensional block-diagonal matrix $S = \exp\Big(\gamma\ \mathbf{n} \cdot \mathbf{L} + \tilde{\gamma}\ \tilde{\mathbf{n}} \cdot \tilde{\mathbf{L}}\Big)$. Here we denote the finite rotation angle as γ, a unit normal vector $\mathbf{n}$, and the dot product denotes $\mathbf{n} \cdot \mathbf{L} = \sum_{i=1}^{3}\ L_i\ n_i$. The same conventions are used for the rotation parameters $\tilde{\mathbf{n}}$ and $\tilde{\gamma}$ corresponding to $\tilde{L}_i$. The block diagonal rotation matrix in terms of these rotation parameters is

$$S = \begin{bmatrix} \exp\left(\gamma\, \mathbf{n} \cdot \mathbf{L}\right) & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \exp\left(\tilde{\gamma}\, \tilde{\mathbf{n}} \cdot \tilde{\mathbf{L}}\right) \end{bmatrix} \tag{14}$$

To express the canonical gates these matrices transform in the adjoint representation, we first look at the example of the canonical gate $\mathbf{Can}(t_1, t_2, t_3)$ defined in (4). In the basis spanned by $L_i, \tilde{L}_i$, the exponent can be written in terms of a purely real and symmetric matrix $T = \sum_{i=1}^{3}\ t_i \left(L_i \tilde{L}_i + \tilde{L}_i L_i\right)$. Using following notation for 3 by 3 diagonal matrices,

$$\begin{aligned} \left[\mathbf{cos}(\,\vec{t}\,)\right]_{3\times 3} &:= \mathrm{diag}(\cos t_1, \cos t_2, \cos t_3) \\ \left[\mathbf{sin}(\,\vec{t}\,)\right]_{3\times 3} &:= \mathrm{diag}(\sin t_1, \sin t_2, \sin t_3) \end{aligned}$$

the result of the matrix exponentiation of T by a resummation of the series expansion results in the block diagonal form

$$\exp(-i\ T) = \begin{bmatrix} \left[\mathbf{cos}(\,\vec{t}\,)\right] & \left[i\,\mathbf{sin}(\,\vec{t}\,)\right] \\ \left[i\,\mathbf{sin}(\,\vec{t}\,)\right] & \left[\mathbf{cos}(\,\vec{t}\,)\right] \end{bmatrix} \tag{15}$$

To identify how local invariants are embedded in this symmetric matrix, we observe that the determinants of the blocks are

$$\begin{aligned} I_e &= \mathrm{Det}\left[\mathbf{cos}(\,\vec{t}\,)\right] = \cos t_1 \cos t_2 \cos t_3 \\ I_o &= \mathrm{Det}\left[i\,\mathbf{sin}(\,\vec{t}\,)\right] = -i \sin t_1 \sin t_2 \sin t_3 \\ I_a &= \mathrm{Det}\left(\left[\mathbf{cos}(\,\vec{t}\,)\right]^T \left[\mathbf{cos}(\,\vec{t}\,)\right] + \left[i\,\mathbf{sin}(\,\vec{t}\,)\right]^T \left[i\,\mathbf{sin}(\,\vec{t}\,)\right]\right) \end{aligned} \tag{16}$$

I_e, I_o match the real and imaginary parts of the local invariant I_1, stated in Appendix (A3). Moreover, I_a is related to the second invariant $I_2 = 4\left(I_e^2 + I_o^2\right) - I_a$. One can check that an arbitrary pair of single-qubit rotations, represented block diagonally in (14), acts on canonical gates represented block diagonally in (15) via both left and right multiplication (which correspond to local gates before and after the canonical gate) keeping I_e, I_o, I_a invariant. This is because local rotations act on each block as special orthogonal rotations, for instance M_1, M_2, and a transformed block $M_1 \,.\, \left[\mathbf{cos}(\,\vec{t}\,)\right] \,.M_2$ has a determinant multiplied by $\mathrm{Det}\,(M_1)\ \mathrm{Det}\,(M_2) = 1$. This is a statement that in the adjoint representation, canonical gates are represented as transformations with a notion

of length captured by the even I_e and odd I_o invariants. When local rotations act from both the left and right, the canonical gate blocks are also rotated and summed, in which case the preservation of angles is captured by I_o.

With the local invariants I_e, I_o, I_a identified in terms of matrix determinants in this representation, we proceed to write all gates in this representation. Using the block diagonal representation (15) for the target unitary $\mathbf{Can}(t_1, t_2, t_3) = e^{-iT}$ and basis gates $\mathbf{Can}(a_1, a_2, a_3) = e^{-iA}$, $\mathbf{Can}(b_1, b_2, b_3) = e^{-iB}$, the block diagonal rotation matrix (14) for the pair of interleaved single-qubit rotations S, the synthesis ansatz (13) is now written as

$$e^{-iT} \equiv e^{-iA}\ S(\gamma, \mathbf{n}, \tilde{\gamma}, \tilde{\mathbf{n}})\ e^{-iB} \tag{17}$$

The canonical coordinates of T can be determined by equating the values taken by the three invariants I_e, I_o, I_a acting on both sides. They can also be determined directly using an inverse map from the invariants to the canonical coordinates. One such inverse map is provided in [40] as a cubic equation whose coefficients are determined by a set of invariants and whose roots are the canonical coordinates corresponding to the set of invariants.

$$p(z) = z^3 - g_3 z^2 + \left(4\sqrt{g_1^2 + g_2^2} - 1\right) z + g_3 - 4g_1 \tag{18}$$

$$\text{for } g_1 = I_e^2 + I_o^2,\ g_2 = 2iI_eI_o,\ g_3 = 4(I_e^2 + I_o^2) - I_a$$

The roots are ordered $z_1 \le z_2 \le z_3$ and determine the canonical coordinates the target unitary: $t_1 = \frac{1}{2}\cos^{-1} z_1$, $t_2 = \frac{1}{2}\cos^{-1} z_2$, $t_3 = \frac{1}{2}\cos^{-1} z_3$ or $t_1 = \pi - \frac{1}{2}\cos^{-1} z_1$ if $g_2 < 0$. This is a set of 3 simultaneous equations with the location of roots z_1, z_2, z_3 parameterized by the invariants I_e, I_o, I_a which are functions of the single-qubit parameters $\gamma, \mathbf{n}, \tilde{\gamma}, \tilde{\mathbf{n}}$. Any single qubit parameters that simultaneously satisfy these all 3 constraints provide a valid set of single-qubit gates necessary for synthesis.

For an analytic solution, we look at special cases where the rotation matrix S described in (14) has fixed normal vectors $\mathbf{n} = \tilde{\mathbf{n}} = [0, 0, 1]$, ie orthogonal rotations are only generated by $L_3, \tilde{L}_3$ (Z rotations). To get a pair of decoupled solutions, we equate the linear combination of the invariants $I_e + I_o, I_e - I_o$ acting on both sides of (17) to get

$$\begin{aligned}\cos\left(t^{(\pm)}\right) = \cos\left(a^{(\pm)}\right)\cos\left(b^{(\pm)}\right) - \\ \sin\left(a^{(\pm)}\right)\sin\left(b^{(\pm)}\right)\cos\left(\gamma^{(\mp)}\right)\end{aligned} \tag{19}$$

where $\gamma^{(\mp)} = \gamma_1 \mp \gamma_2$, $t^{(\pm)} = t_1 \pm t_2$ with the same sum and difference in coordinates for $a^{(\pm)}, b^{(\pm)}$.

The third coordinate is constrained to be $t_3 = a_3 + b_3$ since the single-qubit rotations are restricted to be only Z rotations. The constraint equations are decoupled in the variables $t^{(\pm)}, a^{(\pm)}, b^{(\pm)}$, which has the crucial implication that *one can independently control $\gamma^{\mp}$ to synthesize a set of target coordinates t_1, t_2* in (17).

The constraint equations in these variables are reduced to two copies of a single-qubit synthesis problem, each of which can be solved separately for $\gamma^{\mp}$. Solutions are $\{\gamma^{\mp}, -\gamma^{\mp}\} + 2n\pi$ for integer values of n where

$$\gamma^{\mp} = \arccos\left(\cot\left(a^{\pm}\right)\cot\left(b^{\pm}\right) - \cos\left(t^{\pm}\right)\csc\left(a^{\pm}\right)\csc\left(b^{\pm}\right)\right) \tag{20}$$

Existence of real $\gamma^{\mp}$ implies the existence of a feasible set of coordinates $t^{(\pm)}$, which allows one to bound

$$|a^{(\pm)} - b^{(\pm)}| \le t^{(\pm)} \le \pi - |\pi - |a^{(\pm)} + b^{(\pm)}| \tag{21}$$

Subroutine 1 (Synthesis with 2 basis gates)**.** The synthesis of a target unitary $\mathbf{Can}(t_1, t_2, t_3)$ from the available basis gates $\mathbf{Can}(a_1, a_2, a_3)$ and $\mathbf{Can}(b_1, b_2, b_3)$ proceeds by expressing the ansatz (13) in a block diagonal representation (17). Local equivalence of the composed sequence with the the target is imposed by calculating local invariants as determinants of each block, which gives a set of equalities. given byof the t By fixing the local rotation axes as $\mathbf{n} = \tilde{\mathbf{n}} = [0, 0, 1]$, the problem reduces to . This leads to a decoupled set of constraint equations, Eq. (19), whose solutions for the required single-qubit rotation angles are given in closed form by Eq. (20). These solutions exist whenever the feasibility condition in Eq. (21) is satisfied.

We summarize the resulting subroutine as follows. The synthesis of a target unitary $\mathbf{Can}(t_1, t_2, t_3)$ from two basis gates, $\mathbf{Can}(a_1, a_2, a_3)$ and $\mathbf{Can}(b_1, b_2, b_3)$, is implemented via the gate sequence of Eq. (13), expressed in the adjoint representation as Eq. (17). For the choice of normal vectors $\mathbf{n} = \tilde{\mathbf{n}} = [0, 0, 1]$, the required single-qubit rotation angles are obtained from a decoupled set of constraints given in Eq. (19), with solutions in Eq. (20). *This construction is valid when the feasibility condition in Eq. (21) is satisfied.* We refer to this procedure as the *synthesis subroutine*, which will be applied in subsequent constructions.

Solutions to the problem of synthesis using a sequence of 2 basis gates (13) can be used for longer sequences. For generic canonical gates one can use(18) with $a_3 = b_3 = 0$ analytically controls of rotations in the $t_1 - t_2$ plane using and is useful for canonical gates same set of steps that allow rotations in the $t_1 - t_2$ plane by a choice of $\mathbf{n} = \tilde{\mathbf{n}} = [0, 0, 1]$ can be formulated in $t_2 - t_3$ or $t_3 - t_1$ planes by picking unit normal vectors $\mathbf{n} = \tilde{\mathbf{n}}$ perpendicular to these planes, giving a similar set of parameterized solution as (20). To summarize the complete subroutine that allows the use of a sequence of 2 fractionally entangling basis gates (13), one first finds the smallest basis gates that satisfy the criterion (21). Then the single qubit gates are determined via(20). With another sequential rotation performed in the $t_2 - t_3$ plane or $t_3 - t_1$ plane, the points within the interior of the tetrahedron can be reached.

We now look at specific instances of this general method that are applied to the single parameter canonical gates that were experimentally constructed and

added to the basis gate set. We remind the reader that the single-parameter 2-qubit rotations $XX(c_{\text{eff}}^{(1)}) = \mathbf{Can}(c_{\text{eff}}^{(1)}, 0, 0)$ and $XX(c_{\text{eff}}^{(2)}) = \mathbf{Can}(c_{\text{eff}}^{(2)}, 0, 0)$. With the goal of maximizing the range of canonical coordinates of a target unitary $\mathbf{Can}(c_1, c_2, 0)$, conjugation by a single qubit gate implements a permutation on the second basis gate making the ansatz $\mathbf{Can}(c_{\text{eff}}^{(1)}, 0, 0)$ $(s \otimes \tilde{s})$ $\mathbf{Can}(0, c_{\text{eff}}^{(2)}, 0)$. The 6 single qubit gates that implement permutations and permutations with sign-flips on two coordinates are called Weyl reflections and are described in Appendix B. The ansatz has the following set of feasible solutions

$$c_{\text{eff}}^{(1)} + c_{\text{eff}}^{(2)} \geq c_1 + c_2, \quad c_{\text{eff}}^{(1)} - c_{\text{eff}}^{(2)} \leq c_1 - c_2. \tag{22}$$

The explicit set of solutions for the single qubit angles is given in Appendix C in the form of the angles of the single qubit gates. For the case of a sequence of the three single parameter canonical gates with effective rotations $c_{\text{eff}}^{(i)}$ used to synthesize a target unitary block with canonical coordinates (c_1, c_2, c_3), the feasibility condition that needs to be satisfied is:

$$c_{\text{eff}}^{(1)} + c_{\text{eff}}^{(2)} \geq c_1 + c_2, \quad c_{\text{eff}}^{(1)} - c_{\text{eff}}^{(2)} \leq c_1 - c_2, \quad c_{\text{eff}}^{(3)} \geq c_3. \tag{23}$$

The explicit set of angle parameterizing s_1, s_2 is given in Appendix C. We note that if the gateset contains a CX gate (which is locally equivalent to $\mathbf{Can}(\pi/2, 0, 0)$), these conditions can always be satisfied via $c_{\text{eff}}^{(1)} = c_{\text{eff}}^{(2)} = c_{\text{eff}}^{(3)} = \pi/2$ which is a reduction to the known result that any 2-qubit unitary can be synthesized with at-most 3 CX gates.

We now briefly compare our approach to synthesis with those in the existing literature that also make use of KAK decompositions. If one has a continuous single parameter entangling gate set of the form $XX(\theta)$ (equivalently $ZX(\theta)$ or $ZZ(\theta)$), one can apply it sequentially 3 times with each instance being tuned to c_1, c_2, c_3, respectively. This leads to a simple one-to-one replacement of the canonical coordinates (c_1, c_2, c_3) of the U_{block} to be synthesized and is done in [26], [41]. In contrast, our algorithm applies a sequence of entangling gates with fixed fractional angles whose canonical coordinates are different from the coordinates to be synthesized. Using a discrete set of single parameter fractionally entangling gates for unitary synthesis is discussed in [27], our approach has a focus on developing a set of tools that enable synthesis using multi-parameter fractionally entangling gates and employ a different set of mathematical techniques i.e. the use of local invariants as opposed to monodromy methods.

IV. EXPERIMENTAL ALGORITHMIC BENCHMARKING

We now demonstrate the advantage of using efficient gates in the execution of benchmarking quantum algorithms that are run on the 127-qubit processor IBM Brisbane. We analyze the performance of the one-hot inverse Quantum Fourier Transform algorithm and a Trotterized simulation of the transverse field Ising model over a range of circuit widths. We ensure that the test circuits differ only by substitution of entangling gates. The test circuit is executed on the same qubit layout, with the same number of shots, and with identical run-time settings compared to the control version that uses only default gates. The transpilation process differs only in the final stage; in one case, a re-synthesis of 2-qubit unitary blocks is done with the option of using efficient gates in addition to default backend gates. Identical measurement-error mitigation is applied to the raw counts for all circuits. We refer the reader to Algorithm 1 for a summary of the characterization routine and Algorithm 2 for a summary of the gate construction routine that was specifically implemented in these experiments.

A. Quantum Fourier Transform

The inverse Quantum Fourier Transform is part of standard algorithmic benchmarking suites [42] and is an important subroutine of many important quantum algorithms such as Quantum Phase Estimation. By including an initial layer of single-qubit rotations, the algorithm can be made "one-hot"; that is, its ideal response is a single target bitstring with 100% probability. The metric for evaluation that we use is the success probability of obtaining the target bit string.

In Figure 4, we observe an increasing advantage in using shorter gates as we go to higher circuit depths, while also suppressing coherent errors through our gate characterization. This is an end-to-end execution with an additional entangling gate $XX\,(\pi/4 + \epsilon_i)$ available on 85% of the qubit pairs on the device, in addition to the default CX gate. The 2 qubit rotation angle for the gate has the subtle feature of being different across different qubit pairs, and is denoted by $\pi/4 + \epsilon_i$, with ϵ_i denoting the small correction to the rotation angle that is unique to each qubit pair, and is determined from the characterization experiments. The synthesis motifs that are useful for these algorithms are depicted via the set of feasible regions depicted in Fig 3. The structure of the QFT algorithm predominantly has increasingly smaller 2-qubit rotations, which can be synthesized using the usage of 2 $XX\,(\pi/4 + \epsilon_i)$ gates. Moreover, a common circuit motif involves a swap-gate followed by such a 2-qubit controlled rotation with angle θ, in which case these are consolidated to provide a target unitary block with coordinates $(\pi/2, \pi/2, \theta)$. In such cases as well, one can use a combination of 2 CX gates and 1 efficient gate for synthesis of the target block.

We observe an improvement in algorithmic success probability of up to 3 X for up to $N = 26$ qubits, with the improvement increasing with increasing circuit width. Remarkably, the most likely bitstring no longer

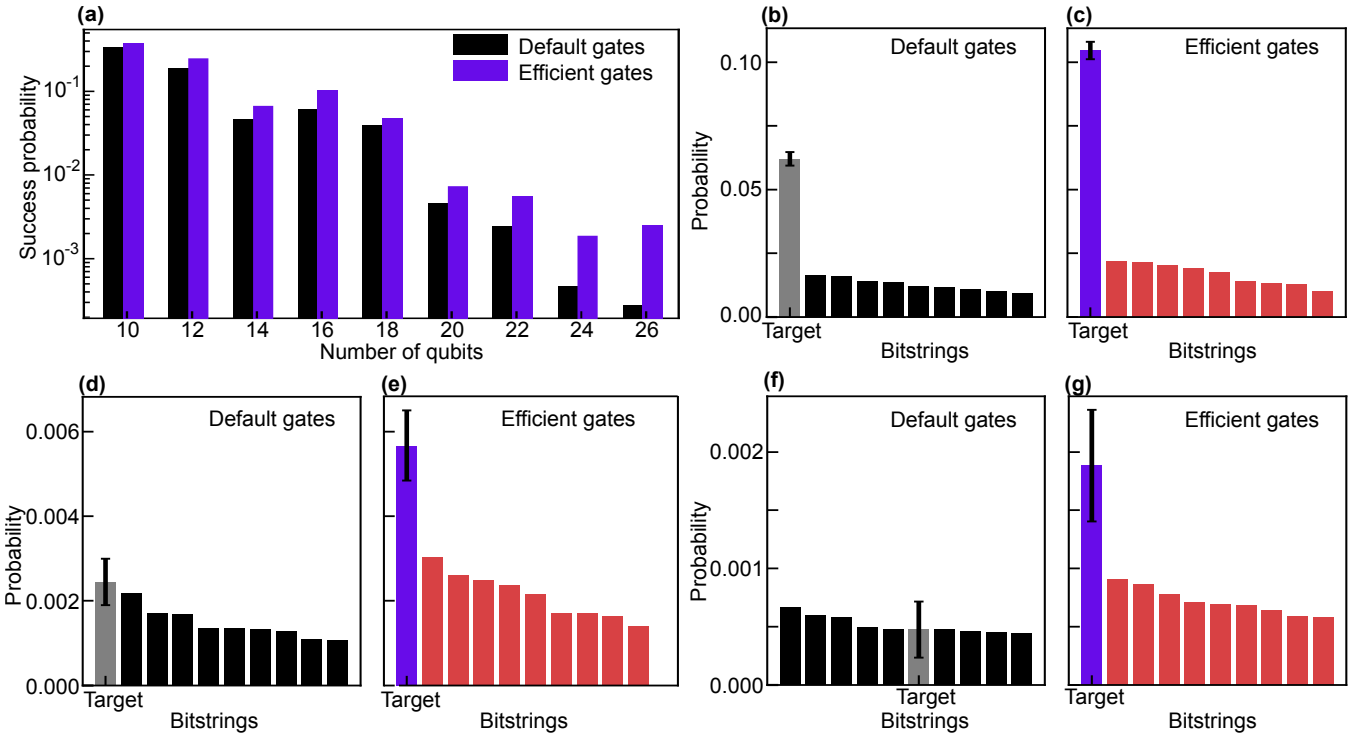

FIG. 4: Experimental comparison of inverse quantum Fourier transform circuits run on ibm brisbane, with identical execution workflows except for the usage of efficient gates in addition to backend default gates versus usage of only backend default gates. The target bitstring is $101010\ldots10$. **a** Observed likelihood of obtaining the target bitstring as a function of circuit width. **b-d** Histograms of observed results for three circuit widths $N = 16, 22, 24$ respectively. Only the 10 most frequently observed bitstrings are depicted. In all cases, the success probability is substantially boosted by using efficient gates. For the 24-qubit QFT, the use of efficient gates allows the correct result to emerge above the noise as the most likely result. Error bars on the target bitstring represent shot noise.

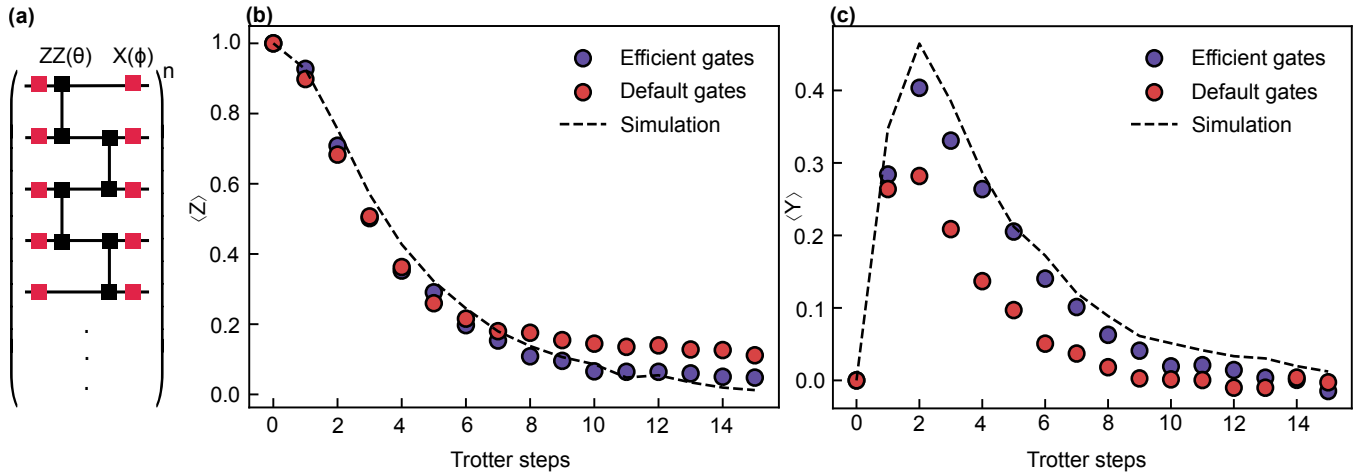

FIG. 5: Experimental comparison of Hamiltonian simulation of the TFIM with default versus efficient gates. The implementation uses $N = 25$ qubits connected on a one dimensional line. **a.** A single Trotter layer. The $\mathrm{ZZ}(\theta)$ gates are synthesized using 2 efficient gates $XX(\pi/4)$ in contrast with 2 ECR gates in the default case. **b.** Observed expectation value of the average Z magnetization. **c.** Observed expectation value of the average Y magnetization. The resulting trajectories using efficient gates are significantly closer to the ideal simulation of the Trotterized evolution compared to the circuits that use only default gates with a 9 times smaller MSE for $\langle Y \rangle$ and a 5 times smaller MSE for $\langle Z \rangle$).

co-coincides with the target bitstring for the $N = 24$ problem instance when only default gates are used, but

is not selected as the target when the new basis gates are used. Since the structure of the QFT algorithm predominantly has increasingly smaller 2-qubit rotations, this benchmark demonstrates the quality of characterization procedure to be sufficient in constructing high-fidelity $XX\,(\theta)$ gates that are useable in algorithmic contexts.

B. Hamiltonian Simulation

The task of simulating the quantum dynamics of Hamiltonians of interest is a promising near-term application with the potential to showcase the utility of quantum computers [43]. Simulating Hamiltonians using Trotter-Suzuki product formulas naturally leads us toward the need of being able to have a calibrated entangling gate set of shorter duration without sacrificing fidelity.

We simulate the 1-dim quantum transverse field Ising model with Hamiltonian

$$H = -J \sum_i Z_i Z_{i+1} + h \sum_i X_i \tag{24}$$

Here J is the coupling between nearest neighboring spins and h is the global transverse magnetic field strength. We use the second-order Trotter Suzuki product formulas to simulate the evolution for the full time duration $t = n\,dt$ by breaking it into n trotter steps. Fig. 5 (a) shows a circuit implementation of each trotter step.

We implement the transverse-field Ising model Hamiltonian with $N = 25$ qubits connected on a one-dimensional line topology, with parameters $J = h = 1$. Each circuit represents an evolution for time $t = n\,dt$, where $dt = \pi/15 \sim 0.209$ and n is the number of Trotter steps. The compiled circuits require $48n$ default ECR gates (two per ZZ(θ) operation), which can be replaced by efficient gates.

In Fig. 5, we plot the observed expectation value of the average Y and Z magnetization. The observed expectation values using efficient gates are significantly closer to the ideal simulation of the trotterized evolution compared to circuits that use only default gates with a 9 times smaller MSE for $\langle Y \rangle$ and a 5 times smaller MSE for $\langle Z \rangle$).

V. SUMMARY AND DISCUSSION

In this work, we have presented an automated and resource-efficient technique to diversify the entangling gate set of a quantum device by adding an additional entangling gate for each qubit pair across the entire device. The added 2-qubit gates are less entangling compared to the maximally entangling CX gate, which is already present in the gate set. This is done while ensuring the new gates have shorter durations and maintain high fidelities. By using unitary synthesis schemes that efficiently leverage these gates, the compiled circuits are shorter in duration, accumulate less coherent error, and are executed with higher fidelity. This allows for a more efficient utilization of the finite-qubit coherence time available on the device.

Calibration and system identification in superconducting devices have challenging problems, an important one being device parameter drift over time [44]. A crucial constraint on the calibration time is that it should be small compared to the time scale over which the system drifts in addition to the time it takes to perform the experiments. This becomes more important as the scale of these devices becomes larger.

Our approach tackles this, by making the procedure of adding a new entangling gate take up a small amount of device time by performing highly time-efficient offline gate Hamiltonian characterization, rather than online closed-loop gate calibration. Instead of calibrating a control pulse to reach a target unitary, we focus on accurately and quickly characterizing the control pulse. The result is a gate definition, which is a map between the control pulse that is executed and a model of the generating Hamiltonian it implements. As long as the gate definition is used in experiments within the parameter-drift timescale, this map is a faithful one and the Hamiltonian accurately describes the dynamics generated by the gate in the 2-qubit subspace it acts on.

An effective Hamiltonian analysis of the microscopic model of the device plays an important role in providing the allowed set of terms that can appear in the model of the generating Hamiltonian of the unitary, emphasizing a hierarchy in the strengths of these terms [35]. We take this form of the effective Hamiltonian as an ansatz, promoting the strengths of all terms that are present to learnable parameters with a bounded range of expected strengths. The actual generating Hamiltonian is then precisely determined using a set of tailored error-amplification characterization sequences.

Our approach offers a different perspective, compared to traditional methods in gate calibration and design [45]. A new 2-qubit basis gate is added to the gate-set whose unitary is a priori only approximately known, based on how the backend default gate pulse was deformed or scaled. QPU time is used judiciously by performing a precise characterization routine, instead of tuning the pulse parameters to achieve a response that precisely realizes a target unitary operation. Importantly, to make such a non-standard gate definition (the unitary generated by $H_{\text{eff}}\,(\nu_{ab})$) usable by the compiler, the pulse is sandwiched with single qubit gates that perform local rotations to create a new composite gate unitary $\mathbf{Can}\left(c_{\text{eff}}^{(1)}, c_{\text{eff}}^{(2)}, c_{\text{eff}}^{(3)}\right)$ where the local rotations accomplish the task of converting the non-standard unitary described by the set of parameters ν_{ab} that describe the model of the effective Hamiltonian into 2-qubit rotations described by atmost 3 parameters. This composite gate can now be considered a basis gate in the sense that our compiler can use it for circuit synthesis, in addition to a default CX gate, with the option to repeat the procedure with a new

pulse to add a discrete set of partially entangling basis gates to the gate set. For re-synthesis of generic circuits in terms of these composite gates, we decompose unitary blocks that appear in the circuit and use a synthesis technique that allows unitary synthesis of each block using a discrete set of basis gates $\mathbf{Can}\left(c_{\text{eff}}^{(1)}, c_{\text{eff}}^{(2)}, c_{\text{eff}}^{(3)}\right)$. These synthesis motifs efficiently use the 2-qubit rotation generated by the nonstandard pulse generated by $H_{\text{eff}}(\nu_{ab})$, to return final compiled circuits with shorter gate sequences and suppressed coherent errors, such as crosstalk.

We experimentally demonstrate our procedure by characterizing the effective Hamiltonian of direct scaled cross-resonance pulses that are constructed by deforming the echoed cross-resonance pulses that generate the backend provided CX gate. The routine to characterize one additional gate on all coupled qubit pairs on the 127-qubit device *ibm_brisbane* is completed in under 3 min of QPU time, resulting a new basis gate $\mathbf{Can}(c_{\text{eff}}, 0, 0) = XX(\pi/4)$ added to 85% of the 144 coupled qubit pairs. The classical computational overhead of this technique is kept low as model optimizations across different 2-qubit pairs are done in parallel. To demonstrate that the presented unitary synthesis schemes can take advantage of the enriched gate set to efficiently synthesize circuits, we benchmark a standard algorithm, the hot inverse quantum Fourier transform (QFT) [42]. The benchmarking protocol only distinguishes between 2 sets of circuits in whether a new basis gate is available in addition to a default CX gate, keeping all other elements of circuit execution the same. The procedure reveals up to a 3X improvement in the algorithmic success probability for up to $N = 26$ qubits and demonstrates that the $XX(\pi/4)$ gates are sufficiently high quality to improve fidelity in algorithmic contexts.

Our second experimental algorithmic benchmark is an implementation of the trotterized Hamiltonian simulation applied to the one-dimensional transverse field ising model on a line [46]. The benchmarking is done for a 25 qubit long chain with a trotter depth of upto 15 steps. At 15 trotter steps, the simulation uses up to 600 basis gates $XX(\pi/4)$ and the time dynamics of averaged observables is compared to an ideal classical simulation. We observe an improvement of upto 9 times in the MSE in the calculation of expectation values of observables, with the dynamics closely resembling the classical simulation.

Hamiltonian simulation algorithms are an important class of near term algorithms with the potential to get large performance benefits from high fidelity 2-qubit gates with natively smaller rotation angles. For trotterized Hamilonian simulations to compete with existing classical approaches, in addition to a large number of qubits, it is necessary to perform simulations for increasingly longer times while keeping a sufficiently high accuracy. This requires being able to perform these simulations not only for longer circuit depth, but also with lower trotterization error. This implies that the Hamiltonian generating the time evolution across each time slice needs to be implemented with smaller coefficients and the availability of an enriched gate set that can implement high fidelity small 2-qubit rotations $XX(c_{\text{eff}})$ in such scenarios, is very useful. It allows for an increase in the the algorithmic depth to which these algorithms can be implemented for the same qubit coherence time.

While there are cases where violation of the generating Hamiltonian ansatz for 2-qubit unitaries, leads to non-convergence in model learning, we find that it converges for the majority of 2-qubit pairs on the devices we considered. Furthoermore, the problem of device parameter drift with time puts a limit on the duration of time for which the gate definition our characterization produces is a valid or useful one. Due to the constraints of accessing commercial devices on the cloud, we were unable to systematically study the temporal dependence of our characterization and leave this study to future work.

This work underlines useful techniques that enable the execution of such near term quantum algorithms with high fidelity. Importantly, it showcases that significant performance gains can be obtained and limitations imposed by current quantum computers can be lifted, by developing solutions purely in software. The co-design of pulse level gates and compiler architecture is a striking example of research in this direction, with significant potential for innovation.

VI. ACKNOWLEDGEMENTS

[1] A. Y. Kitaev, Russian Mathematical Surveys **52**, 1191 (1997).
[2] D. P. DiVincenzo, Phys. Rev. A **51**, 1015 (1995).
[3] S. Lloyd, Physical review letters **75 2**, 346 (1995).
[4] J. Majer *et al.*, Nature **449**, 443 (2007).
[5] L. DiCarlo *et al.*, Nature **460**, 240 (2009).
[6] C. Rigetti and M. Devoret, Phys. Rev. B **81**, 134507 (2010).
[7] J. M. Chow, J. M. Gambetta, A. W. Cross, S. T. Merkel, C. Rigetti, and M. Steffen, New Journal of Physics **15**, 115012 (2013).
[8] S. Sheldon, E. Magesan, J. M. Chow, and J. M. Gambetta, Physical Review A **93**, 10.1103/physreva.93.060302 (2016).
[9] L. Casparis, N. J. Pearson, A. Kringhøj, T. W. Larsen, F. Kuemmeth, J. Nygård, P. Krogstrup, K. D. Petersson, and C. M. Marcus, Physical Review B **99**, 10.1103/physrevb.99.085434 (2019).
[10] K. Wei, E. Magesan, I. Lauer, S. Srinivasan, D. Bogorin, S. Carnevale, G. Keefe, Y. Kim, D. Klaus, W. Landers, N. Sundaresan, C. Wang, E. Zhang, M. Steffen, O. Dial, D. McKay, and A. Kandala, Physical Review Letters **129**,

10.1103/physrevlett.129.060501 (2022).
[11] D. M. Abrams, N. Didier, B. R. Johnson, M. P. da Silva, and C. A. Ryan, Nature Electronics **3**, 744 (2020).
[12] B. Foxen, C. Neill, A. Dunsworth, P. Roushan, B. Chiaro, A. Megrant, J. Kelly, Z. Chen, K. Satzinger, R. Barends, F. Arute, K. Arya, R. Babbush, D. Bacon, J. Bardin, S. Boixo, D. Buell, B. Burkett, Y. Chen, R. Collins, E. Farhi, A. Fowler, C. Gidney, M. Giustina, R. Graff, M. Harrigan, T. Huang, S. Isakov, E. Jeffrey, Z. Jiang, D. Kafri, K. Kechedzhi, P. Klimov, A. Korotkov, F. Kostritsa, D. Landhuis, E. Lucero, J. McClean, M. McEwen, X. Mi, M. Mohseni, J. Mutus, O. Naaman, M. Neeley, M. Niu, A. Petukhov, C. Quintana, N. Rubin, D. Sank, V. Smelyanskiy, A. Vainsencher, T. White, Z. Yao, P. Yeh, A. Zalcman, H. Neven, and J. Martinis, Physical Review Letters **125**, 10.1103/physrevlett.125.120504 (2020).
[13] A. D. Hill, M. J. Hodson, N. Didier, and M. J. Reagor, Realization of arbitrary doubly-controlled quantum phase gates (2021), arXiv:2108.01652 [quant-ph].
[14] M. C. Collodo, J. Herrmann, N. Lacroix, C. K. Andersen, A. Remm, S. Lazar, J.-C. Besse, T. Walter, A. Wallraff, and C. Eichler, Phys. Rev. Lett. **125**, 240502 (2020).
[15] Y. Baum, M. Amico, S. Howell, M. Hush, M. Liuzzi, P. Mundada, T. Merkh, A. R. Carvalho, and M. J. Biercuk, PRX Quantum **2**, 040324 (2021).
[16] H. Ball, M. J. Biercuk, A. Carvalho, J. Chen, M. Hush, L. A. D. Castro, L. Li, P. J. Liebermann, H. J. Slatyer, C. Edmunds, V. Frey, C. Hempel, and A. Milne, Software tools for quantum control: Improving quantum computer performance through noise and error suppression (2020), arXiv:2001.04060 [quant-ph].
[17] S. Machnes, E. Assémat, D. Tannor, and F. K. Wilhelm, Phys. Rev. Lett. **120**, 150401 (2018).
[18] P. V. Klimov, J. Kelly, J. M. Martinis, and H. Neven, arXiv:2006.04594 (2020).
[19] N. Khaneja, T. Reiss, C. Kehlet, T. Schulte-Herbrüggen, and S. J. Glaser, J. Magn. Resonance **172**, 296 (2005).
[20] J. Kelly, P. O'Malley, M. Neeley, H. Neven, and J. M. Martinis, Physical qubit calibration on a directed acyclic graph (2018), arXiv:1803.03226 [quant-ph].
[21] J. Kelly, R. Barends, A. G. Fowler, A. Megrant, E. Jeffrey, T. C. White, D. Sank, J. Y. Mutus, B. Campbell, Y. Chen, Z. Chen, B. Chiaro, A. Dunsworth, E. Lucero, M. Neeley, C. Neill, P. J. J. O'Malley, C. Quintana, P. Roushan, A. Vainsencher, J. Wenner, and J. M. Martinis, Physical Review A **94**, 10.1103/physreva.94.032321 (2016).
[22] A. Javadi-Abhari, M. Treinish, K. Krsulich, C. J. Wood, J. Lishman, J. Gacon, S. Martiel, P. D. Nation, L. S. Bishop, A. W. Cross, B. R. Johnson, and J. M. Gambetta, Quantum computing with Qiskit (2024), arXiv:2405.08810 [quant-ph].
[23] S. Sivarajah, S. Dilkes, A. Cowtan, W. Simmons, A. Edgington, and R. Duncan, Quantum Science and Technology **6**, 014003 (2020).
[24] E. Younis, C. C. Iancu, W. Lavrijsen, M. Davis, E. Smith, and USDOE, Berkeley quantum synthesis toolkit (bqskit) v1 (2021).
[25] J. P. T. Stenger, N. T. Bronn, D. J. Egger, and D. Pekker, Physical Review Research **3**, 10.1103/physrevresearch.3.033171 (2021).
[26] N. Earnest, C. Tornow, and D. J. Egger, Phys. Rev. Res. **3**, 043088 (2021).
[27] E. C. Peterson, L. S. Bishop, and A. Javadi-Abhari, Quantum **6**, 696 (2022).
[28] C. Huang, T. Wang, F. Wu, D. Ding, Q. Ye, L. Kong, F. Zhang, X. Ni, Z. Song, Y. Shi, H.-H. Zhao, C. Deng, and J. Chen, Physical Review Letters **130**, 10.1103/physrevlett.130.070601 (2023).
[29] E. Nielsen, J. K. Gamble, K. Rudinger, T. Scholten, K. Young, and R. Blume-Kohout, Quantum **5**, 557 (2021).
[30] N. Sundaresan, I. Lauer, E. Pritchett, E. Magesan, P. Jurcevic, and J. M. Gambetta, PRX Quantum **1**, 10.1103/prxquantum.1.020318 (2020).
[31] K. X. Wei, E. Pritchett, D. M. Zajac, D. C. McKay, and S. Merkel, Physical Review Applied **21**, 10.1103/physrevapplied.21.024018 (2024).
[32] J. A. Gross, E. Genois, D. M. Debroy, Y. Zhang, W. Mruczkiewicz, Z.-P. Cian, and Z. Jiang, npj Quantum Inf. **10**, 123 (2024), arXiv:2404.12550 [quant-ph].
[33] E. Magesan, J. M. Gambetta, B. R. Johnson, C. A. Ryan, J. M. Chow, S. T. Merkel, M. P. da Silva, G. A. Keefe, M. B. Rothwell, T. A. Ohki, M. B. Ketchen, and M. Steffen, Physical Review Letters **109**, 10.1103/physrevlett.109.080505 (2012).
[34] J. M. Chow, A. D. Córcoles, J. M. Gambetta, C. Rigetti, B. R. Johnson, J. A. Smolin, J. R. Rozen, G. A. Keefe, M. B. Rothwell, M. B. Ketchen, and M. Steffen, Physical Review Letters **107**, 10.1103/physrevlett.107.080502 (2011).
[35] E. Magesan and J. M. Gambetta, Phys. Rev. A **101**, 052308 (2020).
[36] H. Ball, M. J. Biercuk, A. R. R. Carvalho, J. Chen, M. Hush, L. A. D. Castro, L. Li, P. J. Liebermann, H. J. Slatyer, C. Edmunds, V. Frey, C. Hempel, and A. Milne, Quantum Science and Technology **6**, 044011 (2021).
[37] J. Zhang, J. Vala, S. Sastry, and K. B. Whaley, Physical Review A **67**, 10.1103/physreva.67.042313 (2003).
[38] Y. Makhlin, Quantum Information Processing **1**, 243–252 (2002).
[39] B. Kraus and J. I. Cirac, Physical Review A **63**, 10.1103/physreva.63.062309 (2001).
[40] P. Watts, M. O'Connor, and J. Vala, Entropy **15**, 1963–1984 (2013).
[41] M. Sugawara and T. Satoh, Su(4) gate design via unitary process tomography: its application to cross-resonance based superconducting quantum devices (2025), arXiv:2503.09343 [quant-ph].
[42] T. Lubinski, S. Johri, P. Varosy, J. Coleman, L. Zhao, J. Necaise, C. H. Baldwin, K. Mayer, and T. Proctor, Application-oriented performance benchmarks for quantum computing (2023), arXiv:2110.03137 [quant-ph].
[43] S. Lloyd, Science **273**, 1073 (1996).
[44] M. Carroll, S. Rosenblatt, P. Jurcevic, I. Lauer, and A. Kandala, Dynamics of superconducting qubit relaxation times (2022), arXiv:2105.15201 [quant-ph].
[45] M. Kjaergaard, M. E. Schwartz, J. Braumüller, P. Krantz, J. I.-J. Wang, S. Gustavsson, and W. D. Oliver, Annual Review of Condensed Matter Physics **11**, 369–395 (2020).
[46] N. P. Sawaya, D. Marti-Dafcik, Y. Ho, D. P. Tabor, D. E. B. Neira, A. B. Magann, S. Premaratne, P. Dubey, A. Matsuura, N. Bishop, W. A. d. Jong, S. Benjamin, O. Parekh, N. Tubman, K. Klymko, and D. Camps, Quantum **8**, 1559 (2024).

Appendix A: Local equivalence (review)

Theorem 2 from [38] provides a procedure to get the complete set of local invariants of a 2-qubit gate U. To calculate these invariants, one needs the magic matrix Q which transforms computational basis states into the Bell basis.

$$Q = \begin{pmatrix} 1 & 0 & 0 & i \\ 0 & i & 1 & 0 \\ 0 & i & -1 & 0 \\ 1 & 0 & 0 & -i \end{pmatrix} \tag{A1}$$

In this basis, the unitary is first transformed into $U_B = Q^\dagger U Q$, with the spectrum of the matrix $m = U_B^T U_B$ being invariant under local transformations. The coefficients of the characteristic polynomial of m can be used to extract the invariants, which are

$$I_1(U) = \frac{1}{4}\,\mathrm{tr}(m),\;\; I_2(U) = \frac{1}{4}\left(\mathrm{tr}^2(m) - \mathrm{tr}(m^2)\right) \tag{A2}$$

These take the following values for 2-qubits gates U that are locally equivalent to the canonical gate $\mathbf{Can}(c_1, c_2, c_3)$:

$$I_1 = \cos c_1 \cos c_2 \cos c_3 - i \sin c_1 \sin c_2 \sin c_3 \tag{A3}$$
$$I_2 = \cos(2\,c_1) + \cos(2\,c_2) + \cos(2\,c_3) \tag{A4}$$

I_1 is complex, which provides 2 real invariants, while I_2 is a real number, which provides a single real invariant.

Appendix B: Weyl reflections

Weyl reflections are useful tools for synthesizing 2-qubit unitaries specified via their canonical coordinates [37]. We denote a set of 6 single qubit gates p_{ij}, p_{ij*} where $i, j \in 1, 2, 3$ whose action via conjugation on a canonical gate $\mathbf{Can}(c_1, c_2, c_3)$ implements 6 transformations: 3 permutations and 3 permutations along with sign flips. We denote the transformation acting on the indices ij to be the qubits being permuted and $ij*$ denote a sign flip and permutation on ij. For instance, acting on $\mathbf{Can}(c_1, c_2, c_3)$, a permutation exchanging $1, 2$ with and without a sign-flip is implemented by

$$p_{12}\ \mathbf{Can}(c_1, c_2, c_3)\ p_{12}^\dagger = \mathbf{Can}(c_2, c_1, c_3) \tag{B1}$$
$$p_{12*}\ \mathbf{Can}(c_1, c_2, c_3)\ p_{12*}^\dagger = \mathbf{Can}(-c_2, -c_1, c_3)$$

These are implemented in a circuit via the single qubit gates

$$p_{12} = \exp\left(i\pi/4\ \ (I \otimes Z + Z \otimes I)\right) \tag{B2}$$
$$p_{12*} = \exp\left(i\pi/4\ \ (I \otimes Z - Z \otimes I)\right)$$

All 4 remaining reflections are implemented by identifying the permutation indices $12, 23, 13$ with the single qubit gates Z, X, Y. These reflections and permutations are useful in steering trajectories to cover larger regions of the Weyl tetrahedron. They were used by [37] to prove that 3 applications of a continuous family of 2-qubit gates parameterized by t, $\mathbf{Can}(t\,c_1, t\,c_2, t\,c_3)$, can implement universal quantum computation, ie. cover the entire volume of the Weyl tetrahedron.

Appendix C: Circuit synthesis using single parameter two-qubit gates

We look in more detail at simplest example of synthesizing a target unitary using an available set of gates $\mathbf{Can}(c_{\text{eff}}^{(1)}, 0, 0)$ and $\mathbf{Can}(c_{\text{eff}}^{(2)}, 0, 0)$, which in more conventional notation are the same as single parameter 2-qubit rotations $XX(c_{\text{eff}}^{(1)})$, $XX(c_{\text{eff}}^{(2)})$. We would like to construct the feasible set of coordinates that can be synthesized using such a gate-set, The single qubit gates that implement Weyl reflections B allow one to implement permutations on the coordinates of either gate, so that a gate $\mathbf{Can}(c_{\text{eff}}^{(2)}, 0, 0)$ can be transformed into $\mathbf{Can}(-c_{\text{eff}}^{(2)}, 0, 0)$, $\mathbf{Can}(0, \pm c_{\text{eff}}^{(2)}, 0)$ or $\mathbf{Can}(0, 0, \pm c_{\text{eff}}^{(2)})$. This allows these gates to be placed in configurations that maximize the feasible region.

We first look at the problem of finding the feasible set of canonical coordinates (c_1, c_2, c_3) that can be reached by 2 such gates with any 2 single qubits s_1, s_2 inserted between them. The ansatz is of the form

$$\mathbf{Can}(c_1, c_2, 0) = \mathbf{Can}(c_{\text{eff}}^{(1)}, 0, 0)\ (s_1 \otimes s_2)\ \mathbf{Can}(0, c_{\text{eff}}^{(2)}, 0)$$

The feasible set of of coordinates are:

$$c_{\text{eff}}^{(1)} + c_{\text{eff}}^{(2)} \geq c_1 + c_2, \quad c_{\text{eff}}^{(1)} - c_{\text{eff}}^{(2)} \leq c_1 - c_2. \tag{C1}$$

The single qubit gates are $s_i = Z(\phi_i/2)$ with the angles given by:

$$\cos\phi_1 = -\cot c_{\text{eff}}^{(1)} \cot c_{\text{eff}}^{(2)} + \cos(c_1 - c_2) \csc c_{\text{eff}}^{(1)} \csc c_{\text{eff}}^{(2)}$$
$$\cos\phi_2 = \cot c_{\text{eff}}^{(1)} \cot c_{\text{eff}}^{(2)} - \cos(c_1 + c_2) \csc c_{\text{eff}}^{(1)} \csc c_{\text{eff}}^{(2)}$$

These solutions can be thought of as rotations in a 2-dimentional plane, with a fixed co-ordinate $c_3 = 0$, that is not controlled via these gates. The above set of solutions are equivalent to a set of solutions in which c_3 is a non-zero constant before and after the application of 2 gates.

In order to reach a target unitary described by 3 canonical coordinates (c_1, c_2, c_3), one needs to apply 3 gates. This is because a sequential application of a set of 2-dimensional rotations in the c_1-c_2 plane followed

by the c_2-c_3 plane allows synthesis to a target unitary with all three nonzero coordinates (c_1, c_2, c_3). This can be achieved an ansatz of the following form, with the 4 single qubit gates s_1, s_2, s_3, s_4 parameterized as $s_1 = Z(\phi_1/2),\ s_2 = Z(\phi_2/2),\ s_3 = X(\phi_1'/2),\ s_4 = X(\phi_2'/2)$

$$\mathbf{Can}(c_1.c_2, c_3) = \mathbf{Can}(c_{\text{eff}}^{(1)}, 0, 0)\ (s_1 \otimes s_2)\mathbf{Can}(0, c_{\text{eff}}^{(2)}, 0)$$
$$(s_3 \otimes s_4)\ \mathbf{Can}(0, 0, c_{\text{eff}}^{(3)})$$

In this case, there is freedom in the coordinates of the second step, since a single parameter family of coordinates can be further synthesized from this intermediate step. We parameterize this coordinate by δ, and the feasible region is

$$\begin{aligned} c_{\text{eff}}^{(1)} + \delta &\geq c_1 + c_2\,, \\ c_{\text{eff}}^{(1)} - \delta &\leq c_1 - c_2\,, \\ c_{\text{eff}}^{(2)} + c_{\text{eff}}^{(3)} &\geq \delta + c_3\,, \\ c_{\text{eff}}^{(2)} - c_{\text{eff}}^{(3)} &\geq \delta - c_3\,. \end{aligned}$$

with the following set of angles:

$$\begin{aligned} \cos\phi_1 &= \left(-\cot c_{\text{eff}}^{(1)} \cot\delta + \cos(c_1 - c_2) \csc c_{\text{eff}}^{(1)} \csc\delta\right) \\ \cos\phi_2 &= \left(\cot c_{\text{eff}}^{(1)} \cot\delta - \cos(c_1 + c_2) \csc c_{\text{eff}}^{(1)} \csc\delta\right) \end{aligned}$$

and

$$\begin{aligned} \cos\phi_1' &= \left(-\cot\delta \cot c_{\text{eff}}^{(3)} + \cos(c_2 - c_3) \csc\delta \csc c_{\text{eff}}^{(3)}\right) \\ \cos\phi_2' &= \left(\cot\delta \cot c_{\text{eff}}^{(3)} - \cos(c_2 + c_3) \csc\delta \csc c_{\text{eff}}^{(3)}\right) \end{aligned}$$

Appendix D: Conventions for mapping error amplifying sequence measurements to observables

For book-keeping of the results of the error amplifying sequences in Figure 2(a-d), we describe the conventions used to map the set of probe observables $\langle O_i \rangle$ to experimental measurements. The conventions are as follows: the reconstruction sequence in 2 (a-d) correspond to $\alpha = \{0, ..., 3\}$. The final state $\rho_{(\alpha,N)}$ is measured after N repetitions. The expectation value of the single qubit Z operator at qubit location $k = \{0, 1\}$ (denoting the control or target qubit) is measured via

$$\langle O(N) \rangle_{2\alpha+k} = \text{Tr}\left(\rho_{(\alpha,N)} Z_k\right) \tag{D1}$$

Appendix E: Runtime estimate of device wide characterization

Producing an enriched gateset requires additional circuit executions for characterization of the added gates. For a practical and useful scheme, this overhead should be relatively small. Therefore, it is useful to estimate the time taken to characterize a new set of gates to be added across the entire device. For N_{tot} number of circuits that need to be executed on a device, the total time is $N_{\text{tot}} \times \frac{1}{\text{rep-rate}}$ where rep-rate is the fixed number of circuit executions per-second for the machine. N_{tot} gets the following contributions:

$$N_{\text{tot}} = N_{\text{probe-circuits}} \times N_{\text{shots}} \times N_{\text{tot-reps}} \times N_{\text{covering factor}}$$

- $N_{\text{probe-circuits}}$: number different types of characterization circuits per qubit pair.
- $N_{\text{tot-reps}}$: number of gate repetitions per circuit
- $N_{\text{covering-factor}}$: number of independent experiments needed to cover each entangling pair on the device. The minimum value this can take is the minimum edge coloring of the device topology (3 for heavy-hex).

For IBM's reported rep-rate $= 4000$, and for our choices of $N_{\text{probe-circuits}} = 4, N_{\text{tot-reps}} = 8$ and $N_{\text{covering factor}} = 4$, we get a full device characterization QPU time of $150s$.