An integrated tool-set for Control, Calibration and Characterization of quantum devices applied to superconducting qubits

Nicolas Wittler,[1,2,*] Federico Roy,[1,3,*] Kevin Pack,[1,2] Max Werninghaus,[1,3] Anurag Saha Roy,[4] Daniel J. Egger,[3] Stefan Filipp,[3,5] Frank K. Wilhelm,[1,2] and Shai Machnes[1]

[1] *Theoretical Physics, Saarland University, 66123 Saarbrücken, Germany*

[2] *Peter Grünberg Institut – Quantum Computing Analytics (PGI 12), Forschungszentrum Jülich, D-52425 Jülich, Germany*

[3] *IBM Quantum, IBM Research GmbH, Zurich Research Laboratory, Säumerstrasse 4, 8803 Rüschlikon, Switzerland*

[4] *Citizen scientist*

[5] *Department of Physics, Technical University of Munich, 85748 Garching, Germany*

Efforts to scale-up quantum computation have reached a point where the principal limiting factor is not the number of qubits, but the entangling gate infidelity. However, a highly detailed system characterization required to understand the underlying errors is an arduous process and impractical with increasing chip size. Open-loop optimal control techniques allow for the improvement of gates but are limited by the models they are based on. To rectify the situation, we provide a new integrated open-source tool-set for Control, Calibration and Characterization ($\mathbb{C}^3$), capable of open-loop pulse optimization, model-free calibration, model fitting and refinement. We present a methodology to combine these tools to find a quantitatively accurate system model, high-fidelity gates and an approximate error budget, all based on a high-performance, feature-rich simulator. We illustrate our methods using fixed-frequency superconducting qubits for which we learn model parameters to an accuracy of $< 1\%$ and derive a coherence limited cross-resonance (CR) gate that achieves 99.6% fidelity without need for calibration.

I. THE PROBLEM

Scaling up QPUs (Quantum Processing Units) is a monumental task, that requires the community to make progress on multiple fronts, most importantly improving gate fidelities and increasing the number of qubits. Over the past few years, significant emphasis has been placed on creating larger devices, yielding great success [1, 2]. However, the number of qubits has outstripped the limits fidelity places on their utility: In [1], a record quantum volume [3] of 64 was demonstrated, loosely translating to the device being able to perform only $\log_2(64)^2 = 36$ entangling gates before fidelity drops below $^2/_3$, a relatively small number of gates for a chip with 26 qubits; In [2] the circuit fidelity was 0.1% thus requiring 30 million repetitions to achieve the desired statistics. One could even argue that the two-qubit gate fidelities demonstrated in isolation in 2014's [4] (0.994) are comparable with those in 2019's [2] (0.9938), even though the latter are for simultaneous gates in a large 2D qubit array.

The relatively slow progress in improving gate fidelities can be traced back to an incomplete understanding of the sources of error. Indeed, characterization and calibration of QPUs to the desired accuracy is impractical and cumbersome, and operating on devices of increasing qubit number requires entangling gates to be fine-tuned for each individual pair to account for slightly varying properties. The resulting lack of detailed models makes it harder to identify where efforts must be focused to achieve higher fidelity gates [5, 6].

Given that "all models are wrong, but some are useful" [7], we describe a Good Model as follows:

> A *Good Model* is one which predicts the behavior of the system, for the operations we wish to perform, to accuracies we care about.

For a QPU, a Good Model has to have predictive power for the range of feasible gate-generating pulses and for long sequences of such gates, to a fidelity accuracy on the order of 10^{-5}. To the authors' knowledge, no such Good Model for a superconducting QPU has ever been published.

Since models serve as the basis to derive high-fidelity gates in open-loop optimal control [8–15], any inaccuracies of the model will inevitably degrade the experimental accuracy of the resulting gates. This problem is only partially ameliorated by the first-order insensitivity of optimized pulses to model inaccuracies [16, 17]. Methodologies such as Ad-HOC [18] — which combines a model-based open-loop optimization with a closed-loop experimental calibration [12, 19] — address this issue but leave one in an unsatisfactory position as the need for calibration proves the inadequacy of the model: the root causes of the remaining infidelities are unexplained.

Conversely, if a Good Model is known, gates generated by open-loop optimal control will, by definition, work

* These two authors contributed equally; c3@q-optimize.org

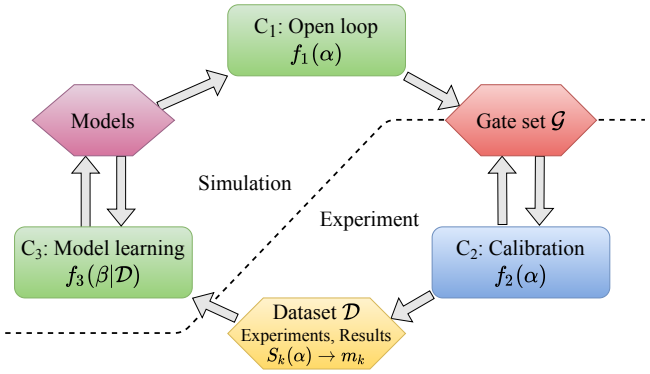


FIG. 1: Diagram of the $\mathbb{C}^3$ tool-set in an integrated characterization loop. $\mathbf{C}_1$ is a tool for obtaining optimal pulses by finding the control parameters α that minimize a goal function $f_1(\alpha)$ in simulation. The **gate-set** $\mathcal{G}$ includes all the operation that one wishes to perform on the experiment, including the information of the ideal logical operations and the optimal pulse parameters α that implement them. $\mathbf{C}_2$ is a model-free experimental calibration procedure that optimizes pulse shapes with a gradient-free search to minimize an infidelity function $f_2(\alpha)$ by varying all parameters at once. A **data-set** is a collection of experiment/result pairs, including information about the pulses parameters used α, the sequences S_k performed and the final outcomes measured, m_k. $\mathbf{C}_3$ is a tool for model learning that determines the model parameters β that best explain the data-set. It minimizes a goal function $f_3(\beta|\mathcal{D})$ obtained by recreating experiments $S_k(\alpha)$ in simulation and comparing the results to the ones in the experiment. In $\mathbb{C}^3$ different parametrized models can be provided to represent various elements of the experiment to find the one that best describes it. After the learning, the resulting model can be the basis for another characterization loop, refining both model and controls.

on the experiment, not requiring further closed-loop calibration. This enables the use of complex pulses which would otherwise require time-consuming calibration: A quantum computer that is only ever calibrating itself is useless. Such a Good Model would also provide an error budget through a process of exploratory interrogation – evaluating the potential performance of the system where certain limitations have been removed, i.e. asking "what if... ?". Therefore, extracting a Good Model efficiently and in a highly automated manner is key to improving fidelities and a crucial step of QPU scale-up.

In this work we present $\mathbb{C}^3$, our proposed approach to control, calibrate and characterize QPUs. The paper is organized as follows: In Sec. II we present the conceptual steps of $\mathbb{C}^3$, and in Sec. III we illustrate the $\mathbb{C}^3$ methodology by example, showing how these steps are implemented. This is followed-up by a more in-depth look into $\mathbb{C}^3$ in Sec. IV, detailing the experiment modelling and simulation, choices of optimization algorithms and cost functions, and methods of model analysis. We conclude with a discussion of the effort's current status and long-term directions. The supplementary material includes: A review of prior work in this field, a comparison of gradient-free algorithms, a description of the open-source implementation and a mathematical framework to compare simulation and experimental data.

II. $\mathbb{C}^3$ - CONTROL, CALIBRATION AND CHARACTERIZATION

Current methodology relies on tailored routines to extract individual parameters of the system's model (characterization) or fine-tune specific parameters of pulses used (calibration) [20]. This approach becomes cumbersome and impractical as the number of model and pulse parameters increase. With $\mathbb{C}^3$ we propose a different paradigm: Optimizing a figure of merit that is sensitive to the set of parameters we care about. This eliminates the need to design per-parameter measurements, and thus provides a more general approach. $\mathbb{C}^3$ at its core is comprised of three separate optimizations, implementing respectively the tasks of control, calibration and characterization:

$\mathbf{C}_1$: Given a model, find the pulse shapes which maximize fidelity with a target operation. Pulse shapes may be constrained by an ansatz or allow direct arbitrary waveform generator (AWG) parameterization. For details see Sec. IV B 1.

$\mathbf{C}_2$: Given pulses, calibrate their parameters to maximize a figure of merit measured by the actual experiment, thus improving beyond the limits of a deficient model. All pulse parameters are calibrated simultaneously. For details see Sec. IV B 2.

$\mathbf{C}_3$: Given control pulses and their experimental measurement outcome, optimize model parameters to best reproduce the results. Enhance the model if needed. For details see Sec. IV B 3.

The tasks of open-loop optimal control, C_1, and calibration, C_2, are fairly established in the community [8–15, 18, 19]. To characterize the system and provide us with a Good Model we introduce C_3, a tool to optimize model parameters by comparing model prediction to experimental data. We refer to this task as model learning. For this purpose, one requires an experimental data-set containing information about the implemented pulses and the corresponding measurement outcomes. To test the model accuracy, we reproduce the data-set, applying the same pulses to a simulation of the experiment, and compare the resulting outcomes, providing a model match score to optimize. Initially, a candidate model is formulated based on previous information or intuition. If the model is suitable to explain the experiment, the optimization will converge to a near perfect match, thus providing numeric values for the model parameters. Instead, if the match is poor, the user supplies a new model, that is either an extension or modification of the previous candidate, and the optimization is repeated. Depending on user choice, learned values are carried over to the parameters of the new model or discarded.

As heterogeneous experimental data is the foundation for model learning, we suggest using the three tasks of $\mathbb{C}^3$ in sequence, as shown in Fig. 1. However, their application is by no means limited to this use case and one may choose to view them as stand-alone routines. The same tools used to realize C_1, C_2 and C_3 can also be used to further interrogate the system: A sensitivity analysis of the optimized model in light of the experimental data and a breakdown of possible error sources.

For most quantum systems simplified models already provide a good qualitative and partly quantitative understanding of their behavior. However, it is a-priori unclear which physical effects must be included in the model for it to satisfy the criteria of a Good Model. $\mathbb{C}^3$ provides a structure with which to explore possible explanations for observed system behavior, reducing the need for laborious, ad-hoc characterization work. Physical intuition and established experimental practice, are not replaced by $\mathbb{C}^3$, rather they inform decisions while using it.

A major practical challenge of all three optimizations is the existence of false or local minima, resulting in suboptimal solutions. A properly chosen search algorithm can alleviate the problem (see discussions in Section IV) but does not eliminate it. Experimental practice suggests a methodology to further reduce this problem by progressing through a series of successively refined models and control complexities, where in each step one builds upon knowledge gathered from the previous step. We suggest to use this methodology when deemed appropriate, helping the algorithms reach optimal solutions.

Furthermore, much attention has been given over the years to the poor scalability of optimal control, calibration and characterization [21–24]. However, the performance of a QPU as a whole can be determined by the performance of subsets of qubits (sets of overlapping nearest-neighbour or next-nearest-neighbour localized tiles) [2], thus making application of $\mathbb{C}^3$ feasible for large-scale QPUs.

III. SYNTHETIC APPLICATION EXAMPLE

The following synthetic example illustrates how $\mathbb{C}^3$ is used to obtain a Good Model in an realistic setting. We simulate a two-qubit QPU device using an underlying model, labeled the "real" model, which includes control discretization effects, electronics transfer functions, Markovian noise, and state preparation and measurement (SPAM) errors.

In this example, the simulated device is treated as a black-box, which we interrogate with $\mathbb{C}^3$. We derive (C_1) and calibrate (C_2) optimal control pulses and use the resulting data to extract a Good Model (C_3) by comparing the black-box to three candidates:

Simple model: Two *uncoupled* qubits, *closed* system dynamics;

Intermediate model: Two *coupled* qubits, *closed* system dynamics;

Full model: Two *coupled* qubits, *open* system dynamics, including *SPAM* errors. Same structure as the "real" model, but a-priori undetermined parameter values.

We systematically enrich the model until it reproduces the behavior of the device observed in C_2. The recovered model is then used to design a two-qubit gate and show that it yields the same performance as on the "real" model. A general description of the simulation and optimizations follows in Sec. IV.

A. The black-box device ("real" model)

The "real" model is composed of two coupled 3-level Duffing oscillators, labeled by A and B, each directly driven by an external field $c_i(t)$. Initialization, dynamics and readout are performed in the dressed basis by solving the Master Equation in Lindblad form [25, 26].

$$\dot{\rho} = -i[H,\rho] + \sum_{\substack{i=A,B \\ j=\phi,\kappa}} L_{i,j}\rho L_{i,j}^\dagger - \frac{1}{2}\Big\{L_{i,j}L_{i,j}^\dagger, \rho\Big\} \tag{1}$$

with

$$\begin{aligned} H/\hbar = \sum_{i=A,B}\left[\omega_i b_i^\dagger b_i - \frac{\delta_i}{2}\Big(b_i^\dagger b_i - 1\Big)b_i^\dagger b_i\right] + \\ +g(b_A + b_A^\dagger)(b_B + b_B^\dagger) + \sum_{i=A,B} c_i(t)\Big(b_i + b_i^\dagger\Big)\ , \end{aligned} \tag{2}$$

where ω_i is the frequency of qubit i, δ_i is the anharmonicity, b_i ($b_i^\dagger$) is the raising (lowering) operator, and g is the coupling strength. Open-system effects are expressed by the dephasing and relaxation Lindblad operators $L_{i,\phi} = b_i b_i^\dagger/T_i^{2*}$ and $L_{i,\kappa} = b_i/T_i^1$ with decay rates $1/T_i^1$ and $1/T_i^{2*}$.
Given the input drive signal $\varepsilon(t)$, we calculate the effective control field $c(t) = \varphi[\varepsilon(t)]$, where the transfer function φ [27] accounts for discretization introduced by the AWG, bandwidth limitations of hardware, and for a constant scaling φ_0, which translates input voltages to field amplitudes $c(t)$. We account for state preparation errors due to a non-zero initial temperature T by starting each experiment from the thermal state

$$\begin{aligned} \rho_{\text{init}} = \frac{1}{Z}\Bigg[|0\rangle\langle 0| + \exp\left\{-\frac{\hbar\omega_q}{k_B T}\right\}|1\rangle\langle 1| \\ + \exp\left\{-\frac{\hbar(2\omega_q+\delta)}{k_B T}\right\}|2\rangle\langle 2|\Bigg] \end{aligned} \tag{3}$$

where $Z = \sum_{k=0}^{2}\exp\{-E_k/k_B T\}$ is the partition function with energies $E_{0,1,2} = 0, \hbar\omega_q, \hbar(2\omega_q+\delta)$, and k_B is the

Boltzmann constant. Read-out mis-classification is included, measuring state $|n\rangle$ as state $|m\rangle$ with probability $p_{n\to m}$. For example the probability of measuring a state $\rho_\psi = |\psi\rangle\langle\psi|$ as $|0\rangle\langle 0|$ is

$$\Pi_0(\rho_\psi) = p_{0\to 0}\ \langle 0|\rho_\psi|0\rangle + \ p_{1\to 0}\ \langle 1|\rho_\psi|1\rangle + \ p_{2\to 0}\ \langle 2|\rho_\psi|2\rangle\,. \tag{4}$$

Similarly to experiment, populations are estimated by averaging the results of multiple projective measurements, simulated as a binomial draw from the distribution with probability Π_n, thus introducing noise stemming from a finite number of measurement repetitions (commonly known as shot noise). The values of the parameters of the "real" model are in Supplement S6.

B. Open-loop Optimal Control, C_1

We assume that at the start of the $\mathbb{C}^3$ procedure the parameters of the system are only known to a rough precision, with its qubit frequencies and anharmonicities chosen to be within a few MHz of their "true" values. In the simple model, the qubits are *uncoupled* 3-level Duffing oscillators, evolution follows *closed* systems dynamics, and state preparation and measurement are assumed perfect. The Hamiltonian is

$$H/\hbar = \sum_{i=A,B} \omega_i b_i^\dagger b_i - \frac{\delta_i}{2}\Big(b_i^\dagger b_i - 1\Big) b_i^\dagger b_i + c_i(t)\Big(b_i + b_i^\dagger\Big). \tag{5}$$

Assuming this model, we design pulses for single-qubit gates using C_1. To mitigate leakage, we choose a pulse ansatz with a Gaussian shape and a DRAG correction [14],

$$\varepsilon(t) = A\,\Omega_{\text{Gauss}}(t)\,\cos((\omega_d + \omega_{\text{off}})t + \phi_{xy}) - \frac{\eta}{\delta}\,A\,\dot{\Omega}_{\text{Gauss}}(t)\,\sin((\omega_d + \omega_{\text{off}})t + \phi_{xy}). \tag{6}$$

Here, Ω_{Gauss} is a Gaussian envelope, $\dot{\Omega}_{\text{Gauss}}(t)$ is its time derivative, A is the amplitude of the drive, ω_{off} is a frequency offset and the DRAG parameter η can be adjusted to reduce leakage into the second excited state [14, 28]. The rotation axis can be freely chosen in the xy-plane by changing the phase of the drive signal $\omega_d t \to \omega_d t + \phi_{xy}$, implementing the unitary rotations $R(\phi_{xy}, \theta) = \exp\{-i(\cos\phi_{xy}\sigma_x + \sin\phi_{xy}\sigma_y)\theta\}$. By setting $\phi_{xy} = n\frac{\pi}{2}$ with $n = 0, 1, 2, 3$ and changing $\alpha = (A, \eta, \omega_{\text{off}})$ we aim to realise the single qubit gate-set

$$\mathcal{G} = \left\{X_{\pi/2}, Y_{\pi/2}, X_{-\pi/2}, Y_{-\pi/2}\right\}\,, \tag{7}$$

for each qubit separately, eight gates in total.

With C_1 we use a gradient descent method to find the parameters α that minimize the mean average gate infidelity

$$f_1(\alpha) = 1 - \frac{1}{|\mathcal{G}|}\sum_{U\in\mathcal{G}} f_{\text{av}}(U) = 1 - \frac{1}{|\mathcal{G}|}\sum_{U\in\mathcal{G}} \frac{\chi_{0,0}d + 1}{d+1}, \tag{8}$$

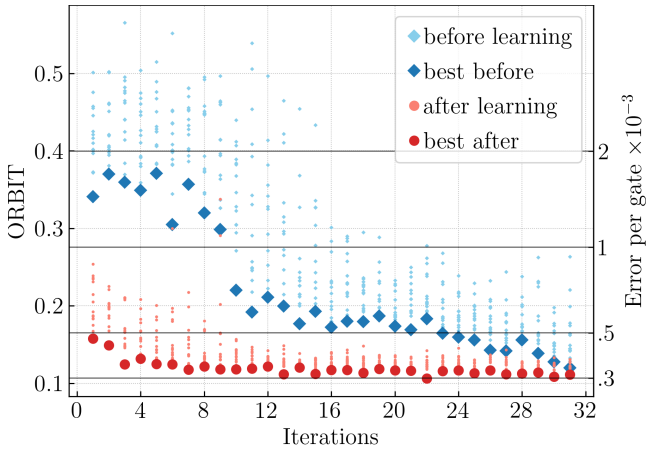


FIG. 2: C_2 calibration on the device for single-qubit gates of qubit B. The initial point is suggested by C_1 before (after) learning of the model. The light blue diamonds (light red dots) represent the values of the ORBIT goal function, Eq. (9), for varying pulse parameters α as chosen by the search algorithm. The larger blue diamonds (larger red dots) highlight the best of 25 points generated and sampled at each iteration. In experimental practice, this batching helps reduce the overhead of loading pulses in AWG programming [35]. Both calibrations achieve the same final fidelity, however the optimal gates derived from the learned model provide a better initial guess. Assuming no SPAM errors the ORBIT value can be translated into an error per gate, indicated on the right axis. This is only meant to provide a rough estimate of the performance of the gate, noting that an ORBIT value of 0.5 represents maximum error per gate, i.e. completely depolarizing channels.

where, $\chi_{0,0}$ is the $(0,0)$-th element of the Chi matrix representation of the gate error $U^\dagger \cdot \widetilde{U}(\alpha)$ between the ideal gate U and the implemented gate $\widetilde{U}(\alpha)$ [29].

We optimize Gaussian pulses with a gate length of $t_g = 7$ ns, for both qubits, using the gadient-based L-BFGS algorithm [30]. The obtained optimal pulses yield a mean infidelity of $f_1(\alpha) = 6.6\times10^{-4}$ and $f_1(\alpha) = 4.9\times10^{-4}$ on the simple model for qubit A and qubit B respectively – realistic values for fast gates using this simple parametrization. Next, we compare the performance of these pulses on the black-box device, where the gates instead yield a mean infidelity of 2.4×10^{-3} for qubit A and 1.5×10^{-3} for qubit B. In fact, performing an experimentally realistic randomized benchmarking (RB) [31–34] measurement on the device yields an error per gate of 2.3×10^{-3} and 1.3×10^{-3} comparable with the theoretical average infidelity. The degradation of performance from optimal control simulation ($\approx 10^{-4}$) to experiment ($\approx 10^{-3}$) shows a clear mismatch between the device and the simple model.

C. Calibration, C_2

The next step is to calibrate the pulses derived by C_1 and improve their performance on the device. We use C_2 and employ a closed-loop, model-free, gradient-free optimization algorithm on an experimentally accessible

figure of merit f_2. Since we want to evaluate a gate-set, we choose f_2 to be the ORBIT [19] (single-length RB) goal function

$$f_2(\alpha) = f_{\mathrm{ORBIT}}(\alpha) = \frac{1}{N}\sum_{k=1}^{N}(1 - m_k(\alpha)), \qquad (9)$$

averaging over N sequences. The survival probability, $m_k = \Pi_0\Big(S_k(\alpha)\rho_{\mathrm{init}}S_k^\dagger(\alpha)\Big)$, is the probability to measure the state $|0\rangle$ (see Eq. 4) after applying random sequences

$$S_k(\alpha) := \left\{\prod_j^{L-1} C_{k,j}\right\} C_{\mathrm{inv}} \qquad (10)$$

composed of L Clifford gates [19], to the initial thermal state ρ_{init}. The $C_{k,j}$ are the random gates sampled from the Clifford group C (for a single qubit $|C| = 24$), and C_{inv} is chosen so that $S_k \equiv \mathcal{I}$ in the ideal case. We use the atomic operations $\mathcal{G}$ from Eq. (7) to construct the set of Clifford gates, e.g. $C_6 = X_{-\pi/2} \circ Y_{-\pi/2} \circ X_{\pi/2}$, and from them construct $N = 25$ RB sequences of length $L = 100$. The survival probabilities m_k are estimated by performing $s = 1000$ projective measurements and averaging.

To minimize f_2, we employ the CMA-ES [36] algorithm - a gradient-free search that samples the loss function in batches, and is fairly robust to local minima and noise. See [35] for an experimental demonstration. The optimal pulse parameters from C_1 are used as the starting point of the optimization, and the parameterization is kept as in Eq. (6). We perform the calibration for each qubit independently, with similar results. See Fig. 2 for the ORBIT calibration data of qubit B. The initial point suggested by C_1 has an ORBIT infidelity of 0.50 and is improved by the optimization to 0.12. To account for SPAM errors, we perform a full RB measurement and estimate the infidelity of the gates before and after as 1.3×10^{-3} and 3.4×10^{-4} respectively. For qubit A we get a similar improvement of RB estimated error from 2.3×10^{-3} to 7.5×10^{-4}.

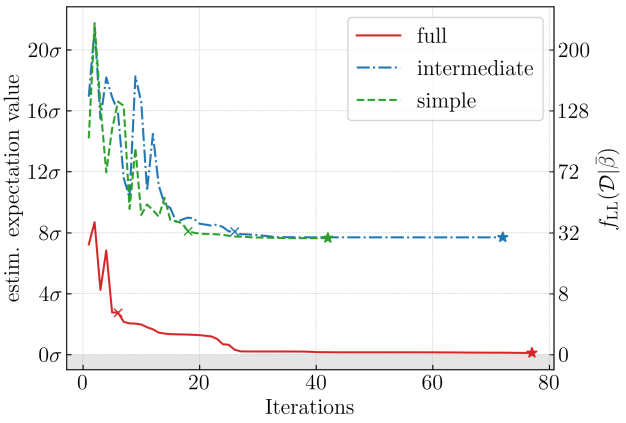


FIG. 3: Progress of the C_3 optimization on a hierarchy of models: simple model (green, dashed), intermediate model (blue, dot-dashed) and full model (red, solid), as described in the text. The Model match goal function $f_{\mathrm{LL}}(\beta)$ is defined in Eq. (26). The crosses show the switch-over from CMA-ES to L-BFGS. The CMA-ES algorithm evaluates a batch of points for each iteration (8, 9 and 12 for the simple, intermediate and full model respectively), only the best of each batch is shown. The L-BFGS algorithm takes on average approximately 1.2 evaluations per iteration, for all three models. The function f_{LL} is rescaled to express the match in terms of standard deviations of the binomial distribution that the experimental results are drawn from. The simple model is a close dispersive approximation of the intermediate model, demonstrated by their similar final match score. By including all relevant device properties the full model reaches an almost perfect match score.

We define the data-set $\mathcal{D} := \{S_k(\alpha_j) \to m_{j,k}\}$ as the collection of the experiments conducted during the C_2 calibration, consisting of pulse parameters α_j, gate sequences $S_k(\alpha_j)$ and corresponding measurement outcomes $m_{j,k}$.

D. Characterization, C_3

In C_3, we use the data-set $\mathcal{D}$ obtained during ORBIT calibration to improve the model of the system. For each measurement result $m_k(\alpha_j)$ we compute the equivalent simulation result $\widetilde{m}_k(\alpha_j, \beta)$ by calculating the dynamics of the sequence $S_k(\alpha_j)$ given a set of model parameter values $\beta = (\omega_i, \delta_i, ...)$. We then construct a goal function $f_3(\beta) = f_{\mathrm{LL}}(\mathcal{D}|\beta)$, Eq. (26), that captures how well the model prediction $\widetilde{m}_{j,k}$ agrees with the recorded values $m_{j,k}$. Since simulating the whole data-set is computationally costly, for the purpose of model learning we make a selection of eight pulse parameter sets j per qubit from the full data-set. Each parameter set includes $k = 1, .., 25$ sequences, meaning that we learn from a total of 400 measurement results $m_{j,k}$. Only after the optimization, we validate the optimal point by evaluating the goal function on the whole data set. To minimize $f_3(\beta)$, we use a combination of two algorithms: Gradient-free (CMA-ES) to avoid local minima and gradient-based (L-BFGS) to converge quickly once the right minimum has been identified.

Due to finite number of measurements, the averaged $m_{j,k}$ are noisy estimates of the population, with a mean $\mu_{j,k}$ and standard deviation $\sigma_{j,k}$. Thus, they cannot be matched perfectly even when all model parameters are exact. However, we can determine the expectation value of the goal function f_{LL} in the scenario where all $\widetilde{m}_{j,k}$ are exactly n standard deviations away from the underlying true value $\mu_{j,k}$. A detailed mathematical discussion is in the supplementary material S4. To provide a more intuitive measure, we will express the match f_{LL} in terms of the number of standard deviations n that would result in the same score.

Fig. 3 shows the convergence of the C_3 optimization for different models. The simple model is not able to reproduce the device's results, as the optimization ends at approximately 8 standard deviations away. This demon-

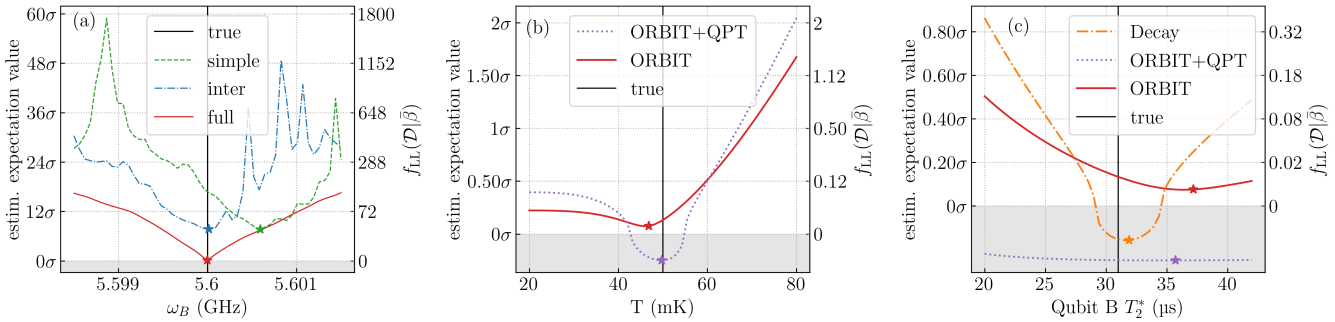


FIG. 4: Sensitivity analysis of selected parameters for different models and data sets. In (a) we sweep the qubit frequency ω_B and evaluate the goal function $f_{\rm LL}$ on the learning data at each value. The star represents the optimal value returned by C_3. Intermediate (blue, dot-dashed) and Full (red, solid) models, show the same frequency value, while the value of the Simple model (green, dashed) is dispersively shifted, as expected. In (b) and (c) we perform the same sweep for the chip temperature and T_2^* of qubit B respectively, evaluated on the full model for different learning data-sets. The ORBIT data is the same used in the full model in (a). Introducing the Quantum Process Tomography (QPT) data (purple, dotted), allows a more precise definition of the temperature. To determine T_2^* we use relaxation and dephasing data (orange, dot-dashed). Match values below 0 can occur because of noisy data, finite sampling and deviation from the assumption of Gaussian distribution of the data. More sensitivity plots are shown in Supplement S5.

strates that the experiment on the device includes behaviour not captured by the simple model. The parameter values resulting from this C_3 process and all following ones are shown in Supplement S6. Going forward an informed decision has to be made about how to enhance the model. Since the true values of the parameters are not known in an experimental setting, we require a tool to determine the precision to which they are learnt. We estimate the sensitivity to changes of model parameters around the optimal values β' by performing one-dimensional scans and observing the degradation in model match score, $f_{\rm LL}(\mathcal{D}|\beta' + \delta\beta)$. Fig. 5a shows that sweeping the value of frequency of qubit B produces a highly irregular landscape of the match score $f_{\rm LL}$.

Spectator effects might be significant even when performing only single qubit experiments, making the completely uncoupled model insufficient. Another source of this inconsistency might be SPAM errors not accounted for in the model, that might play a large role in actual measurement results. The first extension of the simple model to the intermediate model is done by adding the static coupling g of unknown exact value. When repeating C_3, we initialize model parameters from the starting rough values. We do not carry over the learned parameters from the simple model to the intermediate model because, by introducing a coupling, we expect slightly shifted frequencies compared to the bare frequencies of the uncoupled qubits. Convergence of the match score shows no improvement from the simple model, still only reaching within approximately 8 standard deviations from experiment results (Fig. 3) and resulting in a similar sensitivity landscape in Fig. 5a. This suggests that the simple model is a close dispersive approximation of the intermediate model. Indeed, we observe a dispersive shift [37] of 593 KHz, consistent with the expected $g^2/(\omega_B - \omega_A) \simeq$ 666 KHz, given the coupling of $g \simeq 20$ MHz and the frequency difference $\omega_B - \omega_A \simeq 600$ KHz.

Finally, model complexity is increased by adding three relevant features: Markovian noise simulated by Lindblad Master equation, initialization errors due to finite operating temperature and measurement errors in the form of mis-classification. The system model is now of the same structure as the "real" model of the device. Starting from the best intermediate model parameters, the C_3 procedure converges satisfactorily, approaching the 0 standard deviations mark (Fig. 3)

In Fig. 6 we show the value of each parameter of the full model during optimization, as we introduce different learning data (in the next sections), and compare with their true value (dashed lines). By learning the model parameters with the ORBIT data (white left section of each plot) the model frequencies $\omega_{A/B}$, anharmonicities $\delta_{A/B}$, coupling g and line transfer function φ_0 converge to their true value. The temperature and mis-classification parameters are not recovered, and we believe this is due to an extra degree of freedom that is not decoupled by the experiments we have performed, as the effects of mis-classification, Eq. (4), and initial thermal distribution, Eq. (3), are similar and can be partially exchanged. Dephasing and relaxation times (not shown) are also not recovered. In Fig. 5c we show that the sensitivity of the data to dephasing time T_2^* of qubit B is minimal. Similar observations can be made for the other open system parameters, see Supplement S5. RB sequences perform an effective random dynamical decoupling [38], providing a possible explanation to this result.

E. Validation of the learned model

After model matching on a subset of the data in the C_3 step, we now evaluate the predictive power of the learned models by computing the score on the rest of the data set (this is also known as a validation set in machine learning). This verifies that the selected subset

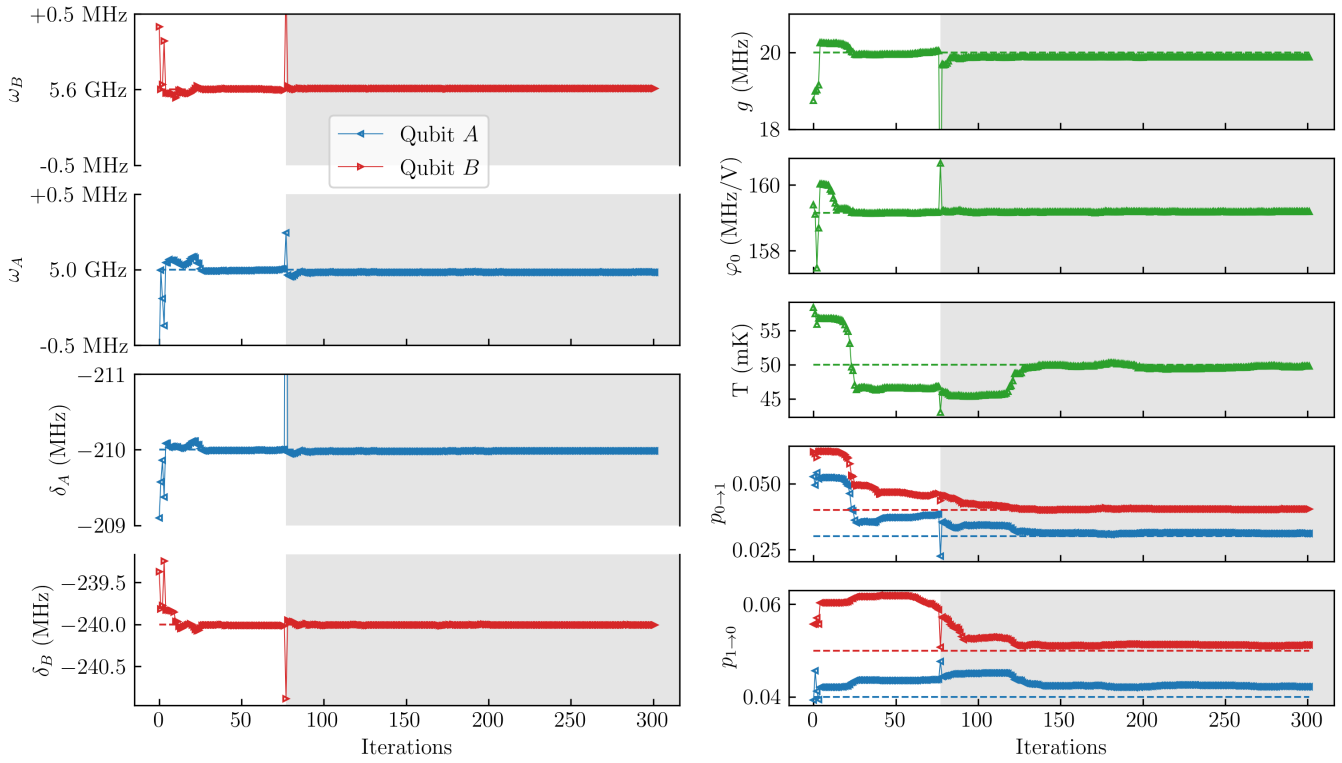


FIG. 6: C_3 learning of the two-qubit model parameters. Blue, left (red, right) triangles indicate qubit A (B) parameters, respectively, while shared properties are shown with green upwards triangles. The true values of the "real" model are indicated as dashed lines. Learning begins using just ORBIT data (left white section) that fixes qubit frequencies, anharmonicities, coupling and and line transfer function to their true values. Then tomography data from a two qubit experiment is added (right grey section), which allows to better identify the chip temperature T and the mis-classification constants $p_{0\to1}$, and $p_{1\to0}$.

captures all relevant behavior and alleviates the danger of overfitting.

Fig. Figures 7a to 7c depict the correlation between calibration data points $m_{j,k}$ and their model-based reconstructions $\widetilde{m}_{j,k}$. We evaluate the goal function $f_{\mathrm{LL}}(\beta)$ over the validation set for the Simple, Intermediate and Full models and obtain values of 36.5 ($\approx 8.4\sigma$), 42.0 ($\approx 9.2\sigma$) and 0.028 ($\approx 0.2\sigma$) respectively. The conclusion is that, even though some parameters were not recovered by C_3, the learned full model is indeed a Good Model, as it reproduces the behaviour of the system on *all previously recorded data points* to satisfying accuracy. This does not prevent additional measurement data to expose new behavior in the system: The notion of the Good Model is always tied to the underlying data-set.

Furthermore, we now repeat the C_1 procedure on the Good Model (yielding average gate infidelities of 6.3×10^{-4} and 1.1×10^{-3} for qubit A and B respectively) and show that the resulting pulses give a near optimal performance on the actual device and allow for faster C_2 convergence, as seen in Fig. 2. One would expect the pulses derived from the Good Model to be exactly optimal on the actual device. Even though it is not the case here, this is not because of an inaccurate model, but rather because of a disparity between the figures of merit used in C_1 (average infidelity) and C_2 (single-qubit ORBIT). Average fidelity captures effects of the whole system, including, in this case, an effective ZZ-coupling between the two qubits caused by a slight repulsion of the $|02\rangle$ and $|11\rangle$ states, that are 300 MHz apart. Minimizing a single-qubit ORBIT infidelity does not adjust for this effect, as we can verify by evaluating both RB (which captures only one qubit at a time) and average infidelity before and after calibration. Indeed, the average infidelity of qubit B is 1.2×10^{-3} (compatible with the performance of 1.1×10^{-3} on the Good Model) but the error per gate is estimated by RB as 4.1×10^{-4}. After the calibration the RB estimate is improved to 2.9×10^{-4} but the average infidelity is worsened to 1.9×10^{-3}. Performing simultaneous RB could resolve this issue.

F. Entangling gate

We further investigate the Good Model which was determined using only single-qubit calibration data by deriving a two-qubit cross-resonance (CR) gate [39, 40] with C_1. Both qubits are driven simultaneously at ω_B, the resonant frequency of qubit B, to accumulate a phase $\pm\pi/2$ conditioned on the state of qubit A [20]. Both drives are parametrized by flattop Gaussians. The resulting CR pulse has a gate infidelity of $f_{\mathrm{av}} = 3.8\times10^{-3}$. When

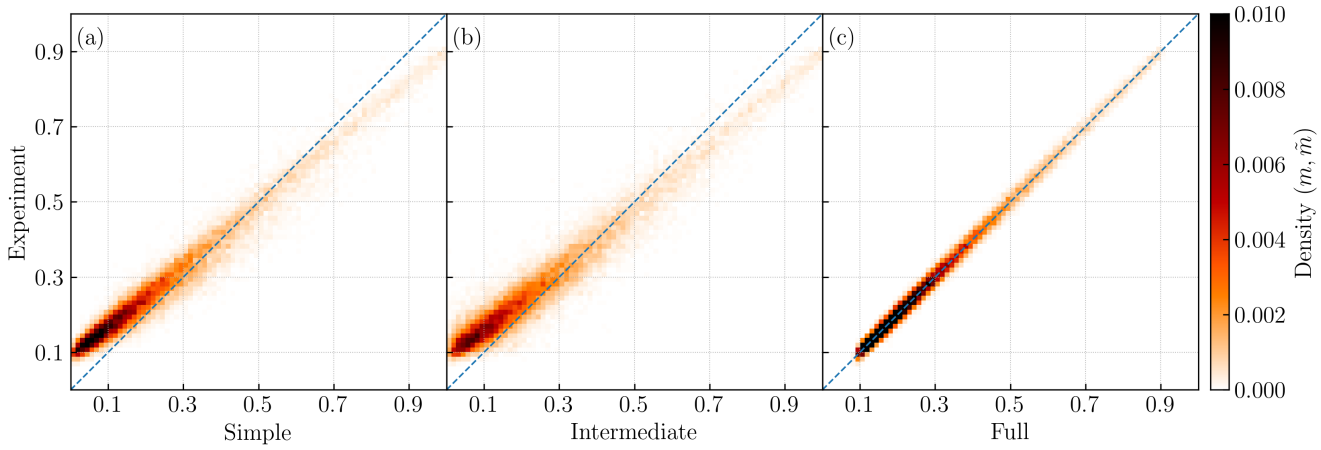


FIG. 7: (a)-(c): Correlation between experiment and simulation expressed as a density of points $(\widetilde{m}_k, m_k)$ for the Simple, Intermediate and Full model, respectively. The data-set is the so called validation set: The data points k that were not used in optimization of the model parameters. The simple and intermediate models show poor correlation as the simulation predicts a wide distribution of measurement outcomes for each recorded value. They also exhibit a tilt that can be attributed to SPAM errors not considered in the models. Only the full model produces a consistently high density distribution centered around the diagonal, with minimal spread due to the noisy data.

evaluated on the "real" model the gate has an infidelity of $f_{\text{av}} = 4.3\times10^{-3}$, again showing that the learned model predicts device behaviour to high accuracy. Notably, the model learned using only single-qubit data was sufficient to accurately predict the performance of the two-qubit gate on the device. We suspect this to be caused by exchange interactions due to coupling and finite temperature: Even when performing only single-qubit gates, the finite temperature causes a partial excitation of higher states, which are then exchanged with the other qubit via the coupling and thus visible in the ORBIT data.

The performance of the gate on the device is verified with Quantum Process Tomography (QPT): We apply the CR gate preceded and followed by single-qubit gates to prepare and measure in the basis states, e.g. $S = \left(X_{\pi/2} \otimes Y_{\pi/2}\right) \circ \text{CR} \circ \left(X_{-\pi/2} \otimes Y_{\pi/2}\right)$ [41], and again collect these measurements into our data-set. We believe that the entangling gate breaks the degree of freedom between mis-classification and initial thermal distribution discussed before, hence we now perform another C_3 optimization, using the QPT data (256 sequences) and one ORBIT parameter per qubit (2×25 sequences) as the learning data. Parameter convergence is shown in the grey areas of Fig. 6, where temperature and confusion matrix values are adjusted closer to the true values.

Fig. 5b substantiates the claim that the entangling gate data allows for a more precise learning of the chip temperature, exhibiting a narrower valley at the true value. However, we are still not able to learn the T_1 and T_2^* parameters, since the sequences in QPT are too short to be sensitive.

G. Relaxation and dephasing

To demonstrate how a specialized measurement is formulated within $\mathbb{C}^3$ we determine the values of T_1 and T_2^*, using simple established sequences that are known to be sensitive to these parameters. The decay rate T_1 is determined by preparing the excited state of the qubit, followed by increasing wait times and then measuring the ground state population. We write the sequence as

$$S_{T1}^{(n)} = X_{\pi/2} \circ X_{\pi/2} \circ \mathcal{I}^n \tag{11}$$

where $X_{\pi/2}$ is our previously optimized $\pi/2$ gate and $\mathcal{I}^n$ signifies n repetitions of the identity gate $\mathcal{I}$. Similarly

$$S_{T2*}^{(n)} = X_{\pi/2} \circ \mathcal{I}^{n/2} \circ X_{\pi/2} \circ X_{\pi/2} \circ \mathcal{I}^{n/2} \circ X_{-\pi/2} \tag{12}$$

defines a Ramsey echo sequence, used to measure the dephasing rate T_2^*. We take 51 logarithmically spaced values of n between 100 and 10000 to capture the full decay curves.

Using this data-set we perform another C_3 optimization, freezing all model parameters learnt until now and varying only the values of T_1 and T_2^*. By doing so we manage to determine their values to within 1μs of the true values (Fig. 8). This procedure is the $\mathbb{C}^3$ equivalent of a common exponential decay fit to the data. However, with $\mathbb{C}^3$ one does not require prior knowledge on the expected structure of the experimental results, i.e. an exponential decay. Hence, when matching the data $\mathbb{C}^3$ also accounts for SPAM errors without the need to adjust the fitting function.

Fig. 5c shows the sensitivity of f_{LL} to the value of T_2^* of qubit B. The new data shows a clear improvement in the

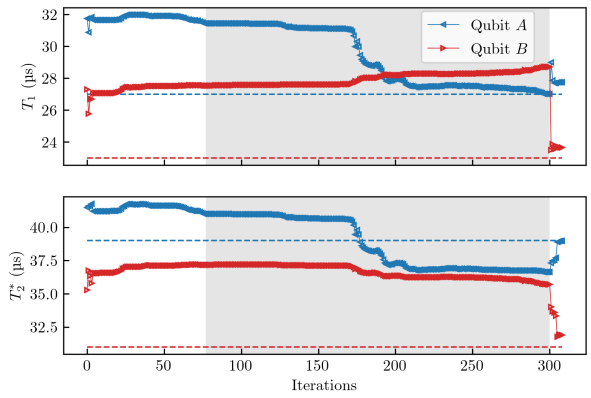


FIG. 8: C_3 learning of the relaxation (T_1) and dephasing (T_2^*) parameters. Blue, left (red, right) triangles indicate qubit A (B) parameters, respectively. The true values of the "real" model are indicated as dashed lines. Background sections represent different learning datasets: just ORBIT data (left white section), a mix of ORBIT and QPT data (center grey section), decay and Ramsey data (right white section). The decay rates are correctly identified only when specific data, sensitive to the decoherence effects, is used for learning, at which point they quickly converge to the real value.

accuracy of the value obtained and the minimum is better defined. For increased sensitivity one would require more decay data to learn from.

H. Sources of error

The Good Model allows us to break down which of the model properties are preventing higher gate fidelities. To this end, we investigate the Good Model for components limiting the performance of the CR gate by idealizing aspects of the model.

We investigate whether the Gaussian ansatz is limiting gate fidelities by further refining the optimal pulses using a piece-wise constant optimization with one pixel per AWG sample (as is done in [35]). Average infidelity improves only marginally from $f_{\text{av}}^{\text{DRAG}} = 3.8 \times 10^{-3}$ to $f_{\text{av}}^{\text{PWC}} = 3.6 \times 10^{-3}$, suggesting other factors are limiting fidelities.

To find out if performance is limited by decoherence effects, we re-optimize the CR gate while disabling Lindbladian dynamics. By open-loop optimization in this idealized coherent setting the error is decreased from 3.8×10^{-3} to 1.3×10^{-5}. Thus, the 100 ns CR gate considered here is coherence limited, as is the case in most experimental implementations [20, 42], making improvements in gate time essential [43].

IV. $\mathbb{C}^3$ IN-DEPTH

Following is a detailed description of the $\mathbb{C}^3$ tool-set.

A. Experiment modelling

To combine control and characterization, $\mathbb{C}^3$ provides a detailed simulation that endeavours to encompass all relevant practical considerations of the experiment such as signal processing, SPAM errors, control transfer functions and Markovian noise. The simulator is used as the basis of the open-loop optimal control optimization (C_1) and the model parameter optimization (C_3). In both cases it is desirable to use gradient-based optimization algorithms [44, 45]. However, it is extremely cumbersome to manually derive the full analytical gradients of the quantum dynamics, especially when it includes the properties described above. Instead, $\mathbb{C}^3$ uses a numerics framework [46] which allows for automatic differentiation [47], making the tool-set more flexible and easily extendable.

1. *Signal processing*

The simulation allows for the specification of control signals $\varepsilon(t)$ as either analytical functions or as direct, piecewise-constant AWG parameterization. Analytic parametrizations are sampled at the resolution of the waveform generator producing the envelope signal $\varepsilon_i = \varepsilon(t_i)$, representing voltages being applied to the control line, where the $\{t_i\}$ are the AWG sample times. The resulting signal will exhibit a *rise time* τ, due to the finite bandwidth of the control electronics. We model this by applying a convolution

$$\widetilde{\varepsilon}(t) = \int_{t_0}^{t_f} \text{interp}(\{\varepsilon_i\})(t) G\{t_f - t\}\, \mathrm{d}t \tag{13}$$

with

$$G(t) = \frac{1}{N} \exp\left\{ -\frac{(t - \tau/2)^2}{8\tau^2} \right\}, \tag{14}$$

modeling a Gaussian filter, and

$$\text{interp}(\{\varepsilon_i\})(t) = \{\varepsilon_i \,|\, t_i \leq t < t_{i+1}\} \tag{15}$$

interpolating the sampled signal to higher resolution for simulation. An I/Q-Mixer combines this envelope with a local oscillator signal of frequency ω_{lo} to

$$u(t) = I(t)\cos(\omega_{\text{lo}} t) - Q(t)\sin(\omega_{\text{lo}} t) \tag{16}$$

where the in-phase and quadrature components

$$\begin{aligned} I(t) &= \widetilde{\varepsilon}(t)\cos(\phi_{xy} - \omega_{\text{off}} t) \\ Q(t) &= \widetilde{\varepsilon}(t)\sin(\phi_{xy} - \omega_{\text{off}} t) \end{aligned} \tag{17}$$

are assigned by a control parameter ϕ_{xy}, and modulated to introduce a frequency offset ω_{off} on the drive. As noted in [48], in practice there will be additional errors during the mixing, which are not currently modelled. In transmitting this signal to the experiment, various distortions can occur, modeled by a response function φ, which also converts the field from line voltage to an amplitude $c(t) = \varphi[u(t)]$.

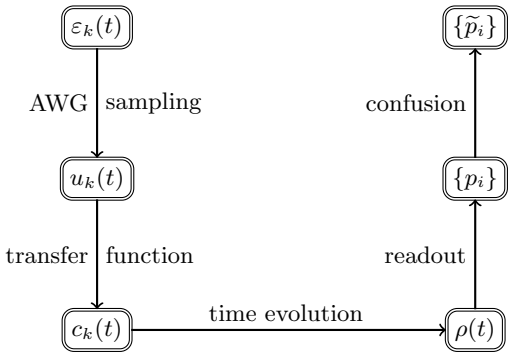


FIG. 9: The process of simulating experimental procedure for signal processing and readout. The k-th control function is specified by some function $\varepsilon_k(t)$ and specifies the line voltage $u_k(t)$ by an arbitrary waveform generator (AWG) with limited bandwidth. Electrical properties of the setup, such as impedances, are expressed as a line transfer function φ, resulting in a control field $c_k(t) = \varphi[u_k(t)]$, as in Eq. (18). After solving the equation of motion for the system, readout and misclassification are modelled by applying rescaling and transformations to the simulated populations $p_i = |\rho_{ii}|^2$, according to Eq. (23).

2. *Time evolution*

The system Hamiltonian is

$$H(t) = H_0 + \sum_k c_k(t) H_k \ , \tag{18}$$

with a drift H_0 and optional control Hamiltonians H_k. The dynamics of the system are described by the time-ordered propagator

$$U(t) = \mathcal{T} \exp\left\{ -\frac{\mathrm{i}}{\hbar} \int_{t_0}^{t} H(t')\mathrm{d}t' \right\} \ , \tag{19}$$

given by solving the time-dependent Schrödinger equation, and approximated numerically by $U(t) \simeq \prod_{i=N}^{0} U_i$. Here, $U_i = \exp\{-\frac{\mathrm{i}}{\hbar} H(t_i)\Delta t\}$, and the total time is divided into N slices of length Δt that are fine enough so that the Hamiltonian can be considered constant in the interval.

In application, we will rarely perform a single gate or pulse in isolation. Experiments such as randomized benchmarking or the various flavors of tomography involve long pulse sequences, that are inefficient to simulate as a whole. Instead, the $\mathbb{C}^3$ simulator computes each propagator G of a defined gateset $\mathcal{G}$ individually and compiles these matrix representations into sequences. This avoids the need to solve the equations of motions multiple times for the same exact pulses. As the propagators are calculated in the dressed laboratory frame (as opposed to the single-particle rotating frame), consecutive gates need to be adjusted to realign with the rotating frame of the drive signal, by applying a Z rotation with an angle of $(\omega_{\mathrm{lo}} + \omega_{\mathrm{off}})t_g$ [48].

To include open system effects, we apply the equivalent procedure to obtain the process matrix

$$\mathcal{E}(t) = \mathcal{T} \exp\left\{ \int_{t_0}^{t} \mathcal{L}(t')\mathrm{d}t' \right\} \tag{20}$$

by solving the master equation in Lindblad form,

$$\dot{\rho} = \mathcal{L}(\rho) = -i[H, \rho] + \sum_j L_j \rho L_j^\dagger - \frac{1}{2}\Big\{ L_j L_j^\dagger, \rho \Big\}, \tag{21}$$

where H is the Hamiltonian from Eq. (18), the L_js are Lindblad operators, and $\mathcal{L}$ is the generator in superoperator form [49]. The evolution of a state is obtained by applying the propagator as $\rho_f = U(t_g)\rho_i U^\dagger(t_g)$ for coherent evolution or $\rho_f = \mathcal{E}(t_g)[\rho_i]$ for incoherent evolution.

3. *Initialization and Read-out*

Given the temperature T of the device, the system is initialized in a mixed state

$$\rho_{\mathrm{init}} = \sum_k \frac{1}{Z} |\phi_k\rangle\langle\phi_k| \exp\{-E_k/k_B T\} \tag{22}$$

where $\{|\phi_k\rangle\}$ is the eigenbasis of H_0 and the normalization is given by the canonical partition function $Z = \sum_k \exp\{-E_k/k_B T\}$.

We simulate readout by post-processing the final state ρ_f: From the density matrix, represented in the dressed basis, we obtain a vector of populations $\vec{p} = (p_k)$ by taking the absolute square of the diagonal. This is consistent with a slow (or weak) readout scheme in experiment. Measurement and classification errors are modeled with a mis-classification (confusion) matrix $(p_{i\to j})_{ij}$ [50] such that the measured populations are

$$\widetilde{p}_j = \sum_i p_{i\to j} p_i \ . \tag{23}$$

To simulate an experimental measurement with an average of l repetitions, we draw from a Multinomial distribution of l trails and with probabilities $\widetilde{p}_j$.

B. Optimizations

Open and closed loop optimal control as well as model learning require performing optimization processes.

1. *Open-loop Model-based Control: C_1*

In the typical setting of open-loop optimal control [8, 9], given a model of a system, we search for the optimal control pulses to drive the system to a desired state or generate a certain gate. Pulses are parametrized by an analytic ansatz (e.g. Gaussian pulse with DRAG correction [14] to remove Fourier components coupling to leakage levels), or by direct AWG samples. Constraints may be imposed to conform with experimental feasibility,

such as power and bandwidth limitations. The goal function to be minimized is selected depending on the specified optimal control task, e.g. state infidelity for state transfer problems, or unitary trace infidelity for quantum gates [9, 29]. We suggest the use of average gate infidelity as the goal function, as it is experimentally accessible by performing RB or QPT, allowing comparison of performance in simulation and experiment.

Formally, the controls are parameterized as a vector of real numbers α. Given a goal function $f_1(\alpha)$, we search for $\min_\alpha f_1(\alpha)$. Optimal control methods such as GRAPE [11], Krotov [13, 51–53], and GOAT [10] have been devised to determine the gradient $\partial_\alpha f_1(\alpha)$ in order to facilitate convergence. These methods require a specific formulation of the problem and the analytical calculation of the gradient any additional elements in the model. Instead, in $\mathbb{C}^3$, automatic differentiation allows to systematically account for any model feature, including, for example, line response functions or SPAM error.

The disadvantage of gradient-based algorithms is their propensity to get trapped in local minima. The severity of the problem is reduced by using a hierarchy of progressively more complex control ansätze. If this is insufficient, a short preliminary gradient-free search to find the convergence basin most often resolves the problem.

2. *Closed-loop Model-free Calibration: C_2*

In calibration, a given pulse is optimized to improve a figure of merit $f_2(\alpha)$, computed from experimental measurement results. In addition to gradient-free optimization algorithms, there are methods to approximate the gradients (e.g. [54]), however, such approaches are generally less efficient than gradient-free algorithms [44, 55] as they require a high number of evaluations [56]. If the initial point of the optimization is given by C_1, this implements the already established AdHOC [18] method. During calibration, sets of control parameters α_j are sent to the experimental setup, alongside instructions of how to evaluate the current controls. For evaluating gate-sets, we suggest the ORBIT figure of merit, as it naturally performs a twirling of all sources of error, providing a single number to optimize. However, protocols tailored to specific needs can also be used, e.g. to obtain a desired conditional phase [5]. C_2 then optimizes the control parameters α_j to minimize a figure of merit.

While specialized measurements provide a straightforward way to fine-tune controls related to specific device properties, they do not generally account for interdependencies. For more complex setups with many parameters, such calibrations cannot be done without extraordinary effort [57]. In contrast, $\mathbb{C}^3$ employs modern gradient-free optimization algorithms, such as CMA-ES (see Supplement S2 for further discussion), capable of optimizing dozens of parameters simultaneously, automating the task.

3. *Model Learning: C_3*

Extracting the model from a data-set $\mathcal{D}$ can be thought of formally as analogous to the C_1 optimization task, where one varies the *model* parameters instead of the control parameters. For each measurement outcome m_k in the data-set,

$$\mathcal{D} = \{S_k \mapsto m_k\}_j \,, \tag{24}$$

the corresponding gate or pulse sequences $S_k(\alpha_j)$ with control parameters α_j are used to simulate the model's prediction $\widetilde{m}_{j,k} = \widetilde{m}(S_k(\alpha_j), \beta)$. The model learning goal function

$$f_3(\mathcal{D}|\beta) = f_3(\{\widetilde{m}_k(\beta)\}, \{m_k\}) \tag{25}$$

quantifies the quality of the match between the data-set and the simulation of a system with parameters β. In this paper, we use a rescaled log-likelihood

$$f_{\mathrm{LL}}(\mathcal{D}|\beta) = \frac{1}{2K}\sum_{k=1}^{K}\left(\left(\frac{m_k - \widetilde{m}_k}{\widetilde{\sigma}_k}\right)^2 - 1\right) , \tag{26}$$

where the $\widetilde{\sigma}_k$ is the standard deviation of a binomial distribution with mean $\widetilde{m}_k$, resulting in a variation of the Mahalanobis distance [58]. This function is stricly correct under the gaussian assumption and a two-level readout. See Supplement S4 for the extension for a multiple outcome readout. The measurement process on any physical device is noisy, i.e. each m_k is an estimate of a true underlying μ_k. Therefore, a realistic data-set $\mathcal{D}$ cannot be matched exactly by a deterministic simulation. The function f_{LL} is designed such that, for n data points, its expectation value is 0 when the model predicts the means μ_k correctly, and $\frac{1}{2}n^2$ if the distance is $\mu_k - \widetilde{m}_k = n\sigma_k$ for all ks, according to Supplement Eq. (5). This provides a more intuitive measure of model match than the abstract value of f_{LL}, i.e. it allows to make a statement like "the model differs from the experiment by approximately n standard deviations".

Due to the complexity of the physical systems, a potentially high number of interdependent parameters and complex features of the landscape, it is difficult for the optimization to converge to the global optimum. Therefore, we take the tried-and-tested experimental approach of starting with a simple model and iteratively refining it. One may modify the model and repeat the C_3 fit, optionally retaining the optimized parameters which are shared by the previous and new model. Alternatively, one may collect additional data and repeat the optimization on the same model. We emphasize that at each of these steps the physicists' insights are required to evaluate the optimization's results, extend or discard models and decide whether collecting additional data is required. Furthermore, employing a gradient-based algorithm can, depending on the initial point, result in a local minimum. The optimizations presented here were successful when starting with a gradient-free CMA-ES search, known to

be robust against local minima, switching over to the faster converging gradient-based L-BFGS method when a promising parameter region is identified. However, further research is required to find the best optimization strategy.

We note that in $\mathbb{C}^3$ a model takes explicit values for all its parameters, and is *not* represented as a high-dimensional distribution over model parameter space. This choice is driven by classical computation-load considerations: Because the $\mathbb{C}^3$ model is highly detailed, and, as consequence, associated simulations are non-trivial, we believe a full Bayesian approach to any of $\mathbb{C}^3$ optimizations is not computationally viable at this time.

C. Model analysis

Both during and after the learning process, it is beneficial to interrogate the model to estimate its properties and their impact on the system behaviour. As part of the $\mathbb{C}^3$ tool-set one can perform sensitivity analysis for system parameters: Sweeping a single parameter, e.g. qubit frequency, across the range of interest, while keeping other parameters at their current best value, evaluating the model match score at each point, exemplified in Fig. 4. The result is a 1-D cut through the optimization landscape that may exhibit a well-defined minimum, multiple extrema indicating a difficult optimization, or even appear flat in the case when a parameter does not affect the behaviour of the current experiment. This landscape depends on both the selected model and data it is compared to. Depending on the ruggedness of the sensitivity, one might choose to utilize a gradient-based algorithm from the start or to first perform a gradient-free exploratory search to avoid local minima. In the case of a flat sensitivity, one may choose two courses of action: If the parameter is of little interest for successive experiments, it may be removed or set to a convenient value within the flat range; otherwise, one needs to design an experiment producing additional data that is sensitive to the parameter. The physicists' knowledge of common experimental practices (e.g. Rabi, Ramsey, Hahn echo) and intuition guides the decision for the experiment design. When suspecting correlations between parameters, cuts in single dimensions are not enough and higher dimensional sweeps are necessary. After a successful learning process, the sensitivity analysis gives an estimate of the precision to which each parameter has been determined.

Furthermore, the simulation allows insight into the behaviour of the system. Using well established methods such as time-resolved state and process tomography, it is possible to identify coherent errors, such as leakage out of the computational subspace, over-rotations, and the effects of noise. A Good Model also provides the basis for an error budget, as it contains the same limitations as the experiment it accurately predicts. The model can be used for extrapolation by idealizing certain aspects suspected as causes of infidelity (e.g. T_1 setting to infinity), and re-deriving control pulses using a C_1 optimization. The respective gain in fidelity gives an estimate of the error that this aspect is responsible for, suggesting areas of improvement for future devices.

V. DISCUSSION AND OUTLOOK

In conclusion, we have described $\mathbb{C}^3$, an integrated methodology to improve quantum device performance that combines characterization, calibration and control. We have detailed its approach and implementation, demonstrating, on a synthetic QPU device, the individual methods and how they are synthesized into a more integrated concept. Analyzing single-qubit calibration data we successfully extracted an accurate model of the device, including realistic experimental considerations: line transfer functions, limitations of control electronics, readout error and finite operating temperature. From this model we were able to derive a working high-fidelity two-qubit gate, without requiring any further calibration.

This approach represents a holistic theoretical take on the experimental workflow of a complex quantum computing experiment, that takes into account interactions between different tasks of an experimental lab. $\mathbb{C}^3$ provides a path to achieve, starting from an incomplete understanding of the system, both high-fidelity pulses and an accurate model. It integrates the tasks of open-loop control (that would require an already accurate model) and of calibration (that would require an experiment-specific fine-tuning procedure). Most notably, it provides the tools to reflect on the experiment outcome and gate performance, improving the model description of the system and providing insight into its behaviour. $\mathbb{C}^3$ is not a "black-box" experiment controller that replaces physicists or engineers - rather, it reduces tedious tasks allowing for interaction with the quantum device on a more structural level. Instead of simply producing high-fidelity operations, $\mathbb{C}^3$ provides meaningful output in the form of a Good Model of the system, and other insights such as an error budget and a sensitivity analysis. In this sense, $\mathbb{C}^3$ is not to be confused with any single optimal control or benchmarking technique, as it includes results from decades of research in these fields aimed at making controls that allow to actually reach high fidelities efficiently [8, 45], unifying them into one framework.

We expect that the application of $\mathbb{C}^3$ will first benefit scalable implementations of quantum processors based on manufactured solid-state systems, such as superconducting and semiconducting qubits. There, the dependence of model parameters on fabrication means that many elements of the model are in fact uncertain. On the other hand, other scalable implementations of quantum computers contain such elements directly in their quantum description: Ion trap gates involve degrees of freedom of the trap, impurity spins depend on their detailed position etc. – thus we expect that $\mathbb{C}^3$ will also play a key role in those types of systems.

We have introduced the initial version of the $\mathbb{C}^3$ tool-

set. In the near future we intend on extending it to the generation of robust controls, automatic experiment design, multi-parameter sensitivity analysis, active model learning, and more. The simulator will be enhanced to include non-Markovian noise, a detailed simulation of the readout process, echoes on control lines and all other phenomena needed to produce a Good Model of real-world systems. We also envision that $\mathbb{C}^3$ can be integrated in existing quantum computing software stacks which would allow users to study custom pulse schedules [59, 60] and perform model learning using data gathered from quantum computers, for example, with Qiskit Ignis [61]. Experimental application of $\mathbb{C}^3$ is ongoing (e.g. [35]).

It is our hope that $\mathbb{C}^3$ will not only provide insights into, and assist optimization of, current experiments, but also help guide the design of next-generation quantum devices, be it manually or by integration into automatic hardware design frameworks [62, 63].

ACKNOWLEDGEMENTS

This work was supported by the European Commission through the Marie Curie ETN QuSCo (Grant Nr. 765267) and the OpenSuperQ project (Grant Nr. 820363), by the Intelligence Advanced Research Projects Activity (IARPA) through the LogiQ (Grant Nr. W911NF-16-1-0114) and by the Germany Ministry of Science and Education (BMBF) through project VERTICONS (Grant Nr. 13N14872).

[1] P. Jurcevic, A. Javadi-Abhari, L. S. Bishop, I. Lauer, D. F. Bogorin, M. Brink, L. Capelluto, O. Günlük, T. Itoko, N. Kanazawa, *et al.*, *Demonstration of quantum volume 64 on a superconducting quantum computing system,* (2020), arXiv:2008.08571.

[2] F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, R. Biswas, S. Boixo, F. G. Brandao, D. A. Buell, *et al.*, *Quantum supremacy using a programmable superconducting processor,* Nature **574**, 505 (2019).

[3] A. W. Cross, L. S. Bishop, S. Sheldon, P. D. Nation, and J. M. Gambetta, *Validating quantum computers using randomized model circuits,* Phys. Rev. A **100**, 032328 (2019).

[4] R. Barends, J. Kelly, A. Megrant, A. Veitia, D. Sank, E. Jeffrey, T. C. White, J. Mutus, A. G. Fowler, B. Campbell, Y. Chen, Z. Chen, B. Chiaro, A. Dunsworth, C. Neill, P. O'Malley, P. Roushan, A. Vainsencher, J. Wenner, A. N. Korotkov, A. N. Cleland, and J. M. Martinis, *Superconducting quantum circuits at the surface code threshold for fault tolerance,* Nature **508**, 500 (2014).

[5] M. C. Collodo, J. Herrmann, N. Lacroix, C. K. Andersen, A. Remm, S. Lazar, J.-C. Besse, T. Walter, A. Wallraff, and C. Eichler, *Implementation of conditional-phase gates based on tunable zz-interactions,* (2020), arXiv:2005.08863.

[6] M. Ganzhorn, G. Salis, D. J. Egger, A. Fuhrer, M. Mergenthaler, C. Müller, P. Müller, S. Paredes, M. Pechal, M. Werninghaus, and S. Filipp, *Benchmarking the noise sensitivity of different parametric two-qubit gates in a single superconducting quantum computing platform,* (2020), arXiv:2005.05696.

[7] G. E. P. Box, *Science and statistics,* J. Am. Stat. Assoc. **71**, 791 (1976).

[8] S. J. Glaser, U. Boscain, T. Calarco, C. P. Koch, W. Köckenberger, R. Kosloff, I. Kuprov, B. Luy, S. Schirmer, T. Schulte-Herbrüggen, D. Sugny, and F. K. Wilhelm, *Training schrödinger's cat: quantum optimal control,* Eur. Phys. J. D **69**, 279 (2015).

[9] S. Machnes, U. Sander, S. J. Glaser, P. de Fouquières, A. Gruslys, S. Schirmer, and T. Schulte-Herbrüggen, *Comparing, optimizing, and benchmarking quantum-control algorithms in a unifying programming framework,* Phys. Rev. A **84**, 022305 (2011).

[10] S. Machnes, E. Assémat, D. Tannor, and F. K. Wilhelm, *Tunable, flexible, and efficient optimization of control pulses for practical qubits,* Phys. Rev. Lett. **120**, 150401 (2018).

[11] N. Khaneja, T. Reiss, C. Kehlet, T. Schulte-Herbrüggen, and S. J. Glaser, *Optimal control of coupled spin dynamics: design of nmr pulse sequences by gradient ascent algorithms,* J. Magn. Reson. **172**, 296 (2005).

[12] P. Doria, T. Calarco, and S. Montangero, *Optimal control technique for many-body quantum dynamics,* Phys. Rev. Lett. **106**, 190501 (2011).

[13] R. Kosloff, S. Rice, P. Gaspard, S. Tersigni, and D. Tannor, *Wavepacket dancing: Achieving chemical selectivity by shaping light pulses,* Chem. Phys. **139**, 201 (1989).

[14] F. Motzoi, J. M. Gambetta, P. Rebentrost, and F. K. Wilhelm, *Simple pulses for elimination of leakage in weakly nonlinear qubits,* Phys. Rev. Lett. **103**, 110501 (2009).

[15] R. Schutjens, F. A. Dagga, D. J. Egger, and F. K. Wilhelm, *Single-qubit gates in frequency-crowded transmon systems,* Phys. Rev. A **88**, 052330 (2013).

[16] U. Dorner, T. Calarco, P. Zoller, A. Browaeys, and P. Grangier, *Quantum logic via optimal control in holographic dipole traps,* J. Opt. B: Quantum Semiclassical Opt. **7**, S341 (2005).

[17] D. J. Egger and F. K. Wilhelm, *Optimized controlled-Z gates for two superconducting qubits coupled through a resonator,* Supercond. Sci. Technol. **27**, 014001 (2013).

[18] D. J. Egger and F. K. Wilhelm, *Adaptive hybrid optimal quantum control for imprecisely characterized systems,* Phys. Rev. Lett. **112**, 240503 (2014).

[19] J. Kelly, R. Barends, B. Campbell, Y. Chen, Z. Chen, B. Chiaro, A. Dunsworth, A. G. Fowler, I.-C. Hoi, E. Jeffrey, A. Megrant, J. Mutus, C. Neill, P. J. J. O'Malley, C. Quintana, P. Roushan, D. Sank, A. Vainsencher, J. Wenner, T. C. White, A. N. Cleland, and J. M. Martinis, *Optimal quantum control using randomized benchmarking,* Phys. Rev. Lett. **112**, 240504 (2014).

[20] S. Sheldon, E. Magesan, J. M. Chow, and J. M. Gambetta, *Procedure for systematically tuning up cross-talk in the cross-resonance gate,* Phys. Rev. A **93**, 060302 (2016).

[21] E. Knill, D. Leibfried, R. Reichle, J. Britton, R. B. Blakestad, J. D. Jost, C. Langer, R. Ozeri, S. Seidelin, and D. J. Wineland, *Randomized benchmarking of quantum gates,* Phys. Rev. A **77**, 012307 (2008).

[22] M. P. da Silva, O. Landon-Cardinal, and D. Poulin, *Practical characterization of quantum devices without tomography,* Phys. Rev. Lett. **107**, 210404 (2011).

[23] B. P. Lanyon, C. Maier, M. Holzäpfel, T. Baumgratz, C. Hempel, P. Jurcevic, I. Dhand, A. S. Buyskikh, A. J. Daley, M. Cramer, M. B. Plenio, R. Blatt, and C. F. Roos, *Efficient tomography of a quantum many-body system,* Nat. Phys. **13**, 1158 (2017).

[24] A. Erhard, J. J. Wallman, L. Postler, M. Meth, R. Stricker, E. A. Martinez, P. Schindler, T. Monz, J. Emerson, and R. Blatt, *Characterizing large-scale quantum computers via cycle benchmarking,* Nat. Commun. **10**, 1 (2019).

[25] G. Lindblad, *On the generators of quantum dynamical semigroups,* Commun. Math. Phys. **48**, 119 (1976).

[26] V. Gorini, A. Kossakowski, and E. C. G. Sudarshan, *Completely positive dynamical semigroups of n-level systems,* J. Math. Phys. **17**, 821 (1976).

[27] B. Girod, R. Rabenstein, and A. Stenger, *Signals and systems* (Wiley, 2001).

[28] Z. Chen, J. Kelly, C. Quintana, R. Barends, B. Campbell, Y. Chen, B. Chiaro, A. Dunsworth, A. G. Fowler, E. Lucero, E. Jeffrey, A. Megrant, J. Mutus, M. Neeley, C. Neill, P. J. J. O'Malley, P. Roushan, D. Sank, A. Vainsencher, J. Wenner, T. C. White, A. N. Korotkov, and J. M. Martinis, *Measuring and suppressing quantum state leakage in a superconducting qubit,* Phys. Rev. Lett. **116**, 020501 (2016).

[29] E. Magesan, R. Blume-Kohout, and J. Emerson, *Gate fidelity fluctuations and quantum process invariants,* Phys. Rev. A **84**, 012309 (2011).

[30] J. Nocedal, *Updating quasi-newton matrices with limited storage,* Math. Comput. **35**, 773 (1980).

[31] E. Knill, D. Leibfried, R. Reichle, J. Britton, R. B. Blakestad, J. D. Jost, C. Langer, R. Ozeri, S. Seidelin, and D. J. Wineland, *Randomized benchmarking of quantum gates,* Phys. Rev. A **77**, 012307 (2008).

[32] J. M. Chow, J. M. Gambetta, L. Tornberg, J. Koch, L. Bishop, A. A. Houck, B. R. Johnson, L. Frunzio, S. M. Girvin, and R. J. Schoelkopf, *Randomized Benchmarking and Process Tomography for Gate Errors in a Solid-State Qubit,* Phys. Rev. Lett. **102**, 090502 (2009).

[33] E. Magesan, J. M. Gambetta, and J. Emerson, *Scalable and robust randomized benchmarking of quantum processes,* Phys. Rev. Lett. **106**, 180504 (2011).

[34] E. Magesan, J. M. Gambetta, and J. Emerson, *Characterizing quantum gates via randomized benchmarking,* Phys. Rev. A **85**, 042311 (2012).

[35] M. Werninghaus, D. J. Egger, F. Roy, S. Machnes, F. K. Wilhelm, and S. Filipp, *Leakage reduction in fast superconducting qubit gates via optimal control,* (2020), arXiv:2003.05952.

[36] N. Hansen, S. D. Müller, and P. Koumoutsakos, *Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (cma-es),* Evol. Comput. **11**, 1 (2003).

[37] A. Blais, R.-S. Huang, A. Wallraff, S. M. Girvin, and R. J. Schoelkopf, *Cavity quantum electrodynamics for superconducting electrical circuits: An architecture for quantum computation,* Phys. Rev. A **69**, 062320 (2004).

[38] J. J. Wallman and S. T. Flammia, *Randomized benchmarking with confidence,* New J. Phys. **16**, 103032 (2014).

[39] C. Rigetti and M. Devoret, *Fully microwave-tunable universal gates in superconducting qubits with linear couplings and fixed transition frequencies,* Phys. Rev. B **81**, 134507 (2010).

[40] J. M. Chow, A. D. Córcoles, J. M. Gambetta, C. Rigetti, B. R. Johnson, J. A. Smolin, J. R. Rozen, G. A. Keefe, M. B. Rothwell, M. B. Ketchen, and M. Steffen, *Simple All-Microwave Entangling Gate for Fixed-Frequency Superconducting Qubits,* Phys. Rev. Lett. **107**, 080502 (2011).

[41] T. Alexander, N. Kanazawa, D. J. Egger, L. Capelluto, C. J. Wood, A. Javadi-Abhari, and D. McKay, *Qiskit pulse: Programming quantum computers through the cloud with pulses,* (2020), arXiv:2004.06755.

[42] A. D. Córcoles, J. M. Gambetta, J. M. Chow, J. A. Smolin, M. Ware, J. Strand, B. L. T. Plourde, and M. Steffen, *Process verification of two-qubit quantum gates by randomized benchmarking,* Phys. Rev. A **87**, 030301 (2013).

[43] S. Kirchhoff, T. Keßler, P. J. Liebermann, E. Assémat, S. Machnes, F. Motzoi, and F. K. Wilhelm, *Optimized cross-resonance gate for coupled transmon systems,* Phys. Rev. A **97**, 042348 (2018).

[44] S. Machnes, E. Assémat, D. Tannor, and F. K. Wilhelm, *Tunable, flexible, and efficient optimization of control pulses for practical qubits,* Phys. Rev. Lett. **120**, 150401 (2018).

[45] F. K. Wilhelm, S. Kirchhoff, S. Machnes, N. Wittler, and D. Sugny, *An introduction into optimal control for quantum technologies,* (2020), arXiv:2003.10132.

[46] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, and M. Isard, *TensorFlow: Large-scale machine learning on heterogeneous systems,* (2015), software available from tensorflow.org.

[47] G. Kedem, *Automatic differentiation of computer programs,* ACM Transactions on Mathematical Software (TOMS) **6**, 150 (1980).

[48] D. C. McKay, C. J. Wood, S. Sheldon, J. M. Chow, and J. M. Gambetta, *Efficient z gates for quantum computing,* Phys. Rev. A **96**, 022330 (2017).

[49] D. A. Lidar, *Lecture notes on the theory of open quantum systems,* (2019), arXiv:1902.00967.

[50] S. V. Stehman, *Selecting and interpreting measures of thematic classification accuracy,* Remote Sens. Environ. **62**, 77 (1997).

[51] A. P. Peirce, Mohammed, A. Dahleh, and H. Rabitz, *Optimal control of quantum-mechanical systems: Existence, numerical approximation, and applications,* Phys. Rev. A **37**, 4950 (1988).

[52] D. M. Reich, M. Ndong, and C. P. Koch, *Monotonically convergent optimization in quantum control using krotov's method,* The Journal of Chemical Physics **136**, 104103 (2012).

[53] M. H. Goerz, D. Basilewitsch, F. Gago-Encinas, M. G. Krauss, K. P. Horn, D. M. Reich, and C. P. Koch, *Krotov: A Python implementation of Krotov's method for quantum optimal control,* SciPost Phys. **7**, 80 (2019).

[54] Z. Leng, P. Mundada, S. Ghadimi, and A. Houck, *Robust and efficient algorithms for high-dimensional black-box quantum optimization,* (2019), arXiv:1910.03591.

[55] J. Nocedal and S. Wright, *Numerical optimization* (Springer Science & Business Media, 2006).

[56] G. Feng, F. H. Cho, H. Katiyar, J. Li, D. Lu, J. Baugh, and R. Laflamme, *Gradient-based closed-loop quantum optimal control in a solid-state two-qubit system,* Phys. Rev. A **98**, 052341 (2018).

[57] 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.

[58] P. C. Mahalanobis (National Institute of Science of India, 1936).

[59] D. C. McKay, T. Alexander, L. Bello, M. J. Biercuk, L. Bishop, J. Chen, J. M. Chow, A. D. Córcoles, D. Egger, S. Filipp, J. Gomez, M. Hush, A. Javadi-Abhari, D. Moreda, P. Nation, B. Paulovicks, E. Winston, C. J. Wood, J. Wootton, and J. M. Gambetta, *Qiskit backend specifications for openqasm and openpulse experiments,* (2018), arXiv:1809.03452.

[60] T. Alexander, N. Kanazawa, D. J. Egger, L. Capelluto, C. J. Wood, A. Javadi-Abhari, and D. C. McKay, *Qiskit pulse: programming quantum computers through the cloud with pulses,* Quantum Science and Technology **5**, 044006 (2020).

[61] G. Aleksandrowicz *et al.*, Qiskit: An Open-source Framework for Quantum Computing, (2019).

[62] A. A. Melnikov, H. P. Nautrup, M. Krenn, V. Dunjko, M. Tiersch, A. Zeilinger, and H. J. Briegel, *Active learning machine learns to create new quantum experiments,* Proceedings of the National Academy of Sciences **115**, 1221 (2018).

[63] B. P. MacLeod, F. G. L. Parlane, T. D. Morrissey, F. Häse, L. M. Roch, K. E. Dettelbach, R. Moreira, L. P. E. Yunker, M. B. Rooney, J. R. Deeth, V. Lai, G. J. Ng, H. Situ, R. H. Zhang, M. S. Elliott, T. H. Haley, D. J. Dvorak, A. Aspuru-Guzik, J. E. Hein, and C. P. Berlinguette, *Self-driving laboratory for accelerated discovery of thin-film materials,* Sci. Adv. **6** (2020), 10.1126/sciadv.aaz8867.

Supplementary material for: An integrated tool-set for Control, Calibration and Characterization of quantum devices applied to superconducting qubits

Nicolas Wittler,[1,2,*] Federico Roy,[1,3,*] Kevin Pack,[1,2] Max Werninghaus,[1,3] Anurag Saha Roy,[4] Daniel J. Egger,[3] Stefan Filipp,[3,5] Frank K. Wilhelm,[1,2] and Shai Machnes[1,†]

[1] *Theoretical Physics, Saarland University, 66123 Saarbrücken, Germany*

[2] *Peter Grünberg Institut – Quantum Computing Analytics (PGI 12), Forschungszentrum Jülich, D-52425 Jülich, Germany*

[3] *IBM Quantum, IBM Research GmbH, Zurich Research Laboratory, Säumerstrasse 4, 8803 Rüschlikon, Switzerland*

[4] *Citizen scientist*

[5] *Department of Physics, Technical University of Munich, 85748 Garching, Germany*

S1. PAST ATTEMPTS AT COMBINING CHARACTERIZATION AND CONTROL

Considered individually, each of $\mathbb{C}^3$'s three main pillars has received extensive attention. It would be presumptuous of us to even attempt a survey of all three. Therefore, we shall limit ourselves to a very limited look at a subset we have subjectively found informative to our own work. Discussion of calibration algorithms is deferred to a forthcoming publication. A review of optimal control algorithms is largely made redundant by the use of automatic differentiation, as implemented in TensorFlow [1] and used by [2], C_1 and C_3. We therefore focus this limited survey to the issue of system characterization. Characterization can be broadly broken down into model-specific methods, generic model-based approaches, and model-free approaches

The standard approach at addressing the lack of a Good Model, as defined above, is to perform a long list of model-specific characterization experiments, each designed to measure a different parameter of the model: measure parameters of the readout resonator using frequency sweeps; qubit frequency measurements and T_1 require Rabi experiments [3] (and with some extra effort the higher levels can be extracted); Ramsey [4] and Hahn echo measurements [5] provides T_2 data (under the Markovian assumption, which is known to be an over-simplification [6, 7]); measuring the control line response functions [8–12], the noise spectra [7, 13, 14], continuous drifts in system parameters [15–18], and discontinuous jumps in parameters such as T_1 [19, 20]; state Preparation and Measurement Errors (SPAM) can be extracted from Randomized Benchmarking (RB; e.g. [21]) or dedicated procedures, such as [22]; qubit cross-talk can be measured by the method described in [23, 24] and many more. Model specific methods also exist for learning spin chain, lattice Hamiltonians and other multi-particle systems with a predefined network topology under limited access [25–29].

Some of the more general-purpose methods, designed, for example, to identify all Hamiltonian elements, are often classified as *system identification* protocols, e.g. [30–35]. Bayesian approaches, e.g. [36–45] can greatly reduce the required experimental sampling count. Such methods are very useful but are also limited to the model and parameters for which they were designed. When attempting to find a Good Model, one usually must resort to a complex, highly detailed model, for which model-specific methods are of limited applicability.

Generic model-based approaches provide tools to determine model parameters which may be applied to any system model, including those going beyond simple Hamiltonian description. Such approaches fall under the machine-learning domain known as *experiment design* [46] and its Bayesian version [47], and have been used in the quantum realm, e.g. [48–51]. In Bayesian approaches, one must maintain a probability distribution function (PDF) over possible model space, and update it (from prior to posterior) in an online fashion, to reflect newly acquired experimental data. Maintaining good approximations of such PDFs are computationally demanding, especially when the models in question have dozens of parameters (PDF dimensions). The alternative approach of using MCMC (Markov Chain Monte Carlo) algorithms is even more unwieldy. The experiment design is a form of a model-based optimal control task where one attempts to maximize a utility function, such as the information gain (i.e. minimize the posterior

* These two authors contributed equally
† shai.machnes@gmail.com

model PDF Shannon entropy). Unfortunately, determining information gain is also computationally expensive. Thus, inserting such computations during an experiment will result in the quantum device spending almost all of its time waiting for the classical computation - an undesirable balance.

Methods for Hamiltonian identification have been developed which do not make many assumptions as to its structure. Examples include methods based on compressed sensing [43, 52, 53], extraction of detailed time traces [54, 55] and others. However, as system model extends beyond Hamiltonians and Lindblandians, to non-Markovian noise spectra, line transfer functions and other non-trivial effects, it is unclear to what degree these methods can be put in practice. Further, treating Hamiltonians without assuming structure not only leaves a lot of a-priori information about the system model unused, it does not make use of the model's ability to reduce the growth in dimensionality of parameter space from exponential to polynomial.

There are countless additional approaches to the task of characterization, of which we shall mention a small subset: [56] propose the use of random control signals and SGD (Stochastic Gradient Descent) to minimize the discrepancy between observed and simulated behavior. However, it is not clear how such an approach would, for example, prevent a random control signal from moving a transmon out of the computational subspace, or how would it be "lucky" enough to operate in parameter ranges where certain phenomena become significant (e.g. generating a cross-resonant effect requires fine-tuning of control frequencies which may not occur by random). Neural Networks have been used for both system identification and optimal control [57–59] with some success. However, it is not a-priori clear why such a representation of quantum dynamics or their outputs is preferable to physics-based models.

Intertwining of control and characterization has been raised in a more general context of control theory [60–63]. But, to our knowledge, $\mathbb{C}^3$ is unique, at least in the context of quantum technology, in its holistic approach tying together model-based optimal control and characterization with model-free control calibration.

There are some works in literature which combine two of the three steps: Ad-Hoc [64] calls for optimal control followed by calibration. A combination of model-based gradient calculations and experimental calibrations is proposed by [65, 66], but the data gathered is not used to improve the system model.

S2. CHOICE OF GRADIENT-FREE ALGORITHMS

Naturally, using an automated approach to solve the gradient-free optimization problem of calibration seems promising, and many experiments are currently using the *Nelder-Mead algorithm* [67, 68] for their calibration tasks. As an example, Fig. S1 shows an underlying parameter landscape of an optimization, obtained by simulating DRAG corrected pulses. Calibration can be rather difficult, as depending on the starting position the optimizer, algorithms will have to overcome local minima and deal with intrinsic noise. It is further noteworthy that this landscape is rather unique to the used parametrization, the chosen goal function and ultimately the properties of the physical system.

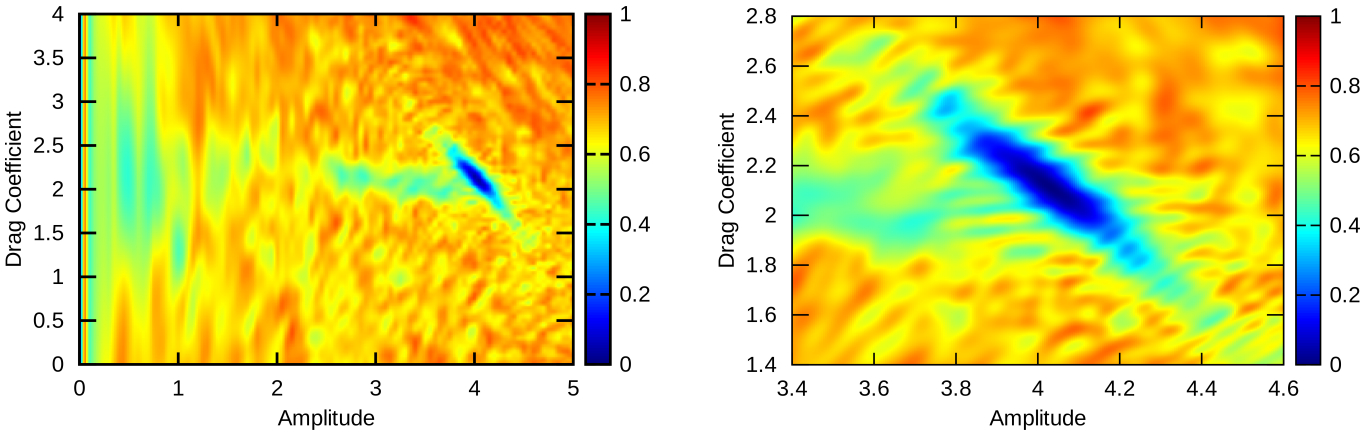


FIG. S1: A 2D cut through the landscape of a calibration goal function. The system chosen is a multi-level qubit, with the control sequence being a single length RB-sequence. Control parameters α_j were the Gaussian pulse amplitude, DRAG coefficient and frequency offset of a DRAG-corrected pulse. The RB-sequence was chosen to implement the identity operation, ideally leaving the qubit in the ground-state. The infidelity (seen as the color ranging from 0 to 1) is defined as the overlap of the final state with the system's ground state. a) The landscape of the simulated system, as a cut through the plane of pulse amplitude and DRAG coefficient, for a fixed RB-sequence. Multiple local minima can be observed. b) A higher resolution plot of the same landscape. Further local minima can be observed in the neighborhood of the global minimum.

Research of gradient-free optimizers is a vast and active field, with hundreds of published black-box optimizers, including Evolutionary Strategies (ES), Particle Swarm Optimization (PSO), Differential Evolution (DE), Random Search, Simultaneous Pertubation Stochastics Approximation (SPSA), Nelder-Mead Method, Bayesian Optimization, and more. From a preliminary investigation, we recommend evolutionary strategies such as CMA-ES, which is currently the default in C_2. This algorithm performed well in most cases, exhibited good robustness to noise, can handle local extrema and requires relatively few evaluations. Similar conclusions have been reached independently by [69]. We make no claim as to the optimality of the optimizer chosen, and defer a more detailed discussion of the subject to future publications.

S3. OPEN-SOURCE IMPLEMENTATION

$\mathbb{C}^3$ is implemented as an open source project available at https://q-optimize.org under the Apache 2.0 license. The software is written in Python to interface conveniently with common experiment controllers, and has already been used in tandem with PycQED [70], Labber [71] and LabView [72]. The interface can occur at various levels of abstraction, from sharing control parameters to sampled waveform values. A modular design allows for Hamiltonian or Lindbladian descriptions of common physical systems (fixed and flux-tunable qubits, resonators, different types of coupling), specification of a list of devices to model the signal chain of the experiment (local oscillator, AWG, mixers, distortions/attenuations), different types of readout processing, and various fidelity functions. All components can be edited by the user or taken from reference libraries, accommodating to different needs. Configurations and data are stored as JSON files, and the full capabilities are accessible as command-line scripts, allowing for easy automation.

Numeric calculations are performed using TensorFlow [1]: The simulation of the dynamics and the pre-/post-processing are formulated as a network, with well defined inputs (e.g. control and model parameters) and outputs (goal function values), connected by a large number of nodes, each performing a relatively simple operation (e.g. matrix exponentiation). TensorFlow enables the numerical computation of the Jacobian of a calculation – the gradient of each of the network outputs with respect to the network inputs (this capability is the evolution of what is known as back-propagation learning process in neural networks [73]). This process of automatic differentiation facilitates the modular structure, as any new component inherits this property, removing the need to analytically derive its gradient. Furthermore, the TensorFlow simulator is scalable, allowing deployment on a variety of high-performance computing hardware.

We note prior efforts simulating quantum circuits which allow for automatic differentiation, e.g. [2, 74], as well as large-scale simulations of quantum circuits, e.g. [75–77]. To our knowledge, the $\mathbb{C}^3$ simulator is the most detailed automatically-differentiating package available.

A. Code structure

Each component of the control stack and model needs to conform to a general boilerplate that specifies what parameters it contains and how they are used. In this modular design, each class represents a component of the experiment that takes an input applies some parameter-dependent function to it and returns a result. For example, an envelope function for pulses would have this structure:

```
import tensorflow as tf
...
def my_envelope_fuction(t, parameters):
        amplitude = parameters["amp"]
        p2 = parameters["p2"]
        ...
return tf.some_math_function(amplitude, p2, t)
```

The only requirement to this code is that mathematical functions have to be taken from the TensorFlow package to allow for automatic differentiation. As an example of a control stack element, the finite rise time of an AWG is realized with the following code:

```
class Response(Device):
    def __init__(..., rise_time, ...):
        ...
        self.params['rise_time'] = rise_time

    def process(self, iq_signal):
        ...
        t = self.params['rise_time']
```

```
        sigma = t / 4
        ...
        # Convolution with a Gaussian
        ...
        return signal
```

A signal processing chain is represented by putting the output of one control stack element into the next. In calculating figures of merit, the user can choose from a library of functions or supply their own. For example, the infidelity of a state transfer process from $|\psi_0\rangle$ to $|\psi_{\text{ideal}}\rangle$, implemented by the simulated propagator U,

```
def state_transfer_infid(U, psi_ideal, psi_0):
    psi_actual = tf.matmul(U, psi_0)
    overlap = tf_abs(
        tf.matmul(tf.linalg.adjoint(psi_ideal), psi_actual)
    )
    infid = 1 - overlap
    return infid
```

S4. C_3 MODEL FITTING GOAL FUNCTION

Here we derive f_{LL}, the function which quantifies the statistical distance, in units of standard deviations, between the model and experimentally obtained data. For further context, see Eq. 26, and associated discussion.

A. Context and Goal

Assume we have performed a series of experiments, $k \in [1, \ldots, K]$, on our quantum device. Each experiment k is repeated for n_k iterations (shots), each resulting in one of $|d|$ possible discrete results (for two qubits $d = \{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$), with the value returned for each shot assumed to be drawn independently at random from a categorical distribution, where the probabilities of the different outcomes are determined by the occupancy of the final state after the experiment and the readout process. We denote the observed frequencies of the d results by the vector m_k (for two qubits $m_k = (N_{00}/n, N_{01}/n, N_{10}/n, N_{11}/n)$, where N_j is the number of occurrences of the $|j\rangle$ result). The vector m_k is an estimate of the experimental results, and it represents a draw from a (rescaled) multinomial distribution with probabilities given by the readout values. Note that the elements of the m_k frequency vector are not independent, as they must sum up to one. We'll use the shorthand $\{m_k\}$ to denote $\{m_k\}_{k=1,\ldots,K}$.

Given a model we would like to quantify how likely it is that it underlies the observed data. We aim to capture this likelihood in a function $f(\{m_k\}|\beta)$ dependent on the model parameters β. We therefore require a method to determine the distance between the experimental result, m_k, and the model prediction, $\widetilde{m}_k(\beta)$. Hence we define $p_k(\cdot\,|\beta)$ be the model-predicted probability distribution function (PDF) for the result of experiment k, and use this PDF to determine the f function, by evaluating the probability $p_k(m_k|\beta)$ that the measured m_k could be obtained by the model with parameters β. However, as the m_k are sampled from the readout distribution, we do not expect $m_k = \widetilde{m}_k(\beta)$. Rather, we aim to define the function f such that its expectation value, $\mathrm{E}[f(\{m_k\}, \beta)]$, is zero if the underlying distributions from which $\{m_k\}$ are drawn are the same as the model-predicted PDFs.

B. The Gaussian Assumption

To simplify calculation of $\mathrm{E}[f(\{m_k\}, \beta)]$, we can make some assumptions regarding the underlying distributions. The natural $p_k(m_k|\beta)$ PDF is multinomial (or binomial if dealing with a single qubit with no leakage levels). Under the assumption that for a large number of shots all possible readouts values are likely to appear, then by the central limit theorem (De Moivre–Laplace theorem), we can approximate p_k with a normal distribution.

However, the multinomial distribution has a non-diagonal covariance matrix, as the probabilities must sum up to one. Fortunately, one can diagonalize the distribution. Which in-turn implies the Gaussian approximation for the multinomial distribution can be decomposed as a product of 1D Gaussians and possibly justify the initial assumption. A rigorous investigation is beyond the scope of this paper.

The product-of-Gaussians approximation of a multinomial experiments voids the need to partition the overall PDF into (diagonalized) readouts individual experiments – we simply have a product of a large number of one-dimensional Gaussians. Specifically, $\sum_k (d_k - 1)$ one dimensional Gaussians, as this removes the extra degree of freedom imposed by the normalization condition. From this point onward, we'll redefine K to equal the previous $\sum_k (d_k - 1)$ and no longer need to consider m_k to be a vector.

C. The f_{LL} distribution

We shall use $\{\widetilde{\mu}_k\}$ and $\{\widetilde{\sigma}_k\}$ to denote the mean and standard deviation of the model-predicted PDFs (after Gaussian assumption and multinomial diagonalization), and $\{\mu_k\}$ and $\{\sigma_k\}$ to denote the commensurate experimental values. We note that $\{\mu_k\}$ and $\{\sigma_k\}$ are unknown and unmeasured, and $\{m_k\}$ only provides an estimate of the mean. Instead the simulation values $\{m_k\}$ represent an exact estimate of the mean, hence $\{\widetilde{m}_k \equiv \widetilde{\mu}_k\}$. Note that the model parameters β determine $\{\widetilde{\mu}_k\}$ and $\{\widetilde{\sigma}_k\}$, i.e. the model parameter determine the implied model-predicted expectation value and standard deviation.

The model-predicted PDF is given by the product of normalized Gaussian distributions, and gives the probability

$$p\left(\{m_k\}\,|\beta\right) = \prod_k \frac{1}{\sqrt{2\pi}\widetilde{\sigma}_k} \exp\left(-\frac{1}{2}\left(\frac{m_k - \widetilde{\mu}_k}{\widetilde{\sigma}_k}\right)^2\right) . \tag{1}$$

when evaluated at the experimental values $\{m_k\}$. From this equation, we construct a goal function as

$$f_{\mathrm{LL}}\left(\{m_k\}\,|\beta\right) = -\log\left[\sqrt[K]{p\left(\{m_k\}\,|\beta\right)\prod_k \sqrt{2\pi}\widetilde{\sigma}_k\sqrt{e}}\right] = \frac{1}{K}\sum_K \frac{1}{2}\left(\left(\frac{m_k - \widetilde{\mu}_k}{\widetilde{\sigma}_k}\right)^2 - 1\right) . \tag{2}$$

This f_{LL} is the related to the average log-likelihood of the $\{m_k\}$ given the model parameters β, that is rescaled to give the desired expectation value. In particular: the $\sqrt[K]{\cdot}$ gives the average over the log-likehoods for each point, the $\sqrt{2\pi}\widetilde{\sigma}_k$ removes the normalization of the Gaussians, such that they take value 1 when $m_k - \widetilde{\mu}_k = 0$, and hence the log returns 0; and the $\sqrt{e}$ removes the residual part of the expectation caused by the noise in the $\{m_k\}$.

Then, in the general case, when the Gaussians determined by the model are not the same as the Gaussians in the experimental data:

$$\mathrm{E}\left[f_{\mathrm{LL}}\left(\{m_k\}\,|\beta\right)\right] = \frac{1}{2K}\sum_k \left(\left(\frac{\mu_k - \widetilde{\mu}_k}{\widetilde{\sigma}_k}\right)^2 + \left(\frac{\widetilde{\sigma}_k}{\sigma_k}\right)^2 - 1\right) , \tag{3}$$

and

$$\mathrm{Var}\left[f_{\mathrm{LL}}\left(\{m_k\}\,|\beta\right)\right] = \frac{1}{K^2}\sum_k \left(\frac{\sigma_k}{\widetilde{\sigma}_k}\right)^2 \left(\left(\frac{\mu_k - \widetilde{\mu}_k}{\widetilde{\sigma}_k}\right)^2 + \frac{1}{2}\left(\frac{\sigma_k}{\widetilde{\sigma}_k}\right)^2\right) . \tag{4}$$

When we assume that both distributions have the same $\sigma = \widetilde{\sigma}$

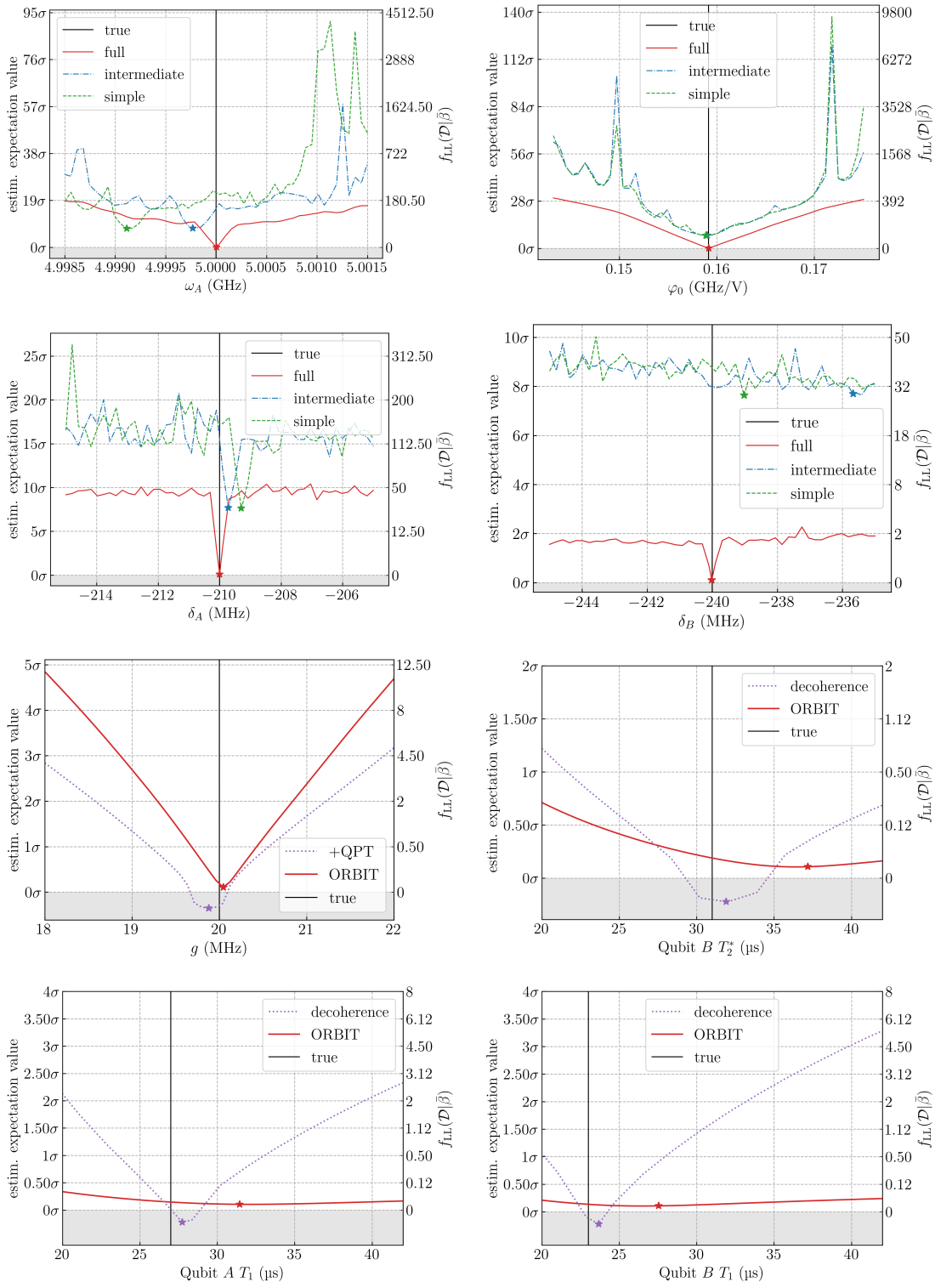
$$\mathrm{E}\left[f_{\mathrm{LL}}^{\sigma\leftarrow\widetilde{\sigma}}\left(\{m_k\}\,|\beta\right)\right] = \frac{1}{2K}\sum_k \left(\frac{\mu_k - \widetilde{\mu}_k}{\widetilde{\sigma}_k}\right)^2 , \tag{5}$$

and

$$\mathrm{Var}\left[f_{\mathrm{LL}}^{\sigma\leftarrow\widetilde{\sigma}}\left(\{m_k\}\,|\beta\right)\right] = \frac{1}{2K} + \frac{1}{K^2}\sum_k \left(\frac{\mu_k - \widetilde{\mu}_k}{\widetilde{\sigma}_k}\right)^2 . \tag{6}$$

Equation 5 then represent the square of the Mahalanobis distance (standardized Euclidean distance), giving an intuitive way to scale the f_{LL} function to understand the model match score. Indeed, when the model is exact and $\mu_k = \widetilde{\mu}_k$ we get $\mathrm{E}\left[f_{\mathrm{LL}}^{\mathrm{exact}}\left(\{m_k\}\,|\beta\right)\right] = 0$. We note, however, that the function can go take values below 0 as the variance for the exact case is $\mathrm{Var}\left[f_{\mathrm{LL}}^{\mathrm{exact}}\left(\{m_k\}\,|\beta\right)\right] = \frac{1}{2K}$. Such values indicate the standard deviation expected by the model, $\widetilde{\sigma}_k$ is larger than the standard deviation observed experimentally, σ_k.

S5. MODEL SENSITIVITY PLOTS



S6. MODEL PARAMETERS

Parameter	Real	Simple		Intermediate		Full (ORBIT)		ORBIT+QPT		Decay	
Learning	Model	Before	After	Before	After	Before	After	Before	After	Before	After
$\omega^{(A)}$ (MHz)	5000	−1.000	−0.886	−1.000	−0.230	−0.230	+0.004	+0.004	−0.016	”	”
$\delta^{(A)}$ (MHz)	210	+1.000	+0.702	+1.000	+0.281	+1.000	+0.400	+0.4008	+0.017	”	”
$\omega^{(B)}$ (MHz)	5600	+1.000	+0.592	+1.000	+0.013	+0.013	−0.003	−0.003	+0.006	”	”
$\delta^{(B)}$ (MHz)	240	+1.000	+0.981	+1.000	+4.32	+1.000	−0.016	−0.016	−0.005	”	”
φ_0 (MHz/V)	159.2	+1.592	−1.634	+1.592	−0.802	−0.802	+0.123	+0.123	+0.246	”	”
g (MHz)	20	–	–	+1.000	−0.665	−0.665	+0.046	+0.046	−0.119	”	”
T (mK)	50	–	–	–	–	+5.000	−3.172	−3.172	−0.216	”	”
$T_1^{(A)}$ (μs)	27	–	–	–	–	+4.000	+4.439	+4.439	+0.021	+0.021	+0.738
$T_{2*}^{(A)}$ (μs)	39	–	–	–	–	+2.000	+1.994	+1.994	−2.353	−2.353	−0.020
$T_1^{(B)}$ (μs)	23	–	–	–	–	+3.000	+4.543	+4.543	+5.704	+5.704	+0.666
$T_{2*}^{(B)}$ (μs)	31	–	–	–	–	+5.000	+6.183	+6.183	+4.716	+4.716	+0.897
$p_{0\to0}^{(A)}$ (%)	97	–	–	–	–	−2.00	−0.84	−0.84	−0.11	”	”
$p_{1\to1}^{(A)}$ (%)	96	–	–	–	–	+0.20	+0.38	+0.38	+0.02	”	”
$p_{0\to0}^{(B)}$ (%)	96	–	–	–	–	−2.00	−0.62	−0.62	−0.03	”	”
$p_{1\to1}^{(B)}$ (%)	95	–	–	–	–	+0.20	+0.08	+0.08	+0.01	”	”

TABLE S1: Overview of the parameters of the “real” model (reference values), and the candidate models, before and after the C_3 learning for different data-sets. Candidate model values are shown as difference from reference values. Dashes (–) indicate parameters not present in the model, quotation marks (”) indicate parameters not being changed.

[1] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, and M. Isard, *TensorFlow: Large-scale machine learning on heterogeneous systems,* (2015), software available from tensorflow.org.

[2] N. Leung, M. Abdelhafez, J. Koch, and D. Schuster, *Speedup for quantum optimal control from automatic differentiation based on graphics processing units,* Phys. Rev. A **95**, 042318 (2017).

[3] I. I. Rabi, *Space quantization in a gyrating magnetic field,* Phys. Rev. **51**, 652 (1937).

[4] N. F. Ramsey, *A molecular beam resonance method with separated oscillating fields,* Phys. Rev. **78**, 695 (1950).

[5] E. L. Hahn, *Spin echoes,* Phys. Rev. **80**, 580 (1950).

[6] B. Pokharel, N. Anand, B. Fortman, and D. A. Lidar, *Demonstration of fidelity improvement using dynamical decoupling with superconducting qubits,* Phys. Rev. Lett. **121**, 220502 (2018).

[7] C. Ferrie, C. Granade, G. Paz-Silva, and H. M. Wiseman, *Bayesian quantum noise spectroscopy,* New J. Phys. **20**, 123005 (2018).

[8] M. A. Rol, L. Ciorciaro, F. K. Malinowski, B. M. Tarasinski, R. E. Sagastizabal, C. C. Bultink, Y. Salathe, N. Haandbaek, J. Sedivy, and L. DiCarlo, *Time-domain characterization and correction of on-chip distortion of control pulses in a quantum processor,* Appl. Phys. Lett. **116**, 054001 (2020).

[9] J. van Dijk, E. Kawakami, R. Schouten, M. Veldhorst, L. Vandersypen, M. Babaie, E. Charbon, and F. Sebastiano, *Impact of classical control electronics on qubit fidelity,* Phys. Rev. Applied **12**, 044054 (2019).

[10] M. Jerger, A. Kulikov, Z. Vasselin, and A. Fedorov, *In situ characterization of qubit control lines: a qubit as a vector network analyzer,* Phys. Rev. Lett. **123**, 150501 (2019).

[11] I. N. Hincks, C. E. Granade, T. W. Borneman, and D. G. Cory, *Controlling quantum devices with nonlinear hardware,* Phys. Rev. Applied **4**, 024012 (2015).

[12] S. Gustavsson, O. Zwier, J. Bylander, F. Yan, F. Yoshihara, Y. Nakamura, T. P. Orlando, and W. D. Oliver, *Improving quantum gate fidelities by using a qubit to measure microwave pulse distortions,* Phys. Rev. Lett. **110**, 040502 (2013).

[13] R. S. Gupta, A. R. Milne, C. L. Edmunds, C. Hempel, and M. J. Biercuk, *Adaptive scheduling of noise characterization in quantum computers,* (2019), arXiv:1904.07225.

[14] R. Harper, S. T. Flammia, and J. J. Wallman, *Efficient learning of quantum noise,* (2019), arXiv:1907.13022.

[15] J. Kelly, R. Barends, A. Fowler, A. Megrant, E. Jeffrey, T. White, D. Sank, J. Mutus, B. Campbell, Y. Chen, *et al.*, *Scalable in situ qubit calibration during repetitive error detection,* Phys. Rev. A **94**, 032321 (2016).

[16] L. Casparis, T. W. Larsen, M. S. Olsen, F. Kuemmeth, P. Krogstrup, J. Nygård, K. D. Petersson, and C. M. Marcus, *Gatemon benchmarking and two-qubit operations,* Phys. Rev. Lett. **116**, 150505 (2016).

[17] C. Miquel, J. P. Paz, and W. H. Zurek, *Quantum computation with phase drift errors,* Phys. Rev. Lett. **78**, 3971 (1997).

[18] R. C. Bialczak, R. McDermott, M. Ansmann, M. Hofheinz, N. Katz, E. Lucero, M. Neeley, A. D. O'Connell, H. Wang, A. N. Cleland, and J. M. Martinis, $1/f$ *flux noise in josephson phase qubits,* Phys. Rev. Lett. **99**, 187006 (2007).

[19] J. J. Burnett, A. Bengtsson, M. Scigliuzzo, D. Niepce, M. Kudra, P. Delsing, and J. Bylander, *Decoherence benchmarking of superconducting qubits,* npj Quantum Inf. **5**, 1 (2019).

[20] M. Y. Niu, V. Smelyanskyi, P. Klimov, S. Boixo, R. Barends, J. Kelly, Y. Chen, K. Arya, B. Burkett, D. Bacon, Z. Chen, B. Chiaro, R. Collins, A. Dunsworth, B. Foxen, A. Fowler, C. Gidney, M. Giustina, R. Graff, T. Huang, E. Jeffrey, D. Landhuis, E. Lucero, A. Megrant, J. Mutus, X. Mi, O. Naaman, M. Neeley, C. Neill, C. Quintana, P. Roushan, J. M. Martinis, and H. Neven, *Learning non-markovian quantum noise from moiré-enhanced swap spectroscopy with deep evolutionary algorithm,* (2019), arXiv:1912.04368.

[21] E. Knill, D. Leibfried, R. Reichle, J. Britton, R. B. Blakestad, J. D. Jost, C. Langer, R. Ozeri, S. Seidelin, and D. J. Wineland, *Randomized benchmarking of quantum gates,* Phys. Rev. A **77**, 012307 (2008).

[22] A. Zhang, J. Xie, H. Xu, K. Zheng, H. Zhang, Y.-T. Poon, V. Vedral, and L. Zhang, *Experimental self-characterization of quantum measurements,* Phys. Rev. Lett. **124**, 040402 (2020).

[23] P. Mundada, G. Zhang, T. Hazard, and A. Houck, *Suppression of qubit crosstalk in a tunable coupling superconducting circuit,* Phys. Rev. Appl **12**, 054023 (2019).

[24] P. Murali, D. C. McKay, M. Martonosi, and A. Javadi-Abhari, *Software mitigation of crosstalk on noisy intermediate-scale quantum computers,* (2020), arXiv:2001.02826.

[25] D. Burgarth, K. Maruyama, and F. Nori, *Coupling strength estimation for spin chains despite restricted access,* Phys. Rev. A **79**, 020305 (2009).

[26] C. Di Franco, M. Paternostro, and M. Kim, *Hamiltonian tomography in an access-limited setting without state initialization,* Phys. Rev. Lett. **102**, 187203 (2009).

[27] L. E. De Clercq, R. Oswald, C. Flühmann, B. Keitch, D. Kienzler, H.-Y. Lo, M. Marinelli, D. Nadlinger, V. Negnevitsky, and J. P. Home, *Estimation of a general time-dependent hamiltonian for a single qubit,* Nat. Commun. **7**, 1 (2016).

[28] A. Sone and P. Cappellaro, *Hamiltonian identifiability assisted by a single-probe measurement,* Phys. Rev. A **95**, 022335 (2017).

[29] P. Yang, M. Yu, R. Betzholz, C. Arenz, and J. Cai, *Complete quantum-state tomography with a local random field,* Phys. Rev. Lett. **124**, 010405 (2020).

[30] R. L. Kosut and H. Rabitz, *Identification of quantum systems,* IFAC Proceedings Volumes **35**, 397 (2002).

[31] S. Schirmer, A. Kolli, and D. Oi, *Experimental hamiltonian identification for controlled two-level systems,* Phys. Rev. A **69**, 050306 (2004).

[32] D. Oi and S. Schirmer, *Quantum system characterization with limited resources,* Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences **370**, 5386 (2012).

[33] D. Burgarth and K. Yuasa, *Quantum system identification,* Phys. Rev. Lett. **108**, 080502 (2012).

[34] J. Zhang and M. Sarovar, *Identification of open quantum systems from observable time traces,* Phys. Rev. A **91**, 052121 (2015).

[35] S.-Y. Hou, H. Li, and G.-L. Long, *Experimental quantum hamiltonian identification from measurement time traces,* Science Bulletin **62**, 863 (2017).

[36] S. Schirmer and D. Oi, *Quantum system identification by bayesian analysis of noisy data: Beyond hamiltonian tomography,* Laser Phys. **20**, 1203 (2010).

[37] M. P. Stenberg, Y. R. Sanders, and F. K. Wilhelm, *Efficient estimation of resonant coupling between quantum systems,* Phys. Rev. Lett. **113**, 210404 (2014).

[38] M. P. Stenberg, O. Köhn, and F. K. Wilhelm, *Characterization of decohering quantum systems: Machine learning approach,* Phys. Rev. A **93**, 012122 (2016).

[39] M. P. Stenberg and F. K. Wilhelm, *Simultaneous model selection and parameter estimation: A superconducting qubit coupled to a bath of incoherent two-level systems,* Phys. Rev. A **94**, 052119 (2016).

[40] A. Sergeevich, A. Chandran, J. Combes, S. D. Bartlett, and H. M. Wiseman, *Characterization of a qubit hamiltonian using adaptive measurements in a fixed basis,* Phys. Rev. A **84**, 052315 (2011).

[41] N. Wiebe, C. Granade, C. Ferrie, and D. G. Cory, *Hamiltonian learning and certification using quantum resources,* Phys. Rev. Lett. **112**, 190501 (2014).

[42] N. Wiebe, C. Granade, C. Ferrie, and D. Cory, *Quantum hamiltonian learning using imperfect quantum resources,* Phys. Rev. A **89**, 042314 (2014).

[43] N. Wiebe, C. Granade, and D. G. Cory, *Quantum bootstrapping via compressed quantum hamiltonian learning,* New J. Phys. **17**, 022005 (2015).

[44] J. Wang, S. Paesani, R. Santagati, S. Knauer, A. A. Gentile, N. Wiebe, M. Petruzzella, J. L. O'Brien, J. G. Rarity, A. Laing, *et al.*, *Experimental quantum hamiltonian learning,* Nat. Phys. **13**, 551 (2017).

[45] J. H. Cole, S. G. Schirmer, A. D. Greentree, C. J. Wellard, D. K. L. Oi, and L. C. L. Hollenberg, *Identifying an experimental two-state hamiltonian to arbitrary accuracy,* Phys. Rev. A **71**, 062312 (2005).

[46] A. F. Emery and A. V. Nenarokomov, *Optimal experiment design,* Meas. Sci. Technol. **9**, 864 (1998).

[47] T. J. Loredo, in *AIP Conference Proceedings*, Vol. 707 (American Institute of Physics, 2004) pp. 330–346.

[48] C. E. Granade, C. Ferrie, N. Wiebe, and D. G. Cory, *Robust online hamiltonian learning,* New J. Phys. **14**, 103013 (2012).

[49] A. A. Gentile, B. Flynn, S. Knauer, N. Wiebe, S. Paesani, C. E. Granade, J. G. Rarity, R. Santagati, and A. Laing, *Learning models of quantum systems from experiments,* (2020), arXiv:2002.06169.

[50] T. J. Evans, R. Harper, and S. T. Flammia, *Scalable bayesian hamiltonian learning,* (2019), arXiv:1912.07636.

[51] J. F. Ralph, S. Maskell, and K. Jacobs, *Multiparameter estimation along quantum trajectories with sequential monte carlo*

methods, Phys. Rev. A **96**, 052306 (2017).
[52] A. Shabani, M. Mohseni, S. Lloyd, R. L. Kosut, and H. Rabitz, *Estimation of many-body quantum hamiltonians via compressive sensing,* Phys. Rev. A **84**, 012107 (2011).
[53] K. Rudinger and R. Joynt, *Compressed sensing for hamiltonian reconstruction,* Phys. Rev. A **92**, 052322 (2015).
[54] J. Zhang and M. Sarovar, *Quantum hamiltonian identification from measurement time traces,* Phys. Rev. Lett. **113**, 080401 (2014).
[55] S. Xue, R. Wu, D. Li, and M. Jiang, *A gradient algorithm for hamiltonian identification of open quantum systems,* (2019), arXiv:1905.09990.
[56] S. Krastanov, S. Zhou, S. T. Flammia, and L. Jiang, *Stochastic estimation of dynamical variables,* Quantum Science and Technology **4**, 035003 (2019).
[57] B. Lusch, J. N. Kutz, and S. L. Brunton, *Deep learning for universal linear embeddings of nonlinear dynamics,* Nat. Commun. **9**, 1 (2018).
[58] E. Flurin, L. S. Martin, S. Hacohen-Gourgy, and I. Siddiqi, *Using a recurrent neural network to reconstruct quantum dynamics of a superconducting qubit from physical observations,* Phys. Rev. X **10**, 011006 (2020).
[59] M. Dalgaard, F. Motzoi, J. J. Sørensen, and J. Sherson, *Global optimization of quantum dynamics with alphazero deep exploration,* npj Quantum Inf. **6** (2020), 10.1038/s41534-019-0241-0.
[60] M. Gevers, *Identification for control: From the early achievements to the revival of experiment design,* European Journal of Control **11**, 335 (2005).
[61] C. R. Rojas, J. S. Welsh, G. C. Goodwin, and A. Feuer, *Robust optimal experiment design for system identification,* Automatica **43**, 993 (2007).
[62] L. Ljung, *Perspectives on system identification,* Annual Reviews in Control **34**, 1 (2010).
[63] X. Bombois, M. Gevers, R. Hildebrand, and G. Solari, *Optimal experiment design for open and closed-loop system identification,* Communications in Information and Systems **11**, 197 (2011).
[64] D. J. Egger and F. K. Wilhelm, *Adaptive hybrid optimal quantum control for imprecisely characterized systems,* Phys. Rev. Lett. **112**, 240503 (2014).
[65] Q.-M. Chen, X. Yang, C. Arenz, R.-B. Wu, X. Peng, I. Pelczer, and H. Rabitz, *Combining the synergistic control capabilities of modelling and experiments: illustration of finding a minimum time quantum objective,* (2018), arXiv:1812.05042.
[66] R.-B. Wu, B. Chu, D. H. Owens, and H. Rabitz, *Data-driven gradient algorithm for high-precision quantum control,* Phys. Rev. A **97**, 042122 (2018).
[67] J. A. Nelder and R. Mead, *A simplex method for function minimization,* The Computer Journal **7**, 308 (1965).
[68] L. Han and M. Neumann, *Effect of dimensionality on the nelder–mead simplex method,* Optim. Method. Softw. **21**, 1 (2006).
[69] J. Rapin, M. Gallagher, P. Kerschke, M. Preuss, and O. Teytaud, in *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, GECCO '19 (Association for Computing Machinery, New York, NY, USA, 2019) p. 1888–1896.
[70] M. Rol, C. Dickel, S.Asaad, N. Langford, C. Bultink, R. Sagastizabal, N. Langford, G. de Lange, X. Fu, S. de Jong, F. Luthi, W. Vlothuizen, and M. S. Moreira, Pycqed_py3, (2019).
[71] Labber.org, Labber – software for instrument control and lab automation, (2020).
[72] C. J. Kalkman, *Labview: A software system for data acquisition, data analysis, and instrument control,* J. Clin. Monit. **11**, 51 (1995).
[73] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, *Learning representations by back-propagating errors,* Nature **323**, 533 (1986).
[74] M. Abdelhafez, D. I. Schuster, and J. Koch, *Gradient-based optimal control of open quantum systems using quantum trajectories and automatic differentiation,* Phys. Rev. A **99**, 052327 (2019).
[75] D. Willsch, M. Nocon, F. Jin, H. De Raedt, and K. Michielsen, *Gate-error analysis in simulations of quantum computers with transmon qubits,* Phys. Rev. A **96**, 062302 (2017).
[76] D. Willsch, M. Willsch, F. Jin, H. De Raedt, and K. Michielsen, *Testing quantum fault tolerance on small systems,* Phys. Rev. A **98**, 052348 (2018).
[77] D. Willsch, *Supercomputer simulations of transmon quantum computers,* (2020), arXiv:2008.13490.