

Quantum optimal control of superconducting qubits based on machine-learning characterization

Élie Genois^{1,*}, Noah J. Stevenson,² Noah Goss,^{2,3} Irfan Siddiqi,^{2,3,4,5} and Alexandre Blais^{1,5}

¹*Institut Quantique & Département de Physique, Université de Sherbrooke, Québec J1K 2R1, Canada*

²*Quantum Nanoelectronics Laboratory, University of California, Berkeley, Berkeley, California 94720, USA*

³*Applied Math & Computational Research Division, Lawrence Berkeley National Lab, Berkeley, California 94720, USA*

⁴*Materials Science Division, Lawrence Berkeley National Lab, Berkeley, California 94720, USA*

⁵*Canadian Institute for Advanced Research, Toronto, Ontario M5G 1M1, Canada*



(Received 18 May 2025; revised 16 August 2025; accepted 19 August 2025; published 26 September 2025)

Implementing fast and high-fidelity quantum operations using open-loop quantum optimal control relies on having an accurate model of the quantum dynamics. Any deviations between this model and the complete dynamics of the device, such as the presence of spurious modes or pulse distortions, can degrade the performance of optimal controls in practice. Here, we propose an experimentally simple approach to realize optimal quantum controls tailored to the device parameters and environment while specifically characterizing this quantum system. Concretely, we use physics-inspired machine learning to infer an accurate model of the dynamics from experimentally available data and then optimize our experimental controls on this trained model. We show the power and feasibility of this approach by optimizing arbitrary single-qubit operations in detailed numerical simulations of a superconducting transmon qubit. We demonstrate that this framework produces an accurate description of the device dynamics under arbitrary controls, together with the precise pulses achieving arbitrary single-qubit gates with a high fidelity of $\sim 99.99\%$.

DOI: [10.1103/d9yg-d3qr](https://doi.org/10.1103/d9yg-d3qr)

I. INTRODUCTION

The precise characterization and control of quantum devices are central to the development of useful quantum technologies. The powerful framework of quantum optimal control (QOC) theory can be used to go from a given description of the quantum dynamics, such as a characterized model, to realizing arbitrary quantum operations with maximal fidelity and minimal duration [1–3]. The successful practical implementation of many QOC approaches, such as the GRAPE [4] and Krotov [5] methods, thus rely on having an accurate model of the system dynamics to produce the desired output quantum state or process [6]. In simulation, optimizing the input controls using these methods can routinely yield quantum operations with decoherence-limited or even machine-precision fidelity, and these controls can have much shorter durations compared to what is achievable with simpler, monochromatic and flat, pulse shapes [7–11].

A critical problem with using open-loop QOC approaches in experiments is model bias, since the performance of the resulting controls is intrinsically limited by the underlying model accuracy. Indeed, any mismatch between the model used to describe the quantum evolution and the actual dynamics of the physical system can cause the optimal controls to perform significantly worse in practice. This performance degradation is routinely observed when considering parameter deviations to the assumed model, and has led to the development of robust QOC approaches [12–15]. These approaches sacrifice some control performance for the benefit of robustness under certain model parameter variations. However, beyond being inaccurate with model parameters deviating from their “true” values due to finite characterization precision and system drifts, the model used for QOC can additionally be incomplete, which also leads to important biases. Out-of-model dynamics typically originate from unaccounted frequency- and power-dependent distortions in the control lines, crosstalk between qubits and control lines, and more generally coupling to spurious modes not included in the system modeling, such as material defects, box modes, and neighboring couplers and readout apparatus. Precisely

*Contact author: elie.genois@usherbrooke.ca

characterizing and modeling the dynamics associated with each of these potential interactions is a challenging task, both experimentally and numerically. There is thus a need for alternative approaches to optimally control quantum systems.

The shortcomings of using inaccurate models in the control optimization has led to recent proposals of closed-loop optimization approaches based on reinforcement learning (RL), which rely on direct interactions between a controller and the quantum system of interest [16–20]. Using such a feedback control approach, the QOC problem can be made model-free, thus alleviating the aforementioned problem of model bias. For example, this approach was applied experimentally to realize high-fidelity single- and two-qubit quantum gates, as well as a quantum error-correction stabilization protocol in superconducting quantum devices [21–23].

Although useful for precisely calibrating a specific operation, these model-free approaches possess important drawbacks, notably in terms of sample efficiency, generalizability beyond performing a single quantum operation, ease of experimental implementation given the need for real-time feedback, and performance of the control solutions. For instance, Porotti *et al.* [20] demonstrated that model-based gradient-ascent approaches, such as GRAPE, significantly outperform model-free RL approaches on standard state-preparation optimal control tasks, in terms of both state fidelity and data efficiency. This can be understood from the fact that, in the model-free setting, the task of the learning agent is much more complex, as it needs to both resolve how a given control pulse (action) impacts the state or process fidelity (reward), and learn how to improve that control by navigating the enormous control parameter space. The first part of this task can be recognized as a quantum characterization problem, whereas the second part of finding the optimal control strategy can be directly achieved using one of the many successful QOC approaches, instead of relying on the trial-and-error exploration of the RL approach.

Based on this understanding, we propose in this work to simplify the quantum optimal control problem by breaking it down into two parts that we perform in succession. First, we frame the quantum characterization problem, or model learning problem, as a supervised machine-learning (ML) task. We use a parameterized representation to learn a description of the quantum system dynamics directly from experimentally available data. Second, we use that trained model in a gradient-based optimal control loop to find the external controls realizing our target operations. In the following, we demonstrate that this modular and easily implementable approach can yield high-fidelity controls using experimentally realistic data, while providing notable benefits in terms of data efficiency, scalability to complex control problems, and device characterization.

The paper is organized as follows. In Sec. II, we describe our machine-learning-based optimal control approach and its benefits over alternative control approaches. Using detailed numerical simulations, we then present a case-study implementation of our approach and demonstrate its performance for realizing high-fidelity arbitrary single-qubit gates in a transmon qubit in Sec. III. In Sec. IV, we analyze the impact of out-of-model dynamics on the optimal control performance, and demonstrate the distinct advantages of our approach in the presence of model bias. We conclude with a short discussion on the relevance of this work in Sec. V.

II. MACHINE-LEARNING BASED QOC

The two main steps of our proposed quantum optimal control approach based on machine-learning characterization are presented schematically in Fig. 1. The characterization task consists of constructing a model that captures the transformation from arbitrary input pulse shapes to the resulting quantum state observables of interest. In this work, we focus on qubit gates and therefore take these observables to be the Pauli matrices on n qubits, which are informationally complete to describe qubit states. We can thus consider that the model we want to construct outputs the full quantum state $\rho(t)$. Note that a more restricted set of observables could be used if acquiring this information is prohibitive, for example, in the case of a large Hilbert space, as long as the relevant metric to optimize can be described by the model output observables.

The idea of our approach is to learn this model directly from data taken on the device of interest using a supervised machine-learning strategy. To construct the dataset, we simply drive the quantum system sequentially with a large set of different pulses, and perform projective measurements to obtain bits of information about how the quantum state was transformed by each of these pulses. After averaging over a chosen number of shots, these measurement outcomes form the labels that the model will be trained to predict, using a loss function quantifying the distance between the model predictions and the labels. As detailed in Sec. III, additional physically motivated loss terms are added to regularize the model's training, as was done in our previous work [24]. We emphasize that this machine-learning training is performed offline such that no real-time feedback between the model and the quantum system is required.

The second step consists of directly using the trained model in a gradient-based optimal control loop to find our control strategy. At this point after the learning stage, the model parameters are fixed and only the parameterized pulse shapes are optimized. Given the autodifferentiable nature of the model and its fast evaluation on graphics processing units (GPUs), these optimizations can be performed in less than a minute on standard hardware.

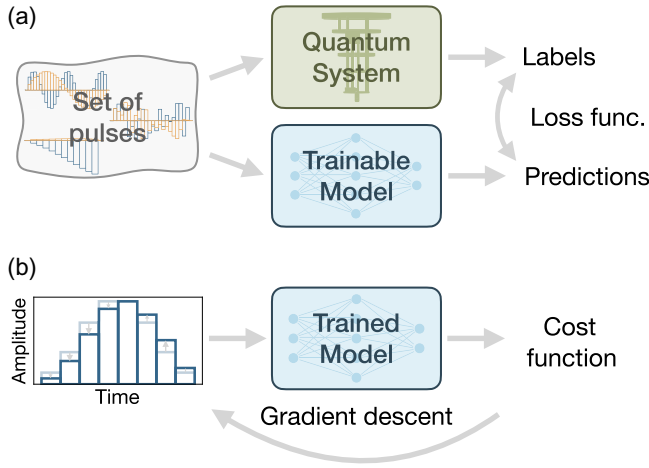


FIG. 1. Optimally controlling a quantum system using machine-learning characterization. (a) A supervised machine-learning approach is used to characterize the quantum dynamics produced by arbitrary input controls. The dataset is constructed by applying a set of diverse pulses sequentially to the quantum system and obtaining labels of the system observables using (projective) measurements. A parameterized representation (blue box) learns this transformation from the input pulse shape to output quantum observables by minimizing a loss function related to the distance between its predictions and the data labels. (b) The trained model is used directly in a gradient-based quantum optimal control loop to find the pulses best realizing the quantum operations of interest. The parameterized pulse shape is iteratively updated so as to minimize a cost function associated with a state or process infidelity.

Additionally, the optimization can be designed to include arbitrary experimental limitations in the cost function [25], so as to directly find optimal controls that are easily implementable in practice. Importantly, in contrast to direct feedback control approaches, the trained representation is not tied to a specific quantum gate, and we can use the trained model to optimize for arbitrary quantum operations which can be described and realized by the same input controls and output observables as the model. For instance, using a model outputting the quantum state of n qubits, we can optimize for arbitrary qubit gates and qubit state preparations. This can be directly generalized to higher-dimensional systems such as qudits and bosonic modes, as long as these additional degrees of freedom are observable in the quantum system of interest.

An important advantage of our two-step approach to quantum control is that it is highly modular and customizable. First, having a trained model describing the system dynamics for arbitrary input controls that is fast to evaluate can be very valuable for understanding, calibrating, and improving the quantum system of interest. Indeed, this model can be used as a heuristic digital twin of the device to perform simulations and to prototype new protocols. This is especially useful given that the

trained model heuristically includes experimentally relevant imperfections that were learned from data and that are typically not accounted for when modeling the system. Additionally, this model is fully differentiable, which means one can replace typically used parameter sweeps or gradient-free approaches with gradient-based optimization of the controls, which are faster and yield better solutions [26,27].

The second aspect making our approach modular is that the specific architecture used to learn from data, together with the optimal control algorithm used subsequently, can be explicitly engineered to satisfy the specific data, speed, and performance requirements of a given quantum system. This is particularly useful to explore the bias-variance trade-offs of learning an accurate representation of the device dynamics and to perform model selection. For instance, using a black-box learning model to remove any potential bias might require too much data to be trained to reach the desired precision, whereas a fully principled approach based on a physics description might be computationally prohibitive to model or unable to describe the dynamics accurately enough. Using a graybox approach [28] combining physical priors about the device properties together with additional freedom might be optimal in such a realistic scenario, which can be directly realized within our framework. Indeed, our approach allows us to use as much prior knowledge as desired in the learning model, anywhere from a parameterized master equation [24,29] to a fully general neural network, as long as the model outputs are informationally complete for quantifying the desired quantum operations fidelity.

We note that approaches of directly combining characterization (system identification) and control have been studied in control theory [30] and applied to quantum systems using a variety of models and data [28,31–36]. Focused on experimental feasibility and on mitigating the problem of model bias, this work provides a comprehensive framework for applying such an approach to superconducting and other solid-state devices and demonstrates its performance through detailed simulations. In addition, an important distinction of our work is the demonstration for the first time of using a trained neural network as the model for performing open-loop quantum optimal control, which allows us to significantly mitigate any model bias in practice. In contrast to the similar approach by Yousry *et al.* [28], we show that learning an explicit Hamiltonian description of the dynamics is not required for performing arbitrary QOC with high fidelity.

III. TRANSMON SINGLE-QUBIT GATES

We now present a comprehensive case study of our approach applied to realizing fast and high-fidelity microwave single-qubit gates in a fixed-frequency superconducting transmon qubit [37]. We emphasize that our

approach is agnostic to the specific physical implementation and could be applied to other platforms following the steps detailed here. The available qubit controls in the chosen superconducting qubit system are microwave pulses generated at room temperature by classical control electronics, which travel down a cryogenic fridge before reaching the qubit. Information about the dynamics resulting from the controls can be acquired via projective measurement of arbitrary qubit operators using standard dispersive qubit readout [38,39]. Given these input controls and available output observables, the task of interest is to design the external microwave pulses so as to realize arbitrary qubit unitaries with the highest fidelity and minimal duration.

The following four subsections detail the four steps required for achieving this task using our machine-learning-based quantum optimal control approach (MLQOC). Namely, the framework consists of: acquiring a dataset on the device; training a machine-learning model on these data; performing QOC optimizations using this trained model; and testing the performance of the optimal controls on the device of interest. As a proof of principle of the method, here the dataset is generated by numerical simulation of the device (first step), and the same is true for testing the performance of the optimized controls (fourth step). The other two steps remain unchanged when working with an experimental device. In our simulations, we take special care not to make approximations that would artificially simplify the model learning and optimal control tasks considered, in an effort to provide a convincing demonstration of our approach in an experimentally realistic scenario.

A. Simulation dataset

1. Physical model

We consider a transmon qubit capacitively coupled to a microwave drive line described by the Hamiltonian ($\hbar = 1$) [37]

$$\begin{aligned}\hat{H}_{\text{tot}}(t) &= 4E_C(\hat{n} - n_g)^2 - E_J \cos \hat{\phi} + \Omega(t)\hat{n} \\ &= \hat{H}_{\text{trans}} + \Omega(t)\hat{n},\end{aligned}\quad (1)$$

with E_C (E_J) the charging (Josephson) energy, $\Omega(t)$ the applied microwave drive coupling to the transmon charge operator $\hat{n}$, and $\hat{\phi}$ the canonically conjugate phase operator. Given that the qubit is operated in the transmon regime, $E_J/E_C = 110$, and that we are interested in quantum operations limited to the qubit subspace, we can safely ignore the gate charge n_g [37]. Throughout this work, we simulate this full Hamiltonian keeping the first five eigenstates of $\hat{H}_{\text{trans}}$. We do not perform any rotating-wave approximation (RWA) in order to capture the fast-oscillating dynamics that are relevant for high-fidelity and fast operations [7,40].

We parameterize the drive pulses $\Omega(t)$ as the input to the arbitrary waveform generator (AWG) that produces in-phase (I) S_I and quadrature (Q) S_Q signals. These signals are then combined to a local oscillator (LO) of frequency ω_{LO} to be up-converted to the desired gigahertz-frequency signals, a process known as sideband mixing [41,42]. This pulse parameterization reflects the actual controls of a realistic experiment, allowing our model to capture potential IQ-mixer imperfections and other pulse distortions, in addition to allowing for precise control of the pulse's frequency spectrum. In the absence of imperfections in the control electronics and pulse distortion in the control lines, the microwave pulse reaching the qubit is described by

$$\Omega(t) = S_I(t) \cos(\omega_{\text{LO}}t) + S_Q(t) \sin(\omega_{\text{LO}}t). \quad (3)$$

To drive the transmon on resonance at ω_q and avoid IQ-mixer imperfections producing distorted output signals at frequencies close to ω_{LO} , the AWG signals are typically generated by convolving the pulse envelope with an intermediate-frequency oscillation at frequency ω_{IF} such that $\omega_{\text{LO}} + \omega_{\text{IF}} = \omega_q$ [42]. We reproduce this experimental condition here using typical values of $\omega_{\text{IF}}/2\pi = 100$ MHz and $\omega_{\text{LO}}/2\pi = 6.198$ GHz to reach the qubit frequency at $\omega_q/2\pi \approx 6.298$ GHz [43].

As shown in Appendix F, when restricting the description to the computational states of the transmon, the effect of the drive $\Omega(t)$ takes the usual form in the rotating frame of the qubit, $\hat{H}_{\text{qubit}} = I(t)\hat{\sigma}_x + Q(t)\hat{\sigma}_y$, for a resonant drive and under the RWA. The real-valued signals I and Q can be expressed as linear combinations of the AWG signals S_I and S_Q . We thus understand how controlling S_I and S_Q allows us to perform arbitrary single-qubit gates. Here, we avoid the approximations mentioned above and simulate the full transmon Hamiltonian in Eq. (2) directly using Eq. (3) with arbitrary time-dependent signals S_I and S_Q .

In the AWG, as in our simulations, the pulse shapes S_I and S_Q are specified by real amplitudes positioned at a finite number of times during the operation, called *pixels*. We take the size of these pixels to be 1 ns. Importantly, we apply a Gaussian filter to these discrete signals to (i) interpolate between pixels and simulate experimentally accurate continuous-time evolutions beyond a piecewise-constant pulse approximation [7,27], and (ii) account for the finite bandwidth of the AWG and the filters used in experiments. We use a Gaussian filter standard deviation of 250 MHz.

To simulate the full time dynamics of the system under the application of the drives, we solve the Lindblad master equation (ME)

$$\frac{d\hat{\rho}}{dt} = -i[\hat{H}_{\text{tot}}(t), \hat{\rho}] + \gamma\mathcal{D}[\hat{b}]\hat{\rho} + 2\gamma_\varphi\mathcal{D}[\hat{b}^\dagger\hat{b}]\hat{\rho}, \quad (4)$$

which accounts for transmon relaxation and dephasing [39]. In this expression, γ (γ_φ) is the relaxation (pure

dephasing) rate, and $\mathcal{D}[\hat{X}]\hat{\rho} = \hat{X}\hat{\rho}\hat{X}^\dagger - \{\hat{X}^\dagger\hat{X}, \hat{\rho}\}/2$ is the Lindblad dissipator. In the following, we use relatively high relaxation and coherence times of $T_1 = T_2 = 300 \mu\text{s}$, corresponding to $\gamma/2\pi = 3.33 \text{ kHz}$ and $\gamma_\phi/2\pi = 1.67 \text{ kHz}$. This choice allows us to resolve the finite control precision of our MLQOC approach, beyond the gate fidelity limit set by decoherence. We use the open-source library `dynamics` to perform these simulations, which allows us to use GPU acceleration and to efficiently batch the simulations over multiple pulse shapes in parallel [44].

2. Experiment and dataset generation

As illustrated in Fig. 2(a), the experiment we consider consists of three steps, where (I) the transmon is prepared in one of the six cardinal states of the Bloch sphere, (II) a microwave drive with a given pulse shape and duration is applied to the qubit, and (III) the qubit is read out in the basis $\sigma_j \in \{X, Y, Z\}$. This same experiment is repeated N_{shots} times to acquire statistics, where $N_{\text{shots}} \geq 1$, and the label associated with this specific pulse shape and preparation is the qubit expectation value estimate resulting from averaging these shots. We emphasize that we never feed the full quantum state to the model and that this label, i.e., the average measurement result, is a realistic noisy estimate of the true qubit expectation value that the model is trying to predict. State preparation and measurement (SPAM) errors can add noise to these labels. This noise can be made effectively unbiased using standard error-mitigation strategies based on inverting the confusion matrix or on measuring half of the shots with the negative measurement operator. For simplicity in evaluating model performances, we did not model the effect of the mitigated SPAM errors which would typically be significantly smaller than shot noise in our simulations ($N_{\text{shots}} = 32$).

To construct the supervised ML datasets, this experiment is repeated for all of the pulse shapes in a chosen set of pulses, while randomizing over the prepared state and measurement axis for each of these pulses. Implementation details for constructing the pulse set, so as to efficiently sample the control space, are presented in Appendix C. We have found that using a combination of random pulse shapes together with physically motivated envelopes, such as Gaussian, sinusoidal, and flat-top envelopes with DRAG (derivative removal by adiabatic gate) components [45] was sufficient for the ML models considered to learn an accurate and generalizable representation of the quantum dynamics.

Putting these data together in the form of supervised learning datasets, each input consists of a one-hot encoded vector $\vec{p}$ describing which of the six cardinal states was prepared, together with a two-dimensional real array $\mathbf{S} = (\vec{S}_I, \vec{S}_Q)$ containing the pulse-shape amplitudes at each

pixel of the evolution for the two drive quadratures. The output labels are a combination of a one-hot encoded vector $\vec{m}$ capturing which Pauli operator was measured together with a single floating-point number representing the associated qubit expectation value estimate $\text{Tr}[\sigma_j \rho(t)]$. Finally, we split this dataset into training, validation, and test sets, as is standard in supervised learning approaches both to avoid overfitting by selecting the best model from the performance on the validation set, and to obtain unbiased model performance metrics by evaluating the chosen models on a separate test set [46].

B. Model learning

1. Model architecture

The parameterized representation we use to learn the quantum dynamics from the data described above is the physics-inspired neural network illustrated in Fig. 2(a). It is principally composed of a recurrent-neural-network (RNN) architecture [46,47], which processes the input pulse shapes one pixel at a time so as to preserve the time-ordered structure of the learning problem. The RNN, composed here of a long-short-term memory (LSTM) unit cell [48], is performing the same set of operations at each time step in a recurrent fashion, using the new input together with a hidden state (h) to encode information about the context of that input. This vector allows the model to capture correlations within the input data and to potentially keep a memory of previous inputs. Given our quantum characterization problem where the model needs to output the quantum state at different times, we engineer a direct correspondence between the hidden state of the model h_n and the quantum state of the system ρ_n . We thus use an encoding layer, a small fully connected neural network, to map the initially prepared quantum state to the initial hidden state of the model, and use a similar decoding layer to map this hidden state back to the qubit state prediction $\tilde{\rho}(t)$.

To have a model that can efficiently train on a finite and realistic dataset, we have designed the model architecture to explicitly preserve most of the structure of the physical problem at hand. Nonetheless, the model is general enough to go significantly beyond the usual assumptions of a Markovian, single-mode description of the qubit dynamics. As demonstrated in Sec. IV, this choice is motivated by our objective of obtaining high-fidelity controls in scenarios where the physical description used in typical open-loop control approaches is insufficient. We emphasize that many other architecture choices could be directly used within our framework, such as a transformer model [49] or a parameterized master equation.

To train the model, we use a loss function principally composed of a mean-squared error (MSE) loss between the model predictions and the labels, as is common for regression tasks in supervised machine learning. Additionally,

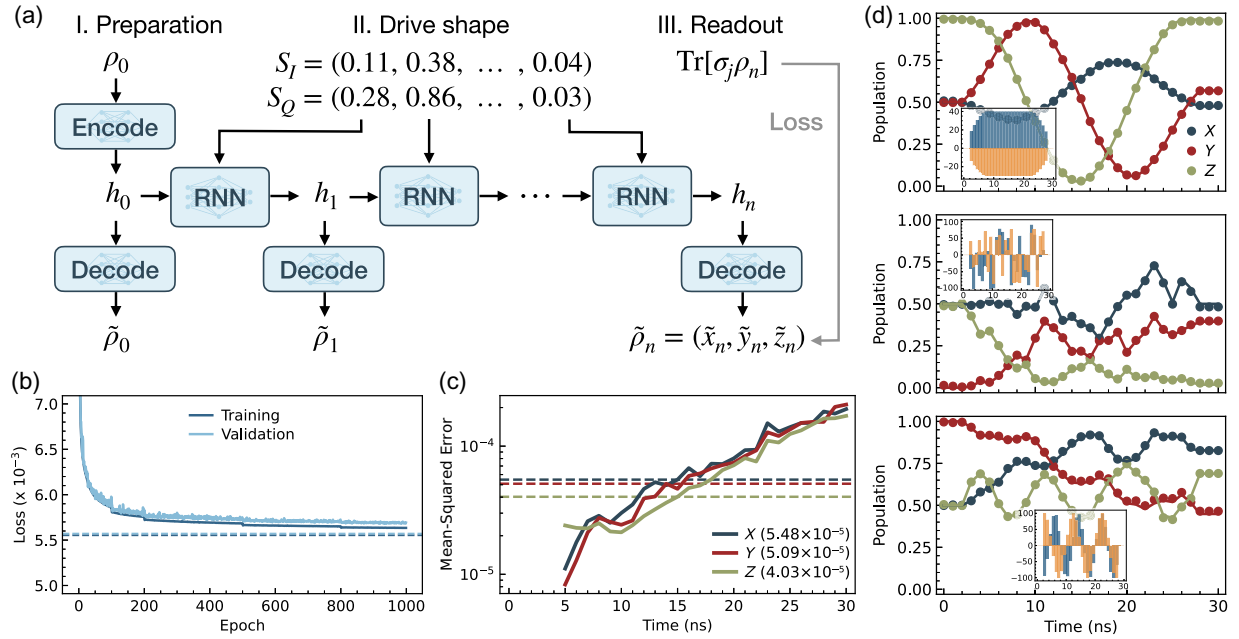


FIG. 2. Learning-driven qubit dynamics using supervised machine learning. (a) Model architecture. Preserving the structure of the three-step experiment, the model uses a small fully connected neural network (FNN; *Encode*) to map the nominally prepared quantum state (ρ_0) to the initial hidden state representation of the model (h_0). Each pixel of the input pulse shape is fed sequentially to a recurrent-neural-network cell (RNN). A second FNN (*Decode*) is used to produce the predicted quantum state $\tilde{\rho}_n$, here parameterized using Pauli expectation values. During training, the experimental readout outcomes are used as labels and the model parameters are adjusted by minimizing a loss function, here principally composed of a mean-squared error (MSE) between the labels and predictions. (b) Loss on the training and validation datasets during a supervised LSTM training. All hyperparameters and dataset characteristics are detailed in Appendix B. The dashed lines correspond to the sampling noise floor of the noisy labels for both datasets; see text for details. (c) Mean-squared error of the predicted quantum state expectation values as a function of pulse duration, averaged over the entire test dataset. Dashed lines correspond to the median over time, with values reported in the legend. The MSE is zero for the first 4 ns because we impose a zero padding of 2 ns at the beginning and end of every pulse. (d) Sample model predictions (circles) and true quantum trajectories (full lines) for three input pulse shapes unseen during training. The envelopes of the Gaussian flat top (top), random (middle), and sinusoidal (bottom) pulses are shown in the insets, with axes corresponding to drive amplitude in megahertz and time in nanoseconds. Note that, during training, the ML model never has access to the true quantum trajectories used here for evaluating model performance in panels (c) and (d), but only has access to the noisy labels obtained from $N_{\text{shots}} = 32$ projective measurements.

following Ref. [24], we expand the loss function with terms assuring that the RNN outputs are valid quantum states, i.e., that they are positive, and that the initial state predicted by the model corresponds to the known prepared state of the qubit. These physically motivated loss terms help regularize the model training, as it explores the parameter space of valid representations more efficiently [50]. An explicit expression for the full loss function, together with the hyperparameters used, can be found in Appendix B.

2. Quantum characterization results

A typical training of the RNN model is presented in Fig. 2(b) on a training dataset comprising 3.2 million pulse shapes of maximal duration of 30 ns, each measured for 32 shots. As demonstrated in Appendix A, similar performances can be reached using significantly fewer data. Both the training and validation MSE losses reach a value close to the sampling noise floor of about 5.6×10^{-3} , which

is obtained by computing the same MSE metric assuming perfect knowledge of the qubit expectation values, as given by the simulation data. This imprecision is significant, as it corresponds to an average distance between the noisy labels and the model output qubit state probabilities of about 7%. Such a high noise floor raises the question of whether the ML model is learning an accurate description of the quantum dynamics.

Leveraging the fact that we are working with simulation data, we can go beyond the experimentally available noisy labels and compare the model predictions directly to the ground-truth quantum trajectories obtained from simulation. In Fig. 2(d), we can qualitatively observe the high accuracy of the trained model in describing the qubit dynamics for various input pulse envelopes, as the model predictions (dots) match closely the true trajectories (full lines). This agreement is quantified in Fig. 2(c), showing a median MSE of about 5×10^{-5} across the three measurement axes and over different pulse lengths on the

previously unseen test dataset. This order-of-magnitude improved accuracy on the predicted outcome probabilities indicates that, remarkably, the ML model is able to use the noisy labels that one can obtain experimentally in order to learn the mapping from arbitrary pulse shapes to highly accurate quantum dynamics. The error associated with the model predictions is lowest at the initial time, where we have the most accurate information about the qubit state given the finite number of prepared states shared amongst all experiments. The finite model precision for evolving the quantum state for every pixel then leads to a linear increase in the prediction error over time, which is translated into a quadratic increase on the MSE in Fig. 2(c). This behavior is expected for any model with a finite accuracy, and is akin to evolving the state using the ME with slightly inaccurate model parameters, or to numerically solving a differential equation with a finite-precision solver.

The accuracy of the ML model can be significantly improved by extending the training dataset, for example, using more shots so as to construct more precise labels, and performing more experiments with different pulse shapes so as to explore more of the control space. As we will show next, the amount and quality of the training data used here are sufficient to obtain high-fidelity controls from the model. Additional results exploring how the quality and quantity of training data impact the learned representation and the performance of the resulting optimal controls are presented in Appendix A.

The neural-network model takes about an hour to train on a standard Nvidia RTX 3080 GPU. This time could be significantly reduced if needed for a specific implementation, for example, by pretraining the model on simulation data before training on the experimental measurements. This time scale is comparable to the experimental data acquisition time for building the datasets using transmon qubits, which is a few hours per million pulse shapes with 32 shots, including the overhead of pulse sequence uploading on the control electronics.

C. Quantum optimal control

Having demonstrated that an RNN can be used to accurately learn quantum dynamics from experimentally realistic data, we now present how one can successfully use that representation to perform quantum optimal control. We focus on single-qubit quantum gates and optimize the external controls so as to maximize the average gate fidelity of a given unitary U_{target} . As schematically illustrated in Fig. 1(b), the optimization procedure simply consists of making each pixel of the input pulse a free parameter and computing the evolution of the quantum system due to these controls using the trained ML model with fixed internal parameters. By evolving a set of initial quantum states forming a complex projective 2-design, one can compute the average gate fidelity of the arbitrary

black-box quantum channel [51,52], which is represented here by a neural network. For the case of one qubit, the six cardinal states used as preparations in our experiment form a complex projective 2-design [53]. Finally, an optimization algorithm is used to update the input controls so as to maximize the fidelity of the operation of interest.

Making use of the fact that the RNN is fully autodifferentiable, we employ a gradient-based approach and design the cost function to include a multitude of experimentally relevant constraints, without having to analytically derive an expression for their gradient. In addition to the cost associated with the average gate infidelity of the operation of interest, we use regularizing costs that yield smooth and low-amplitude controls, which are desirable in the experiment to mitigate crosstalk and pulse distortion effects [25,54]. In particular, we use a cost term proportional to the average absolute pulse amplitude to penalize for unnecessarily strong pulses, a cost for amplitudes $|\Omega|/2\pi > 100$ MHz, together with terms proportional to the first and second derivatives of the pulse shapes to find smooth controls with a limited frequency spectrum. We perform the optimizations using the Adam optimizer [55]. The full cost function and optimization parameters used here are presented in Appendix D.

A typical control optimization on the trained RNN is shown in Fig. 3(a) for the case of a 20-ns $R_X(\pi)$ transmon qubit gate. Benefiting from the fast forward and backward evaluation of the RNN, 30 randomly initialized pulses are optimized in parallel on a single GPU card in under a minute. This batched optimization and ability to perform a large number of iterations significantly mitigates the convergence issues that many GRAPE-like approaches face. As a result, we can use the trained model to obtain optimal controls for a wide variety of gates on a very short time scale, which can be used to create a universal gate set of optimal pulses or to compile continuously parameterized gates.

The optimal control found by our MLQOC approach yields a 99.994% average gate fidelity on the full five-level transmon model. The pulse shape and resulting qubit dynamics are illustrated in Figs. 3(b) and 3(c). Importantly, this optimal pulse is smooth and easily implementable in practice with limited-bandwidth electronics. The main sinusoidal oscillations seen in the optimal pulse correspond to the expected intermediate frequency of about $2\pi \times 100$ MHz. Deconvolving this intermediate frequency, we obtain a smooth envelope in the quadrature effectively driving σ_X (black line), together with a small orthogonal component (gray line) necessary to adjust the Z phase of the resulting unitary and to mitigate leakage to the $|2\rangle$ state [45]. As shown in Fig. 3(c), the pulse produces coherent dynamics on the qubit that resemble the one of standard sinusoidal and Gaussian control pulses, according to both the RNN model (dots) and the true master equation dynamics (lines).

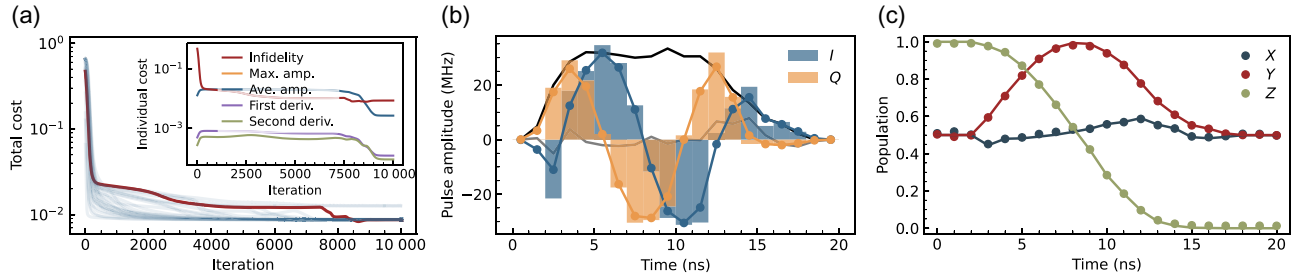


FIG. 3. Quantum optimal control of a 20-ns $R_X(\pi)$ gate on a trained RNN. The optimal MLQOC pulse fidelity is 99.994% on the five-level transmon. (a) Parallel optimization of 30 randomly initialized pulse shapes. The optimization of the best-performing pulse is highlighted in red, with the inset showing its decomposition into individual costs. (b) Resulting raw (bars) and Gaussian filtered (dots with line) optimal pulse shape for the two drive quadratures I and Q . The associated smooth envelope effectively driving σ_X (σ_Y) is shown in black (gray). (c) Predicted (dots) and true (lines) quantum state evolution when applying the optimal pulse shape on the initial state $|1\rangle$.

D. MLQOC performance

An important advantage of our approach is that the trained model is not tied to a specific set of gates and can be used to optimize for arbitrary quantum operations captured by the model inputs and outputs, here any single-qubit unitary. As a demonstration, we use the same trained RNN model presented in Fig. 2 to optimize for a variety of gates and compile the results in Table I. We first optimize for π and $\pm\pi/2$ rotations around the three axes of the Bloch sphere. This gate set, together with the identity operation I realized by applying no drive, generates the full single-qubit Clifford group. We obtain an average fidelity of about 99.99% for this gate set, demonstrating that a finite-precision model trained on realistic data can yield high-fidelity quantum operations. We have also optimized 100 unitaries sampled uniformly at random from the Haar measure [56], and obtained gates with similar performances. This result further demonstrates that the trained RNN representation is accurately capturing arbitrary dynamics on the qubit subspace of our transmon model such that it can be used directly for performing QOC.

TABLE I. Gate performance of the MLQOC approach, evaluated on the full five-level transmon. The trained model can be used to find arbitrary Clifford and Haar random unitaries with high fidelity. Reported fidelities are the average over the different axes (X , Y , and Z) for Clifford gates and the median over 100 uniformly sampled gates for the Haar random gates. The transmon coherence limits are 99.9975% and 99.9967% for 15- and 20-ns gates, respectively. Note that the gate durations include a 4-ns zero padding to avoid gate bleed-through.

SQ gate	Duration (ns)	Fidelity (%)
$R_{X,Y,Z}(\pi)$	20	99.993
$R_{X,Y,Z}(\pm\pi/2)$	15	99.983
Haar random	20	99.984

The fact that the trained ML model does not fully reach the coherence-limited average gate fidelities of $\sim 99.997\%$ can be attributed to the finite precision of the heuristic that the RNN learns to represent the quantum dynamics. As shown in Appendix A, reducing shot noise and increasing the training dataset size does not significantly improve the control fidelity results, and important stochasticity in the performance of similar models remains, even for trained models performing the quantum characterization task significantly better than the one presented in Fig. 2. Given that it is possible to train a model that would yield coherence-limited gates, such as a model approaching the master equation used to generate the data, it would be interesting to try refining the training data by sampling around the optimal pulses or to explore the performance of different machine-learning architectures. Despite not fully reaching the coherence limit, we show in the next section that our MLQOC approach can significantly outperform open-loop QOC approaches in practical scenarios by alleviating the problem of model bias.

IV. COMPARISON WITH OPEN-LOOP QOC

Having presented proof-of-principle results of our approach on transmon single-qubit gates, we now demonstrate the advantage of using MLQOC in realistic experimental settings where model bias will necessarily be present. In particular, we show that the MLQOC approach studied here achieves significantly better quantum gates than typical open-loop QOC methods under realistic model bias. This improvement is achieved by learning an accurate representation of the device dynamics from data, and using that model to design high-fidelity controls.

We consider model bias solely coming from out-of-model dynamics, i.e., effects that are present in the quantum device but not in the physical modeling used by the open-loop QOC approach. We take these dynamics to be unaccounted pulse distortions in the control lines, which produce discrepancies between the pulse shape the qubit

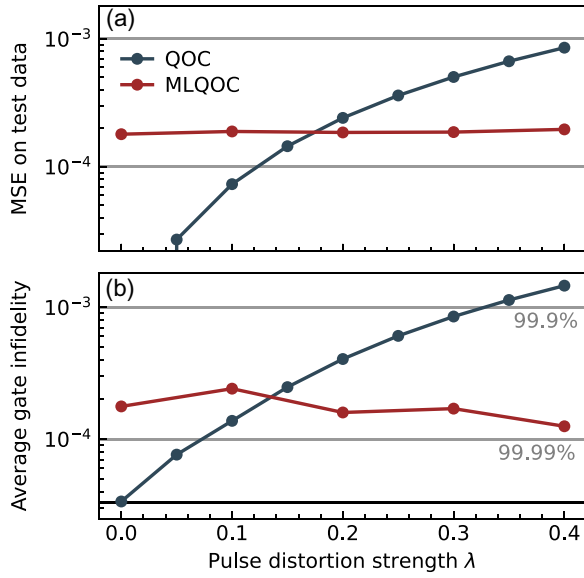


FIG. 4. Demonstrating the robustness of the proposed MLQOC approach under a realistic source of model bias in the form of random pulse distortions. (a) Predicting the quantum state dynamics of the test dataset using the master equation model (blue) and RNN models trained on experimentally available data containing 1 million pulse shapes, each sampled for 32 shots (red). (b) Performance of the 20-ns $R_X(\pi)$ gates found by optimizing on the models of panel (a), as evaluated on the full transmon model. The gate coherence limit is indicated by a black line.

receives and the one we model. Note that considering pulse distortions is simply a choice for demonstrating the performance of MLQOC in the presence of model bias, although this choice is motivated by realistic experimental scenarios with superconducting qubits [57–60]. The methodology used to generate the random pulse distortions is detailed in Appendix E.

In Fig. 4, we present the performance of the open-loop (blue) and machine-learning-based (red) QOC approaches in performing the two tasks identified within our framework, namely the quantum characterization as reported by the MSE in Fig. 4(a) and the optimal control as reported by the obtained average gate infidelity in Fig. 4(b), both as a function of the amplitude of the random pulse distortions λ . Scaling this amplitude λ directly translates into scaling the magnitude of the model bias. For instance, a distortion of $\lambda = 0.2$ corresponds to pulses deviating on average by only 3% from their original shape, which is less than 0.3 MHz. See Appendix E for more details on the definition of the parameter λ .

Since the model used for the open-loop QOC approach is the full master equation used to describe the transmon qubit, it perfectly describes the quantum system when no model bias is present ($\lambda = 0$), and achieves coherence-limited single-qubit gates. However, as we scale

the amplitude of the pulse distortions ($\lambda > 0$), the mean-squared error for predicting the true quantum states of our test dataset grows sharply. Consequently, the controls optimized using the ME model yields single-qubit gate fidelities that rapidly drop below 99.9% when amplifying the out-of-model dynamics as shown in Fig. 4(b).

In contrast, our proposed ML framework is designed to learn the entire transformation describing our control of the quantum system, from the input pulses of the AWG all the way to the measurement bits we extract, which includes pulse distortions as well as any other dynamics of the quantum device that can affect our measurements. As shown in Fig. 4(a), the ML model precision is then virtually unaffected by the added pulse distortions, simply because these dynamics are present in the data it learns from. Using these trained models in Fig. 4(b) then allows us to find optimal controls that remain close to 99.99% fidelity, even under important pulse distortions.

These results constitute a clear demonstration that, in a realistic scenario where model bias limits our ability to perform open-loop QOC, our approach of employing ML can lead to a considerable gain in quantum control fidelity. We note that, even in a well-characterized system where our physical model yields similar quantum characterization performance to the trained ML model, our MLQOC approach can still offer an advantage. This is the case here for $\lambda = 0.2$, where the MSE are similar for both approaches, see Fig. 4(a), but the resulting fidelity is superior for MLQOC, see Fig. 4(b). This is explained by the fact that the trained model can capture the full device dynamics and thus provide optimal pulses that account for spurious effects such as pulse distortions. As opposed to the physical ME model, the ML model is providing an effectively unbiased estimator of the true quantum dynamics, which for the same level of precision can yield better-performing controls in practice.

V. CONCLUSION

In this work, we have presented an experimentally simple two-step approach to quantum control aimed at successfully realizing optimal quantum operations in practice. We tackled the problem of model bias by learning a parameterized representation of the system dynamics directly from experimentally available data, before using that representation to find optimal controls. Using accurate transmon simulations, we have demonstrated that our machine-learning-based quantum optimal control approach can leverage a practical amount of experimental data to provide a precise characterization of the qubit dynamics and produce high-fidelity ($\sim 99.99\%$) controls for arbitrary single-qubit gates, thus going significantly beyond realizing a single operation with high fidelity.

The MLQOC framework promises to enable many quantum optimal control applications where model bias

is an important bottleneck, without necessitating any real-time feedback control or having to rely on data-intensive black-box optimizations of single operations. Beyond experimental ease of implementation, our approach is applicable to a wide range of quantum control problems because its modular features can be adapted to the specificities and requirements of a given quantum system. Importantly, our method of reducing the complexity of the learning task to only the quantum characterization, together with the use of powerful gradient-based QOC techniques, suggests it might be more data-efficient and more scalable to complex control problems than model-free approaches. It will be interesting to explore how these approaches perform as we expand towards optimally controlling multiqubit entangling and parallel operations.

ACKNOWLEDGMENTS

We thank Pierre Guilmin, Ronan Gauthier, and Boris Varbanov for insightful discussions. This work is supported by a collaboration between the U.S. Department of Energy and other agencies. This material is based upon work supported by the U.S. Department of Energy, Office of Science, National Quantum Information Science Research Centers, Quantum Systems Accelerator. Additional support is acknowledged from the Natural Sciences and Engineering Research Council, the Canada First Research Excellence Fund, and the Ministère de l'Économie et de l'Innovation du Québec.

DATA AVAILABILITY

The data that support the findings of this article are openly available [61].

APPENDIX A: DATA REQUIREMENTS

The success of our approach to quantum optimal control relies on constructing an accurate model of the quantum dynamics from data. We explore here how the amount and quality of training data impact our results. We change the accuracy of the data labels by numerically sampling more measurements (N_{shots}), from the experimentally inexpensive 32 shots (results of the main text), up to the limit of perfect labels. See Appendix C for a discussion on different approaches to sampling the pulse shapes we use for training.

In Fig. 5, we present the impact of the training dataset size and experiment repetitions (N_{shots}) used on both tasks considered in this work: the ML-based characterization of the quantum dynamics in Fig. 5(a) and the resulting quantum optimal control gate fidelity in Fig. 5(b). Unsurprisingly, using more training data leads to models with better prediction accuracy on the unseen test data. However, we show that such an improvement in the ML model

is not necessarily needed to obtain high-fidelity quantum controls. For example, we can get the MSE on the model prediction down by a factor greater than 2 (from 2.1×10^{-4} to 8.5×10^{-5}) using 3.2 million training pulse shapes instead of 1 million. However, the optimal controls found by both of these models lead to a single-qubit gate fidelity of about 99.99%, with most of the variability in the resulting fidelities explained by the stochastic nature of both the model training and the control optimization. Similarly, increasing the number of shots leads to significant improvements in the model prediction accuracy (about an order-of-magnitude reduction in the MSE at 3.2 million pulses between the three cases considered), but this improvement does not translate into significant improvements in the resulting control fidelities.

We explain this feature of our approach by the fact that, even though the ML model has a finite precision, which can be thought of as error bars on the model parameter estimates, as long as the learned parameterization does not lead to a significant bias in the quantum dynamics, the optimal controls on the finite precision model should also be close to optimal on the true quantum system. We can think of the QOC optimization on the trained neural network, which has a finite precision but very small bias, as adding Gaussian noise to the quantum state evolved using the master equation. The controls that maximize the gate fidelity on average using such a noisy model should also be

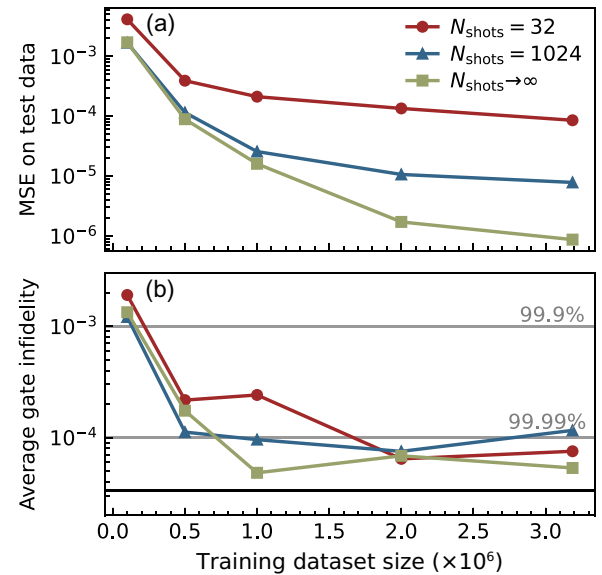


FIG. 5. Impact of the quantity and quality of training data on the model learning and optimal control performance. (a) Mean-squared prediction error on the unseen test dataset of RNN models trained on increasingly more pulse shapes with different sampling repetitions (N_{shots}). (b) Average gate infidelity of 20-ns $R_X(\pi)$ gate optimized on the models of panel (a), as evaluated on the true five-level transmon model. The black line is the coherence limit of the gate.

optimal on the true noiseless model. Here, we show that, as long as we have enough data to train a model with a mean-squared prediction error of less than about 2×10^{-4} , we can obtain quantum gates with 99.99% fidelity. Given the important time cost associated with acquiring large amounts of data on the quantum system, it is then desirable to use as few data as required for the MLQOC protocol to yield the desired control fidelities.

We emphasize that on the megahertz-scale repetition rate typical of quantum systems such as superconducting qubits, only a few hours are required to acquire the number of data considered here. Given that our framework can be easily extended to parallel single-qubit gates with virtually no additional experimental time, optimizing quantum operations using MLQOC is easily achievable with current experiments. Additionally, the limited data requirements we obtain for performing arbitrary single-qubit operations open the door to realistically using MLQOC for more complex control tasks such as two-qubit gates, readout, and reset operations, which is beyond the scope of this work.

APPENDIX B: MACHINE-LEARNING TRAINING

In this section, we present the implementation details of our machine-learning (ML) model training, whose architecture is illustrated in Fig. 2. Our numerical implementation is based on PyTorch [62]. The recurrent neural network we use is the vanilla long-short-term memory unit implemented in `torch.nn.lstm` with a single layer and a hidden size of 48. (Please refer to the PyTorch documentation for more details.) The *Encode* transformation is a depth-2 fully connected neural network (FCNN) with a hidden size of 96 and a sigmoid activation function, whereas the *Decode* transformation is a single-layer linear network, using again a sigmoid activation function to map the network output into proper probabilities $\in (0, 1)$. We emphasize that none of these hyperparameters were carefully optimized, as the objective in this work is to demonstrate how our proposed MLQOC framework can be successful with a simple implementation. Such a hyperparameter optimization, together with using more powerful neural-network architectures such as transformer models [49], could yield slightly better performance, but would not change any of our conclusions.

We train this model using gradient-descent based on the Adam optimizer [55] with a learning rate of 0.001. During training, we use an initial batch size of 256 and double it at epochs [100, 200, 500, 800] in order to reduce the stochastic noise in the optimal model parameters as training progresses. We use the following physics-inspired loss function to train our model [24]:

$$\mathcal{L} = \mathcal{L}_{\text{pred}} + w_{\text{posit}} \mathcal{L}_{\text{posit}} + w_{\text{prep}} \mathcal{L}_{\text{prep}}, \quad (\text{B1})$$

where

$$\mathcal{L}_{\text{pred}} = \frac{1}{N} \sum_{n=1}^N [(\Pi_n^j - \tilde{\Pi}_n^j)^2], \quad (\text{B2})$$

$$\mathcal{L}_{\text{posit}} = \frac{1}{N(N_t + 1)} \sum_{t=0}^{N_t} \sum_{n=1}^N \text{ReLU}(|\tilde{\rho}_{t,n}|^2 - 1), \quad (\text{B3})$$

$$\mathcal{L}_{\text{prep}} = \frac{1}{N} \sum_{n=1}^N |\tilde{\rho}_{0,n} - \rho_{0,n}|^2. \quad (\text{B4})$$

Here $\text{ReLU}(\cdot)$ is the rectified linear unit (see below), and $\tilde{\rho}_{t,n} = (\tilde{x}_{t,n}, \tilde{y}_{t,n}, \tilde{z}_{t,n})$ is the model prediction of the qubit states for the n th quantum trajectory at time index t , given N pulse shapes with N_t pixels. We use the hyperparameters $w_{\text{posit}} = w_{\text{prep}} = 1.0$.

The prediction loss $\mathcal{L}_{\text{pred}}$ is a simple mean-squared error between the labels and the model predictions for the qubit population in the $j \in \{X, Y, Z\}$ basis, given by $\tilde{\Pi}_n^j = (1 - \langle \sigma^j \rangle)/2 \in [0, 1]$. Here the labels are also the probabilities of measuring the qubit state in the -1 eigenstate of the operator σ^j , which are constructed from averaging the $N_{\text{shots}} = 32$ projective measurements for each input pulse shape. The MSE given by this loss term is the metric we care most about to obtain an accurate quantum characterization of the qubit dynamics. However, there is more information about the RNN output that we can use to quantify if it accurately describes our physical problem.

We leverage that information with the following two loss terms, which allow us to make our model training more accurate and more data-efficient. The positivity loss $\mathcal{L}_{\text{posit}}$ insures that the ML model predictions correspond to valid quantum states, i.e., that the predicted density matrices are positive. The rectified linear unit $\text{ReLU}(x) = \max(0, x)$ allows us to penalize only for nonphysical states and not for mixed state predictions, given that we are learning from data with relaxation and dephasing. Finally, the preparation loss $\mathcal{L}_{\text{prep}}$ makes the RNN predictions accurate at the beginning of the quantum trajectory, $t = 0$, by using our knowledge of the finite set of prepared initial states. When dealing with experimental data, or more generally with data containing state preparation and measurement errors, we can set the targeted prepared states $\rho_{0,n}$ to be consistent with our best estimate of these errors. For example, we can average over subsets of the data where the projective readout immediately follows the preparation, and effectively perform quantum state tomography of the six cardinal states of the Bloch sphere [63].

We note that the model architecture, together with the loss functions we use, can easily be extended to modeling the dynamics of n qubits or multilevel systems. For example, the input dimension of the RNN could be $2n$ for n microwave drives with two quadratures, and the prepared and output quantum states can be represented using

$4^n - 1$ expectation values for n qubits. This natural extension would be sufficient to apply our framework to optimize for two-qubit gates and parallel single-qubit gates. Of course, the data requirements and model complexity necessary for successfully applying MLQOC to controlling a quantum system with an exponentially large Hilbert space should also scale unfavorably with the number of qubits. However, using the heuristic representation of a machine-learning model might be beneficial for tackling such complex control problems.

APPENDIX C: PULSE DATASETS

The quantum characterization task is achieved by training a parameterized representation, such as a neural network acting as a universal function approximator, to describe the transformation from arbitrary input pulse shapes into the dynamics of the quantum systems under study. Given the enormous size of the input control parameter space, with every pixel taking an arbitrary floating-point value, and the fact that we want to acquire a finite dataset in a reasonable experimental time, it is natural to try uniformly sampling at random the input controls to form our ML datasets. This approach works well and the trained ML model can predict the dynamics for random pulse shapes accurately. However, with that choice the model performs poorly at predicting the dynamics of smooth pulses, which for example possess a very different frequency spectrum than random pulses (data not shown). Given that in an experimental scenario we want to use smooth control pulses, so as to respect the bandwidth limitations of the control electronics and avoid crosstalk problems arising from having signals with a wide frequency spectrum, having a trained model that is inaccurate at capturing the dynamics of smooth pulses is practically not useful.

To remedy the situation, we simply include smooth pulses in the training data. Here, we generate these smooth pulses using physically motivated envelopes that are commonly used to perform single-qubit gates. Specifically, in addition to uniformly sampled random envelopes, our set of pulses is composed of flat, Gaussian, Gaussian with a flat top, Gaussian with an orthogonal DRAG component, and sinusoidal pulses. Heuristically, we chose proportions of 25% of flat pulses with random amplitudes, 25% of flat-top Gaussian pulses with random widths, amplitudes, and standard deviations, 25% of uniformly sampled random pulses, 12.5% of Gaussian pulses with an orthogonal DRAG component with random amplitudes, and 12.5% of sinusoidal pulses with random amplitudes and frequencies. We generated a total 4.25 million pulses, and used 75% for the training set, 15% for the validation set used to avoid overfitting, and 10% for the test set used to evaluate and compare model performances.

Given our pulse parameterization at the level of the AWG before IQ mixing, our pulses should have a main sinusoidal oscillation corresponding to the intermediate frequency, here around 100 MHz, such that the resulting pulses reach the qubit frequency and produce nontrivial dynamics. We thus generate all of the aforementioned pulses at the envelope level, i.e., without any explicit oscillating component. We then convolve these envelopes with the intermediate-frequency oscillations, which result in the final pulses of the dataset. In order to explicitly sample the impact of detuned drives on the qubit dynamics, we randomly sample the intermediate frequency we use to convolve with our envelopes, using a Gaussian distribution centered at the nominal $\omega_{\text{IF}}/2\pi = 100$ MHz with a standard deviation of 1 MHz.

We note that fine-tuning the quality of the training dataset by using pulses that are known to achieve the target operations with good fidelity, and perhaps increasing the sampling (N_{shots}) for these pulses, could be beneficial to the performance of the MLQOC approach. However, as we have demonstrated in the main text, such fine-tuning is unnecessary and the approach works well as long as the pulses seen during training have the same characteristics as the controls we implement in the end. In that sense, although such an informed construction of the training dataset will necessarily bias the model learning towards a given set of dynamics, this bias should be beneficial for learning a good model efficiently, as long as the bias is consistent with the constraint we put on the following QOC optimizations. Indeed, it is desirable to have a trained model that is good at predicting the dynamics of smooth pulses, even if it is somewhat specialized and not fully generalizable, since ultimately this is what our optimal control task requires. This bias is enforced explicitly in this work by using a cost function minimizing the first and second derivatives of the optimal controls.

APPENDIX D: QOC IMPLEMENTATION DETAILS

In this section, we present the implementation details of our quantum optimal control approach illustrated in Fig. 1(b). We use an open-loop optimization where the pulses are parameterized at every pixel, the trained ML model with fixed parameters is used to compute the dynamics resulting from the pulses, and the cost function is expressed as follows [25]:

$$\mathcal{C} = \mathcal{C}_{\text{fidel}} + w_{\text{clamp}}\mathcal{C}_{\text{clamp}} + w_{\text{mean}}\mathcal{C}_{\text{mean}} + w_{\text{first}}\mathcal{C}_{\text{first}} + w_{\text{second}}\mathcal{C}_{\text{second}}, \quad (\text{D1})$$

where

$$\mathcal{C}_{\text{fidel}} = \frac{1}{N} \sum_{n=1}^N 1 - \text{AGF}(\rho_{n,T}), \quad (\text{D2})$$

$$\mathcal{C}_{\text{clamp}} = \frac{1}{N(N_t + 1)} \sum_{t=0}^{N_t} \sum_{n=1}^N \text{ReLU}(\Omega_{n,t} - \Omega_{\text{max}}), \quad (\text{D3})$$

$$\mathcal{C}_{\text{mean}} = \frac{1}{N(N_t + 1)} \sum_{t=0}^{N_t} \sum_{n=1}^N |\Omega_{n,t}|^2, \quad (\text{D4})$$

$$\mathcal{C}_{\text{first}} = \frac{1}{NN_t} \sum_{t=1}^{N_t} \sum_{n=1}^N |\partial_t \Omega_{n,t}|^2, \quad (\text{D5})$$

$$\mathcal{C}_{\text{second}} = \frac{1}{N(N_t - 1)} \sum_{t=2}^{N_t} \sum_{n=1}^N |\partial_t^2 \Omega_{n,t}|^2. \quad (\text{D6})$$

Here Ω_{max} is the maximal allowed drive amplitude, which is $2\pi \times 100$ MHz in this work. This constitutes a reasonable choice to avoid significantly amplifying the effect of classical crosstalk on a transmon chip, as typical sinusoidal 20-ns single-qubit π gates require $|\Omega|/2\pi \approx 31$ MHz.

The partial time derivatives in the smoothing cost functions are implemented as finite differences numerically with $\partial_t \Omega_{n,t} = \Omega_{n,t} - \Omega_{n,t-1}$, and the average gate fidelity is computed as [64]

$$\text{AGF}(\rho_{n,T}) = \frac{\sum_j \text{Tr}(U_{\text{target}} \rho^j U_{\text{target}}^\dagger \text{RNN}(\rho^j)) + d^2}{d^2(d+1)}, \quad (\text{D7})$$

where $d = 2$ for a qubit, T is the final time index, σ^j are the six cardinal states (our chosen complex projective 2-design), and $\text{RNN}(\rho^j)$ represents the transformation realized by our trained ML model, which acts here in place of the usual quantum channel.

As mentioned in the main text, we optimize for a batch of 30 pulses in parallel on the same GPU card in under a minute, and select the best-performing pulse on the full cost function as the optimal control. We use the Adam optimizer with a learning rate of 0.001 to perform the gradient descent [55]. As in the ML model training, all the gradients are computed numerically using the autodifferentiable feature of PyTorch [62].

We have observed that, whereas the clamping cost $\mathcal{C}_{\text{clamp}}$ simply allows the pulses to respect an experimental constraint, the average amplitude cost $\mathcal{C}_{\text{mean}}$ together with the smoothing cost functions $\mathcal{C}_{\text{first}}$ and $\mathcal{C}_{\text{second}}$ are very important for constraining the parameter exploration during the optimization towards controls where the trained RNN has an accurate representation of the dynamics. Indeed, the trained model predicts dynamics which are very close to the true master equation dynamics for smooth input pulses, which allows us to find optimal controls with over 99.99% fidelity when using these regularizing cost functions. However, optimizing the controls on the trained model using only the fidelity cost can lead to pulses where the predicted and true dynamics diverge significantly. This problem can

be completely avoided using the smoothing cost functions, and if necessary a simple postselection based on the same smoothness criteria over the optimization results of the batch. We optimize our pulses for all the single-qubit gate results presented in the main text using $w_{\text{clamp}} = 10.0$, $w_{\text{mean}} = 0.1$, and $w_{\text{first}} = w_{\text{second}} = 0.01$. We have observed that fine-tuning these hyperparameters does not lead to significant changes in the resulting optimal gate fidelities.

APPENDIX E: RANDOM PULSE DISTORTIONS

We simulate pulse distortions numerically by applying a fixed causal transfer function to the pulses. We first generate the random transfer function $F(\omega)$ in the frequency domain, using the noise parameter λ that we scale from 0 to 0.4 in Fig. 4. We add the noise to the trivial flat transfer function such that

$$F(\omega \geq 0) = 1 + U(-\lambda, \lambda), \quad (\text{E1})$$

where $U(\text{min}, \text{max})$ is the uniform distribution, and we replicate the values for the negative frequencies such that $F(\omega)$ is even. To obtain a smooth transfer function, we sample 11 points from the uniform distribution, which we attribute to frequencies up to 600 MHz, before using a Gaussian filter with a 2.5-GHz bandwidth to interpolate between these values for 1001 points in the range $[-1.5, 1.5]$ GHz.

Instead of using Fourier transforms to convolve all the pulses of our datasets with the noisy transfer function defined in the frequency domain, we express $F(\omega)$ in the time domain using a transfer matrix which can be directly applied to the pulses using a simple matrix multiplication. This transfer matrix is obtained by numerically solving [7]

$$T_{jk} = \int_{-\infty}^{\infty} F(\omega) \frac{\sin(\omega \Delta t / 2) \cos(\omega(j-k) \Delta t)}{\pi \omega} d\omega, \quad (\text{E2})$$

where Δt is the pixel size (in units of time) and j and k are the pixel indices.

Finally, given that this noise models a physical process, we impose causality by setting the lower triangular elements to zero and renormalizing it. Using this approach, we can simulate a realistic scenario where out-of-model dynamics are present in the quantum system we are trying to control, and we can have a single tunable parameter λ to quantify the resulting model bias.

APPENDIX F: IQ MIXING AND QUBIT DRIVE

In this section, we detail the IQ mixing transformation applied to the input microwave pulses, typically generated by an arbitrary waveform generator, before reaching the

transmon qubit. We then show how these mixed pulses can be viewed as the usual σ_X and σ_Y drive Hamiltonian prefactors. As detailed in Refs. [41,42], the two AWG signals S_I and S_Q get mixed with a local oscillator signal of frequency ω_{LO} in a process known as sideband mixing to produce the drive:

$$\Omega(t) = S_I(t) \cos(\omega_{LO}t) + S_Q(t) \sin(\omega_{LO}t). \quad (F1)$$

We then define the AWG signals with an envelope combined with sinusoidal oscillations at an intermediate frequency ω_{IF} and a phase ϕ , such that

$$S_I(t) = I(t) \cos(\omega_{IF}t + \phi), \quad (F2)$$

$$S_Q(t) = -Q(t) \sin(\omega_{IF}t + \phi). \quad (F3)$$

We then directly get

$$\begin{aligned} \Omega(t) = & \frac{I(t)}{2} [\cos(\omega_+t + \phi) + \cos(\omega_-t + \phi)] \\ & - \frac{Q(t)}{2} [\cos(\omega_-t + \phi) - \cos(\omega_+t + \phi)], \end{aligned} \quad (F4)$$

where $\omega_{\pm} = \omega_{IF} \pm \omega_{LO}$. We thus see that, by setting $I(t) = Q(t) = A(t)$, we obtain a single carrier frequency signal at the upper sideband ω_+ ,

$$\Omega(t) = A \cos(\omega_+t + \phi). \quad (F5)$$

Using an LO close to the qubit frequency such that $\omega_{IF} + \omega_{LO} \approx \omega_q$, we can then precisely control the frequency spectrum of the qubit pulses we send, with a limitation set by the AWG bandwidth and potentially the cable filters used in the experiment. Note that one can also choose a different phase between the S_I and S_Q signals to drive the lower sideband.

We now demonstrate that such a drive can be used to perform arbitrary qubit gates. Starting from Eq. (2), we can introduce the creation and annihilation operators to diagonalize the quadratic terms of the transmon Hamiltonian and obtain, after expanding the $\cos \hat{\phi}$ term and truncating after the second order [39]:

$$\hat{H}(t) \approx \omega_q \hat{b}^\dagger \hat{b} - \frac{E_C}{2} \hat{b}^\dagger \hat{b}^\dagger \hat{b} \hat{b} + \Omega(t) \frac{i}{2} \left(\frac{2E_C}{E_J} \right)^{1/4} (\hat{b}^\dagger - \hat{b}). \quad (F6)$$

Going into the qubit rotating frame with the transformation $\hat{U}(t) = \exp(i\omega_q \hat{b}^\dagger \hat{b}t)$ and applying the rotating-wave

approximation, we obtain the time-dependent drive Hamiltonian

$$\hat{H}_d(t) \approx \frac{i}{2} \left(\frac{2E_C}{E_J} \right)^{1/4} \frac{A(t)}{2} (\hat{b}^\dagger e^{i(\Delta t + \phi)} - \hat{b} e^{-i(\Delta t + \phi)}), \quad (F7)$$

where $\Delta = \omega_+ - \omega_q$. Performing a two-level truncation with $\hat{b}^\dagger \rightarrow \sigma_+$, $\hat{b} \rightarrow \sigma_-$, and $\sigma_{\pm} = (\sigma_X \pm \sigma_Y)/2$, redefining the prefactor in front of the last parentheses to be $\tilde{A}(t)$, and assuming that the drive is on resonance with the qubit ($\Delta = 0$), we finally get the textbook single-qubit drive Hamiltonian

$$\hat{H}_d^{\text{qubit}}(t) = \tilde{A}(t) [\cos(\phi)\sigma_X - \sin(\phi)\sigma_Y]. \quad (F8)$$

We thus understand how controlling the pixel amplitudes of the AWG inputs $S_I(t)$ and $S_Q(t)$ allows us to perform arbitrary single-qubit operations.

-
- [1] A. P. Peirce, M. A. Dahleh, and H. Rabitz, Optimal control of quantum-mechanical systems: Existence, numerical approximation, and applications, *Phys. Rev. A* **37**, 4950 (1988).
 - [2] J. Werschnik and E. Gross, Quantum optimal control theory, *J. Phys. B: At., Mol. Opt. Phys.* **40**, R175 (2007).
 - [3] C. P. Koch, U. Boscain, T. Calarco, G. Dirr, S. Filipp, S. J. Glaser, R. Kosloff, S. Montangero, T. Schulte-Herbrüggen, D. Sugny, and F. K. Wilhelm, Quantum optimal control in quantum technologies. Strategic report on current status, visions and goals for research in Europe, *EPJ Quantum Technol.* **9**, 19 (2022).
 - [4] 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).
 - [5] V. Krotov, *Global Methods in Optimal Control Theory* (CRC Press, Boca Raton, Florida, 1995).
 - [6] U. Boscain, M. Sigalotti, and D. Sugny, Introduction to the Pontryagin maximum principle for quantum optimal control, *PRX Quantum* **2**, 030203 (2021).
 - [7] F. Motzoi, J. M. Gambetta, S. T. Merkel, and F. K. Wilhelm, Optimal control methods for rapidly time-varying Hamiltonians, *Phys. Rev. A* **84**, 022307 (2011).
 - [8] 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).
 - [9] K. Rojan, D. M. Reich, I. Dotsenko, J.-M. Raimond, C. P. Koch, and G. Morigi, Arbitrary-quantum-state preparation of a harmonic oscillator via optimal control, *Phys. Rev. A* **90**, 023824 (2014).
 - [10] M. H. Goerz, F. Motzoi, K. B. Whaley, and C. P. Koch, Charting the circuit QED design landscape using optimal control theory, *npj Quantum Inf.* **3**, 1 (2017).
 - [11] L. S. Theis, F. Motzoi, and F. K. Wilhelm, Simultaneous gates in frequency-crowded multilevel systems using fast,

- robust, analytic control shapes, *Phys. Rev. A* **93**, 012324 (2016).
- [12] J. H. Wesenberg, Designing robust gate implementations for quantum-information processing, *Phys. Rev. A* **69**, 042323 (2004).
- [13] X. Wang, L. S. Bishop, J. Kestner, E. Barnes, K. Sun, and S. Das Sarma, Composite pulses for robust universal control of singlet–triplet qubits, *Nat. Commun.* **3**, 997 (2012).
- [14] D. Buterakos, S. Das Sarma, and E. Barnes, Geometrical formalism for dynamically corrected gates in multiqubit systems, *PRX Quantum* **2**, 010341 (2021).
- [15] A. Koswara, V. Bhutoria, and R. Chakrabarti, Quantum robust control theory for Hamiltonian and control field uncertainty, *New J. Phys.* **23**, 063046 (2021).
- [16] M. Bukov, A. G.-R. Day, D. Sels, P. Weinberg, A. Polkovnikov, and P. Mehta, Reinforcement learning in different phases of quantum control, *Phys. Rev. X* **8**, 031086 (2018).
- [17] S. Borah, B. Sarma, M. Kewming, G. J. Milburn, and J. Twamley, Measurement-based feedback quantum control with deep reinforcement learning for a double-well nonlinear potential, *Phys. Rev. Lett.* **127**, 190403 (2021).
- [18] V. V. Sivak, A. Eickbusch, H. Liu, B. Royer, I. Tsioutsios, and M. H. Devoret, Model-free quantum control with reinforcement learning, *Phys. Rev. X* **12**, 011059 (2022).
- [19] R. Porotti, A. Essig, B. Huard, and F. Marquardt, Deep reinforcement learning for quantum state preparation with weak nonlinear measurements, *Quantum* **6**, 747 (2022).
- [20] R. Porotti, V. Peano, and F. Marquardt, Gradient-ascent pulse engineering with feedback, *PRX Quantum* **4**, 030305 (2023).
- [21] Y. Baum, M. Amico, S. Howell, M. Hush, M. Liuzzi, P. Mundada, T. Merkh, A. R. Carvalho, and M. J. Biercuk, Experimental deep reinforcement learning for error-robust gate-set design on a superconducting quantum computer, *PRX Quantum* **2**, 040324 (2021).
- [22] V. V. Sivak, A. Eickbusch, B. Royer, S. Singh, I. Tsioutsios, S. Ganjam, A. Miano, B. L. Brock, A. Z. Ding, L. Frunzio, S. M. Girvin, R. J. Schoelkopf, and M. H. Devoret, Real-time quantum error correction beyond break-even, *Nature* **616**, 50 (2023).
- [23] L. Ding, M. Hays, Y. Sung, B. Kannan, J. An, A. Di Paolo, A. H. Karamlou, T. M. Hazard, K. Azar, D. K. Kim, B. M. Niedzielski, A. Melville, M. E. Schwartz, J. L. Yoder, T. P. Orlando, S. Gustavsson, J. A. Grover, K. Serniak, and W. D. Oliver, High-fidelity, frequency-flexible two-qubit fluxonium gates with a transmon coupler, *Phys. Rev. X* **13**, 031035 (2023).
- [24] É. Genois, J. A. Gross, A. Di Paolo, N. J. Stevenson, G. Koolstra, A. Hashim, I. Siddiqi, and A. Blais, Quantum-tailored machine-learning characterization of a superconducting qubit, *PRX Quantum* **2**, 040355 (2021).
- [25] 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).
- [26] J. Nocedal and S. Wright, *Numerical Optimization*, Springer Series in Operations Research and Financial Engineering (Springer New York, New York, 2006).
- [27] S. Machnes, E. Assémat, D. J. Tannor, and F. K. Wilhelm, Gradient optimization of analytic controls: The route to high accuracy quantum optimal control, *Phys. Rev. Lett.* **120**, 150401 (2018).
- [28] A. Youssry, Y. Yang, R. J. Chapman, B. Haylock, F. Lenzini, M. Lobino, and A. Peruzzo, Experimental graybox quantum system identification and control, *npj Quantum Inf.* **10**, 1 (2024).
- [29] S. Krastanov, K. Head-Marsden, S. Zhou, S. T. Flammia, L. Jiang, and P. Narang, Unboxing quantum black box models: Learning non-Markovian dynamics, *arXiv:2009.03902*.
- [30] P. Albertos and A. Sala Eds., *Iterative Identification and Control* (Springer, London, 2002).
- [31] 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).
- [32] S. Krastanov, S. Zhou, S. T. Flammia, and L. Jiang, Stochastic estimation of dynamical variables, *Quantum Sci. Technol.* **4**, 035003 (2019).
- [33] Q.-M. Chen, X. Yang, C. Arenz, R.-B. Wu, X. Peng, I. Pelczer, and H. Rabitz, Combining the synergistic control capabilities of modeling and experiments: Illustration of finding a minimum-time quantum objective, *Phys. Rev. A* **101**, 032313 (2020).
- [34] N. Wittler, F. Roy, K. Pack, M. Werninghaus, A. S. Roy, D. J. Egger, S. Filipp, F. K. Wilhelm, and S. Machnes, Integrated tool set for control, calibration, and characterization of quantum devices applied to superconducting qubits, *Phys. Rev. Appl.* **15**, 034080 (2021).
- [35] H. Ball, M. J. Biercuk, A. R. R. Carvalho, J. Chen, M. Hush, L. A. D. Castro, L. Li, P. J. Liebermann, H. J. Slatyer, C. Edmunds, V. Frey, C. Hempel, and A. Milne, Software tools for quantum control: Improving quantum computer performance through noise and error suppression, *Quantum Sci. Technol.* **6**, 044011 (2021).
- [36] A. Fyrrillas, O. Faure, N. Maring, J. Senellart, and N. Belabas, Scalable machine learning-assisted clear-box characterization for optimally controlled photonic circuits, *arXiv:2310.15349*.
- [37] J. Koch, T. M. Yu, J. Gambetta, A. A. Houck, D. I. Schuster, J. Majer, A. Blais, M. H. Devoret, S. M. Girvin, and R. J. Schoelkopf, Charge-insensitive qubit design derived from the Cooper pair box, *Phys. Rev. A* **76**, 042319 (2007).
- [38] 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).
- [39] A. Blais, A. L. Grimsmo, S. M. Girvin, and A. Wallraff, Circuit quantum electrodynamics, *Rev. Mod. Phys.* **93**, 025005 (2021).
- [40] R. Shillito, J. A. Gross, A. Di Paolo, É. Genois, and A. Blais, Fast and differentiable simulation of driven quantum systems, *Phys. Rev. Res.* **3**, 033266 (2021).
- [41] P. Krantz, M. Kjaergaard, F. Yan, T. P. Orlando, S. Gustavsson, and W. D. Oliver, A quantum engineer’s guide to superconducting qubits, *Appl. Phys. Rev.* **6**, 021318 (2019).
- [42] M. Baur, *Realizing quantum gates and algorithms with three superconducting qubits*, Ph.D. thesis, ETH Zurich, 2012.

- [43] A. Hashim, R. K. Naik, A. Morvan, J.-L. Ville, B. Mitchell, J. M. Kreikebaum, M. Davis, E. Smith, C. Iancu, K. P. O'Brien, I. Hincks, J. J. Wallman, J. Emerson, and I. Siddiqi, Randomized compiling for scalable quantum computing on a noisy superconducting quantum processor, *Phys. Rev. X* **11**, 041039 (2021).
- [44] P. Guilmin, R. Gautier, A. Bocquet, and É. Genois, Dynamiqs: An open-source Python library for GPU-accelerated and differentiable simulation of quantum systems (2024), dynamiqs.org.
- [45] 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).
- [46] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning* (MIT Press, Cambridge, Massachusetts, 2016). <http://www.deeplearningbook.org>.
- [47] Z. C. Lipton, J. Berkowitz, and C. Elkan, A critical review of recurrent neural networks for sequence learning, [arXiv:1506.00019](https://arxiv.org/abs/1506.00019).
- [48] S. Hochreiter and J. Schmidhuber, Long short-term memory, *Neural Comput.* **9**, 1735 (1997).
- [49] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, in *Advances in Neural Information Processing Systems*, Vol. 30 (Neural Information Processing Systems Foundation, Inc. (NeurIPS), Long Beach, California, 2017).
- [50] M. Raissi, P. Perdikaris, and G. Karniadakis, Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, *J. Comput. Phys.* **378**, 686 (2019).
- [51] C. Dankert, R. Cleve, J. Emerson, and E. Livine, Exact and approximate unitary 2-designs and their application to fidelity estimation, *Phys. Rev. A* **80**, 012304 (2009).
- [52] D. M. Debroy, É. Genois, J. A. Gross, W. Mruczkiewicz, K. Lee, S. Hong, Z. Chen, V. Smelyanskiy, and Z. Jiang, Context-aware fidelity estimation, *Phys. Rev. Res.* **5**, 043202 (2023).
- [53] More generally, a set of 4^n states are required to form a complex projective 2-design on n qubits.
- [54] R. W. Heeres, P. Reinhold, N. Ofek, L. Frunzio, L. Jiang, M. H. Devoret, and R. J. Schoelkopf, Implementing a universal gate set on a logical qubit encoded in an oscillator, *Nat. Commun.* **8**, 94 (2017).
- [55] D. P. Kingma and J. Ba, Adam: A method for stochastic optimization, [arXiv:1412.6980](https://arxiv.org/abs/1412.6980).
- [56] F. Mezzadri, How to generate random matrices from the classical compact groups, *Notices Am. Math. Soc.* **54**, 592 (2007).
- [57] S. Gustavsson, O. Zwiernik, 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).
- [58] 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).
- [59] S. Lazăr, Q. Ficheux, J. Herrmann, A. Remm, N. Lacroix, C. Hellings, F. Swiadek, D. C. Zanuz, G. J. Norris, M. B. Panah, A. Flasby, M. Kerschbaum, J.-C. Besse, C. Eichler, and A. Wallraff, Calibration of drive nonlinearity for arbitrary-angle single-qubit gates using error amplification, *Phys. Rev. Appl.* **20**, 024036 (2023).
- [60] E. Hyypä, A. Vepsäläinen, M. Papič, C. F. Chan, S. Inel, A. Landra, W. Liu, J. Luus, F. Marxer, C. Ockeloen-Korppi, S. Orbell, B. Tarasinski, and J. Heinsoo, Reducing leakage of single-qubit gates for superconducting quantum processors using analytical control pulse envelopes, [arXiv:2402.17757](https://arxiv.org/abs/2402.17757).
- [61] Python code used to generate the simulation data, define and train the machine-learning models, and perform quantum optimal control on them is available at: <https://zenodo.org/records/16888062>.
- [62] A. Paszke *et al.*, in *Advances in Neural Information Processing Systems*, Vol. 32 (Neural Information Processing Systems Foundation, Inc., Vancouver, Canada, 2019).
- [63] M. Paris and J. Rehacek, *Quantum State Estimation*, Lecture Notes in Physics (Springer, Berlin, Heidelberg, 2004).
- [64] M. A. Nielsen, A simple formula for the average gate fidelity of a quantum dynamical operation, *Phys. Lett. A* **303**, 249 (2002).