


Error-Robust Quantum Logic Optimization Using a Cloud Quantum Computer Interface

Andre R. R. Carvalho¹,* Harrison Ball, Michael J. Biercuk, Michael R. Hush, and Felix Thomsen²
Q-CTRL, Sydney, NSW, Australia, and Los Angeles, California, USA

 (Received 4 November 2020; revised 22 April 2021; accepted 20 May 2021; published 22 June 2021)

We describe an experimental effort designing and deploying error-robust single-qubit operations using a cloud-based quantum computer and analog-layer programming access. We design numerically optimized pulses that implement target operations and exhibit robustness to various error processes including dephasing noise, instabilities in control amplitudes, and crosstalk. Pulse optimization is performed using a flexible optimization package incorporating a device model and physically relevant constraints (e.g., bandwidth limits on the transmission lines of the dilution refrigerator housing IBM Quantum hardware). We present techniques for conversion and calibration of physical Hamiltonian definitions to pulse waveforms programmed via Qiskit Pulse and compare performance against hardware default derivative removal by adiabatic gate (DRAG) pulses on a five-qubit device. Experimental measurements reveal default DRAG pulses exhibit coherent errors an order of magnitude larger than tabulated randomized-benchmarking measurements; solutions designed to be robust against these errors outperform hardware-default pulses for all qubits across multiple metrics. Experimental measurements demonstrate performance enhancements up to: about a 10 times single-qubit gate coherent-error reduction; about a 5 times average coherent-error reduction across a five-qubit system; about a 10 times increase in calibration window to one week of valid pulse calibration; about a 12 times reduction gate-error variability across qubits and over time; and up to about a 9 times reduction in single-qubit gate error (including crosstalk) in the presence of fully parallelized operations. Randomized benchmarking reveals error rates for Clifford gates constructed from optimized pulses consistent with tabulated T_1 limits, and demonstrates a narrowing of the distribution of outcomes over randomizations associated with suppression of coherent errors.

DOI: [10.1103/PhysRevApplied.15.064054](https://doi.org/10.1103/PhysRevApplied.15.064054)

I. INTRODUCTION

Quantum computers are growing in complexity, capability, and utility across sectors, realizing major scientific accomplishments in the field of advanced computing [1] despite the fact that their performance remains critically limited by hardware imperfections and instabilities [2]. Superconducting qubits are a leading hardware candidate for the realization of large-scale quantum computers [3–5], and have enabled a variety of cloud-based systems accessible to the public [6,7].

Until recently, cloud access to such quantum computing hardware has been limited to provision of a default gate set [6–9]—implemented as fixed, predefined control waveforms—as is appropriate for end users unfamiliar with the details of the underlying physical hardware. IBM Qiskit Pulse [10] is an early attempt to provide analog-layer access to the programming of quantum computer hardware over a cloud application programming interface (API). Instead of only defining circuits composed of precalibrated

operations, analog-layer access gives users the ability to directly program the time-domain modulation of control waveforms that drive quantum logical operations. IBM Qiskit Pulse provides functionality to modulate the local oscillator (LO) frequency, phase, and amplitude envelope of microwave signals with full control over timing and sequencing (we refer to the resulting objects as “pulses”). In this approach users may define custom operations and even implement new modes of hardware operation beyond the default gate model [11].

Accessing this form of analog-layer programming enables the incorporation of low-level quantum control as a strategy to improve the performance of quantum computing hardware. Open-loop dynamic error suppression provides a validated approach to deterministically mitigate errors in individual hardware elements and individual qubit operations [12–18], complementing the considerably more demanding task of quantum error correction [19–22]. This framework permits the implementation of error-robust quantum logic operations by exploiting symmetries in the mathematical space of quantum-logic operations and the ubiquitous presence of temporal and spatial noise correlations in real laboratory environments [23].

*andre.carvalho@q-ctrl.com

To date, error-robust pulses have been of practical use only in systems for which relatively simple Hamiltonian models accurately describe the controlled quantum dynamics—notably ion traps [16] and historically nuclear magnetic resonance systems [24]. For these systems, optimized pulses can immediately be applied in experiment to yield reliable and high-performance control. In contrast, applying such control methods to superconducting qubits has been a long outstanding problem due to effects that are hard to accurately reproduce in simulation, including instrument noise, bandlimited control lines, additional modes, and coupling to unwanted quantum systems. Standard optimal control techniques such as GRAPE (gradient ascent pulse engineering) [25] have been applied to single-qubit operations for superconducting systems [26]; however, this approach cannot produce controls that are robust to these model inaccuracies or noise processes, and therefore only works for short time periods and fails to work in multiqubit operations where the model is not completely characterized [27]. Earlier works on robust control with superconducting qubits were theoretical [28,29] and did not recognize the critical importance of including experimental bandwidth limits.

In this manuscript we demonstrate a model-based approach to error-robust pulse optimization that employs two critical steps [18]. (1) We use graph-based computational methods that efficiently automate the differentiation process used in optimization, and allow incorporation of bandwidth limits not possible through standard GRAPE. (2) We optimize for robust control as opposed to optimal control employing a multidimensional filter function framework to capture noise susceptibility. Using this framework, we employ analog-layer programming on superconducting cloud quantum computer hardware to implement and test an error-robust single-qubit gate set.

Analog pulse waveforms are numerically optimized to enact gates resilient against dephasing, control-amplitude fluctuations, and crosstalk. For this, we have developed a versatile optimization engine based on a graphics processing unit (GPU)-compatible graph architecture coded in TensorFlow [18,30], an open-source artificial intelligence library designed for creating large-scale neural networks with many layers using data flow graphs. We perform experiments on IBM Quantum hardware backends *Valencia* and *Armonk* and program in the IBM Qiskit Pulse API [31], describing protocols for calibration of the programmable control channels, as well as tuneup of the optimized pulses to account for small distortions. Our results consistently reveal that pulses optimized to be robust against amplitude or dephasing errors, and including both a 30-MHz-bandwidth sinc-smoothing function and temporal discretization to match hardware programming, outperform default calibrated DRAG (derivative removal by adiabatic gate) [32,33] operations under native noise conditions.

We use an analytic framework based on previously published error models to quantitatively analyze pulse performance in the presence of both coherent and incoherent errors [34]. Experimental measurements show good agreement with a model incorporating both coherent rotation errors and an incoherent T_1 -type decay. Using optimized pulses, extracted measures of the effective error per gate demonstrate improvements up to and exceeding an order of magnitude in single-qubit coherent-error rates, with average device-wide performance improvements of about 5 times. The same optimized pulses reduce gate-error variability across qubits and over time by an order of magnitude; we demonstrate homogenization of qubit performance in a narrow band around a mean error better than the best-performing default pulse, and extend the useful calibration window to approximately one week as compared with the daily calibration cycle required for default pulses. These experiments are cross-validated with Clifford randomized benchmarking where we observe that optimized pulses outperform default pulses by a narrower margin due to the insensitivity of randomized benchmarking to certain coherent errors [35,36]. Optimized pulses exhibit key signatures of coherent-error suppression via narrowing of the randomized-benchmarking distribution over sequences. Independent quantitative analyses suggest error-robust optimized pulses exhibit T_1 -limited performance, and extracted incoherent-error magnitudes are consistent with expected errors using tabulated T_1 times as reported from the hardware backend.

Finally, we demonstrate that, for parallel gate operations (all qubits on a chip driven simultaneously), error-robust optimized pulses exhibit performance largely indistinguishable from serial gate operations (all operations temporally separated such that only one qubit is driven in any timestep), while default pulses are degraded by about 2 times. Our measurements—and a change in the observed error dynamics of the default pulses under parallel gate operation—suggest that, under parallel execution, gates on certain qubits exhibit an additional error channel comprising a combination of a direct $\hat{\sigma}_x$ coupling as well as an effective ac-Stark shift proportional to $\hat{\sigma}_z$. Here, amplitude-noise-robust pulses demonstrate the best performance, providing an 8.6 times reduction in effective-gate error, averaged across all five qubits and data sets acquired over a four-day measurement period.

The remainder of this manuscript is organized as follows. In Sec. II we introduce the physical model on which we perform numeric optimization, and indicate the specific optimization approach employed to build consistency with IBM Quantum hardware constraints. In Sec. III we describe the process of mapping optimized pulses to executable commands via the Qiskit Pulse API, pulse calibration on hardware, and validation of noise robustness for optimized pulses. This is followed by an extensive performance comparison under native operating conditions

using various benchmarking techniques in Sec. IV. Additional benefits in “virtualizing” error statistics are demonstrated in Sec. V before we present evidence of crosstalk-error mitigation using optimized pulses in Sec. VI. We conclude with a brief discussion and future outlook in Sec. VII.

II. ERROR-ROBUST PULSE OPTIMIZATION FOR CLOUD QUANTUM COMPUTER HARDWARE

Our task is to define drop-in replacement definitions for the standard physical gates in use on IBM Quantum hardware, based on the concepts of open-loop control. We target solutions that provide high-fidelity operations in the presence of instabilities and noise in the hardware, often referred to as error robust or dynamically corrected gates. Experiments are carried out using a select set of IBM Quantum hardware backends supporting the Qiskit Pulse programming framework. The relevant hardware systems used in this study were *Armonk* [37] and *Valencia* [38], offering one and five qubits, respectively. All programming is conducted remotely via the Qiskit PYTHON package in combination with custom optimization and transpiler tools from Q-CTRL, again accessed via PYTHON.

Arbitrary single-qubit operations are defined and compiled in Qiskit using an Euler-angle decomposition in the Cartesian basis (Fig. 1) of the form

$$U_3(\theta, \phi, \lambda) = R_z(\phi)R_x\left(-\frac{\pi}{2}\right)R_z(\theta)R_x\left(\frac{\pi}{2}\right)R_z(\lambda), \quad (1)$$

where $R_z(\xi)$ is a virtual gate (passive frame shift) through phase ξ , and $R_x(\pm\pi/2)$ denotes a driven $\pm(\pi/2)$ rotation about the x axis. Qiskit Pulse provides flexibility to define optimized replacements for the $R_x(\pm\pi/2)$ rotations in Eq. (1), or to directly deploy resonantly driven pulses implementing arbitrary rotations $R_{\hat{n}}(\theta)$ through angle θ about an axis $\hat{n} = [\cos(\phi), \sin(\phi), 0]$ in the x - y plane on the Bloch sphere. We demonstrate benefits in performance using optimized pulses within this expanded gate set [11].

To define the appropriate optimized replacement, we employ a model-based approach that begins with defining the total system

$$H_{\text{tot}}(t) = H_{\text{ctrl}}(t) + H_{\text{leakage}} + H_{\text{noise}}(t), \quad (2)$$

where the single-transmon control Hamiltonian is expressed as

$$\begin{aligned} H_{\text{ctrl}}(t) &= \frac{1}{2}[\Omega(t)e^{i\phi(t)}\hat{a} + \text{H.c.}] \\ &= \frac{1}{2}I(t)\hat{a}_I + \frac{1}{2}Q(t)\hat{a}_Q. \end{aligned} \quad (3)$$

Here a and a^\dagger are the lowering and raising operators for the transmon, and we have introduced Hermitian quadrature operators functioning in a manner analogous to the

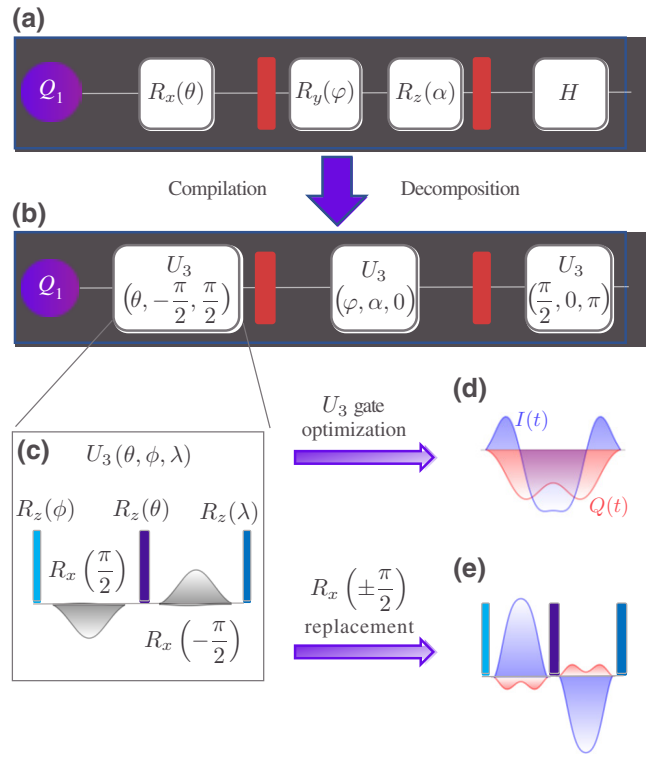


FIG. 1. Schematic description of pulse-waveform integration into compiled Qiskit circuits. (a) A single-qubit circuit composed of physical rotations and “barriers” (red bars), which prevent a compiler from mathematically condensing a circuit, implements a desired transformation [6]. (b) The Qiskit compiler converts operations into a U_3 decomposition, respecting barriers established in the circuit. (c) Each U_3 operation is decomposed into driven rotations, $R_x(\pm\pi/2)$, and virtual frame shifts. (d),(e) Two optimization paths are pursued to craft replacement operations with the same net effect. Here red and blue waveforms represent the Hamiltonian terms $I(t)$ and $Q(t)$ composing the optimized pulse waveform $\gamma(t)$.

Pauli $\hat{\sigma}_{x,y}$ operators for a pure two-level system, defining $\hat{a}_I = \hat{a} + \hat{a}^\dagger$ and $\hat{a}_Q = -i(\hat{a} - \hat{a}^\dagger)$. In the control Hamiltonian, the coupling term $\gamma(t) = \Omega(t)e^{i\phi(t)} \equiv I(t) + iQ(t)$ represents the complex-valued time-dependent (control) pulse waveform.

Next we define the Hamiltonian terms for the dominant error sources: subspace leakage and fluctuations in either the pulse waveform or ambient magnetic fields,

$$H_{\text{leakage}}(t) = \frac{1}{2}\chi(\hat{a}^2)^\dagger(\hat{a}^2), \quad (4)$$

$$H_{\text{noise}}(t) = H_{\text{deph}}(t) + H_{\text{amp}}(t), \quad (5)$$

where χ is the transmon anharmonicity, and the noise Hamiltonian includes contributions from dephasing (deph) and multiplicative amplitude (amp) errors, taking the

forms

$$H_{\text{deph}}(t) = \frac{1}{2}\Delta(t)\hat{a}^\dagger\hat{a}, \quad (6)$$

$$H_{\text{amp}}(t) = \begin{cases} \epsilon_\Omega H_{\text{ctrl}}(t), \\ \frac{1}{2}\epsilon_I I(t)\hat{a}_I + \frac{1}{2}\epsilon_Q Q(t)\hat{a}_Q. \end{cases} \quad (7)$$

Here Δ is the detuning error between the qubit and LO frequency and the $\epsilon_{\Omega, I, Q}$ are small multiplicative errors on the modulus of I and Q components of the pulse waveform, respectively. The first case in H_{amp} describes common-quadrature noise on the pulse modulus, and the second describes differential noise on the I and Q components.

Employing error-robust control in this setting involves defining the pulse waveform $\gamma(t)$, to effectively compensate for the presence of both H_{leakage} and $H_{\text{noise}}(t)$ during execution of an arbitrary unitary operation $U_{\text{ctrl}}(\tau)$ over time τ . This new pulse waveform (Fig. 1) will then be used to implement the target unitary when appropriately compiled in a circuit. In principle, these Hamiltonian terms may have unknown magnitude and temporal dynamics; we require a high-performing solution under a wide range of conditions, making this a problem in robust control rather than optimal control. Formally, given the Schrödinger equation

$$i\dot{U}_{\text{ctrl}}(t) = H_{\text{ctrl}}(t)U_{\text{ctrl}}(t), \quad (8)$$

one attempts to define $H_{\text{ctrl}}(t)$ such that

$$U_{\text{target}} = U_{\text{ctrl}}(\tau), \quad (9)$$

$$\tilde{U}_{\text{noise}} = \mathbb{I}, \quad (10)$$

where \mathbb{I} is the identity operation on the control system. The first line addresses the quality with which the executed unitary matches a target operation. The second incorporates the distorting effect of noisy dynamics via expression of the total propagator as

$$\tilde{U}_{\text{noise}}(\tau) = U_{\text{tot}}(\tau)U_{\text{ctrl}}(\tau)^\dagger. \quad (11)$$

The residual operator $\tilde{U}_{\text{noise}}(\tau)$ defined in Eq. (11) is referred to as the *error action operator*. This unitary satisfies the Schrödinger equation in an interaction picture corotating with the control. In this framework, the condition for optimal control is therefore established by Eq. (9), while the robust-control problem is defined by the condition in Eq. (10).

Control robustness is evaluated as the noise-averaged operator distance between the error action operator, Eq. (11), and the identity, giving a corresponding fidelity

measure

$$\mathcal{F}_{\text{robust}}(\tau) = \left\langle \left\| \frac{1}{D} \langle \tilde{U}_{\text{noise}}(\tau), \mathbb{I} \rangle_F \right\|^2 \right\rangle, \quad (12)$$

where the outer angle brackets $\langle \cdot \rangle$ denote an ensemble average over realizations of the noise processes, and the inner angle brackets $\langle \cdot, \cdot \rangle_F$ denote a Frobenius inner product [Eq. (A1)]. In this approach, an explicit range of values for the uncertain parameters is not used. Instead, the robustness term in the optimization cost enforces that the derivative of the fidelity along the direction of the noise process is close to zero, ensuring that the fidelity of the operation is still high, even when small changes in that parameter are present. See Ref. [18] for a detailed discussion of this error-robust control framework.

We design error-robust pulses using a flexible numerical optimization package based on TensorFlow [18], allowing the system Hamiltonian to be represented using almost-arbitrary functions of the controllable parameters. Hamiltonian terms defining qubit frequencies and anharmonicities are taken from backend specifications provided for each hardware system for either optimization or implementation. The relevant IBM hardware systems exhibit single-qubit coupling rates to the pulse waveform that are significantly throttled relative to many laboratory systems; the gate time for a default $R_x(\pi) \equiv U_3(\pi, -\pi/2, \pi/2)$ rotation takes values $\tau_g \approx 284$ ns on *Armonk* [37] and $\tau_g \approx 71$ ns on *Valencia* [38]. Therefore, it is generally sufficient to consider a two-level approximation and ignore leakage terms when achieving high-fidelity gates.

Optimization defines a map from controllable parameters to the Hamiltonian; the underlying structure of this map is a TensorFlow graph, which can be efficiently evaluated and differentiated using the TensorFlow automatic differentiation library. Once constructed, this mapping (or graph) is used to calculate the optimization cost function. The resulting optimized pulses thus achieve the desired objectives, and do so within the constraints imposed by the user-defined Hamiltonian structure. Full code-based examples are available from Refs. [30,39].

The limited hardware-frequency response of IBM systems is captured directly in our optimizations [39]. Based on our observations of the IBM hardware, we focus on pulses that are transformed as $\gamma(t) \rightarrow L\{\gamma\}(t)$, where $L\{\cdot\}$ represents convolution of the time-domain control with the impulse response of a linear time-invariant filter. We select a sinc function to serve as a smoothing filter that efficiently imposes bandwidth limits on the optimized pulses (see Fig. 9), before resampling into discrete time for compatibility with Qiskit Pulse programming [31]. We have also tested pulses optimized using a bound-slew constraint (limiting the per-segment rate of change in the waveform, $\Delta\gamma/\Delta t$), and pulses with completely unconstrained slew rates; however, these solutions produced inferior

performance compared to smoothed pulses, and we do not present them here.

The optimized pulses presented here are designed to produce robustness against either dephasing errors, amplitude errors, or both errors simultaneously (dual robust). In general, they provide additional robustness against error at the expense of extended operational duration; the optimized pulses used in this work are longer than their IBM default counterparts by factors of about 40%–140% in the *Valencia* backend. Full details of optimized pulses created and used in this work are presented in Fig. 9 in Appendix C.

III. IMPLEMENTATION AND VALIDATION OF ERROR-ROBUST OPTIMIZED PULSES

Qiskit Pulse formatting permits the definition of analog pulse waveforms corresponding to programming of the I and Q terms in our control Hamiltonian. IBM default pulses are implemented by simply calling their daily calibrated $U_3(\theta, \phi, \lambda)$ pulses with the appropriate angles.

For optimized controls, one could opt to replace the individual $R_x(\pm\pi/2)$ pulses in the default U_3 decomposition with corresponding optimized pulses or, alternatively, replace the entire U_3 construction with an optimized pulse implementing the same gate, as shown in Fig. 1. All pulses in this work are obtained using the latter approach. Appropriate for current hardware settings,

optimized waveforms are discretized in units of the minimum backend timing resolution, $dt = 0.22$ ns, and are designed to have total duration (gate time) $\tau_g = n_1\tau_s$, consisting of n_1 segments of uniform duration $\tau_s = n_2dt$, subject to the condition that $n_1n_2 = 16m$ for integers n_1, n_2, m . Once optimized, as described in Sec. II, execution of a custom pulse in Qiskit requires only a single command `Play(Waveform(custom_pulse), drive_channel)`.

Arbitrary multiqubit circuits combining multiple default or optimized-pulse waveforms in time across all available qubits are sequenced into a Qiskit Pulse compatible format via a custom transpiler. Circuits such as randomized benchmarking, calibration routines, or algorithms may be efficiently deployed, replacing effective gates with arbitrary predefined pulse waveforms. See Ref. [40] for a code-based implementation using Qiskit Pulse.

The first step of pulse execution in hardware requires mapping a complex-valued and piecewise-constant waveform $\gamma(t)$ —the result returned from a numerical optimization—to instructions for the Qiskit Pulse API (Fig. 2). Each analog pulse is defined for IBM Quantum hardware in terms of dimensionless variables $A = A_I + iA_Q$ with real and imaginary components proportional to physical input voltages, and bounded as $A_{I,Q} \in [-1, 1]$, $|A| \leq 1$. A physical calibration routine is used to establish the map $(A_I, A_Q) \leftrightarrow (I, Q)$, from programmable amplitudes to control-pulse amplitudes in the laboratory frame. We

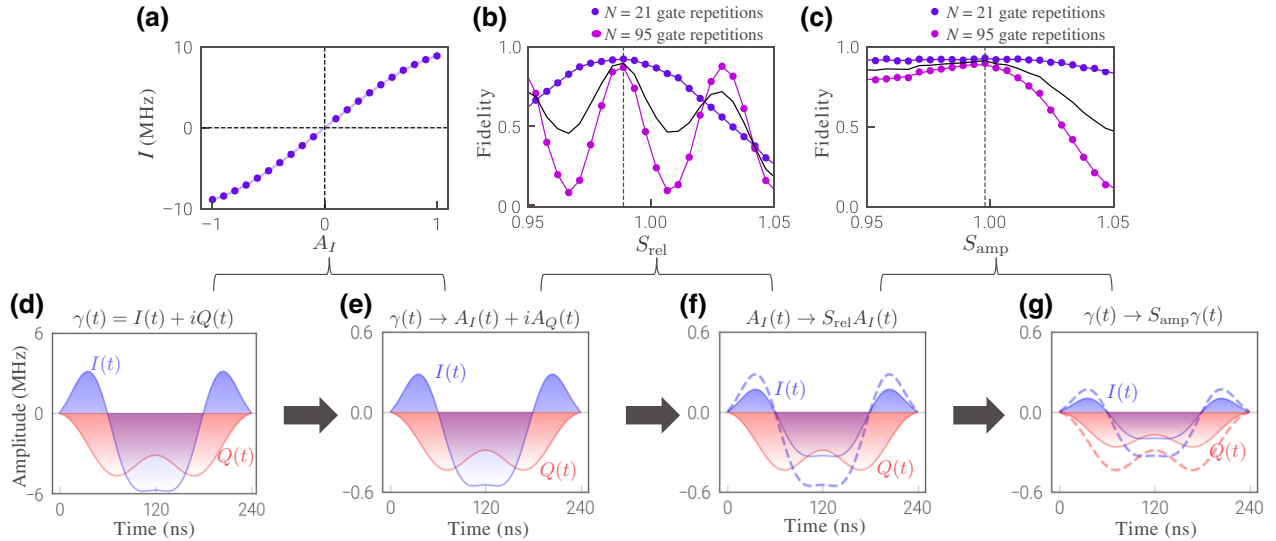


FIG. 2. Pulse transformation from Hamiltonian representation and calibration for execution on hardware. (a) Response of *Armonk* hardware defining the mapping $(I, Q) \rightarrow (A_I, A_Q)$ from physical to programmable waveforms, used to perform the calibration step in panels (d) and (e). Each point in panel (a) corresponds to a Rabi rate derived from fitting to a standard Rabi flopping experiment at a fixed square-pulse amplitude of varying duration. (b) Calibration measurements used to determine the calibration parameter S_{rel} used to tune the relative scaling between I and Q waveforms, resulting in the deformations (not to scale) in panels (e) and (f). Each trace in panel (b) corresponds to a measured state fidelity for different pulse-repetition numbers. Finding a common maximum denoted by the average (solid line) provides enhanced sensitivity to S_{rel} while excluding net rotation errors mod (2π) . (c) Similar procedure used to find the calibration parameter S_{amp} used to tune the common-quadrature scaling of the waveform, resulting in the deformations (not to scale) in panels (f) and (g).

first coarsely calibrate the channels (A_I, A_Q) by measuring Rabi oscillations in the time domain using square, fixed-amplitude, single-quadrature pulses on I or Q channels independently. Fitting the respective Rabi oscillations to a cosine, we extract the corresponding Rabi frequency and interpolate between measurements to produce a smooth functional dependence (A_I, A_Q) \leftrightarrow (I, Q), as shown in Fig. 2(a). A desired waveform may then be predistorted via this interpolation function to account for any nonlinearities in the hardware response.

Additional hardware imperfections are routinely experienced, leading to small offsets and distortions in the pulse waveforms programmed by the variables $A_{I/Q}(t)$. We implement fine calibration of the absolute and relative scaling of the I and Q channels by expressing the pulse waveform using two scaling factors as $\gamma(t) = S_{\text{amp}}[S_{\text{rel}}A_I(t) + iA_Q(t)]$. Calibration proceeds by applying repeated pulses to amplify coherent miscalibration errors and varying the parameters S_{amp} and S_{rel} for different numbers of pulse repetition [Figs. 2(b) and 2(c)]. Scaling factors are selected that simultaneously yield maximum state-transfer fidelity for two different calibration-sequence lengths. Supplementary calibration sequences that maximize sensitivity to the orthogonality of $R_x(\theta)$ and $R_y(\theta)$ may also be employed and follow a similar routine.

Once properly calibrated we can begin to explore the performance enhancements delivered through use of optimized pulses. We begin by comparing the error robustness of optimized, calibrated $R_x(\pi)$ pulses to the IBM default implementation on IBM's *Armonk* system [Figs. 3(a)–3(d)]. Here we apply an intentionally engineered detuning between the microwave frequency and qubit resonance [Fig. 3(e)], or an over-rotation error realizing quasistatic terms in H_{noise} [Fig. 3(f)]. As the strength of the detuning or over-rotation error is increased, the state-transfer fidelity of the IBM default pulse degrades rapidly. By contrast, for noise processes corresponding to up to 1-MHz detuning from the qubit frequency and deviations of 20% in the pulse amplitude, we see that optimized robust pulses provide high-fidelity state transfer, a key signature of robustness to the engineered H_{noise} term. This robustness is primarily exhibited against the noise channel for which optimization is performed, though the dephasing-robust pulse also shows some robustness against amplitude errors (see also Fig. 9 in Appendix C for information on the robustness of pulses designed for operation on *Valencia*). Data match theoretical predictions calculated based on the implemented waveforms well for all pulses, with minor deviations observed for large error magnitudes. We note that optimized pulses lacking band limits are not able to demonstrate this form of robustness, likely due to low-fidelity reproduction of the pulse waveform at the qubits.

These initial measurements validate that we are able to implement calibrated operations enacting high-fidelity

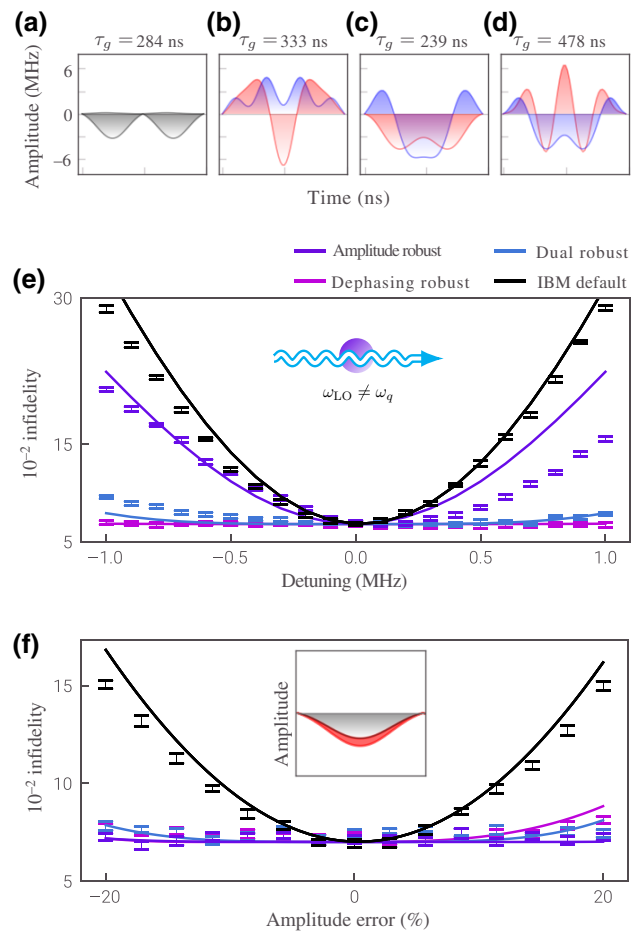


FIG. 3. Demonstration of optimized-pulse robustness against dominant error sources, using the single-qubit backend *Armonk*. (a) IBM default pulse, (b) amplitude-robust pulse, (c) dephasing-robust pulse, (d) dual-robust pulse. All pulses implement a $R_x(\pi)$ gate. Red and blue represent $I(t)$ and $Q(t)$, respectively. On this backend Rabi rates are reduced relative to other hardware, resulting in about 2–3 times longer pulses than executed on *Valencia*. (e) Infidelity for each pulse identified above as a function of an engineered detuning $\Delta = (\omega_q - \omega_{\text{LO}})/2\pi$, obtained by sweeping the LO frequency, ω_{LO} , away from the qubit resonance, ω_q , determined by the IBM backend. Lines represent theoretical simulations and the error bars represent the standard deviation over 20 independent experimental measurements for each pulse type. Engineered detunings are established in the Qiskit Pulse API. The solutions from simulations are vertically offset to match the baseline measurement-limited experimental fidelity. (f) Similar as above for a pulse-amplitude error. Inset shows schematic change in the pulse area leading to an over- or under-rotation error due to the applied pulse-amplitude error.

state transfer to within measurement error, and that optimized pulses exhibit robustness against the appropriate Hamiltonian terms. Next we provide a comparative performance analysis for these pulses against the calibrated IBM default pulses under native operating conditions on the five-qubit *Valencia* system.

IV. ERROR MITIGATION UNDER NATIVE HARDWARE PERFORMANCE CONDITIONS

Under native operating conditions, single-qubit gates are tabulated to achieve operational fidelities exceeding 99.9%. By contrast, for the backends used in our experiments, the reported readout error using Qiskit “measurement level 2” [41] without any postprocessing can exceed 5%, as seen in Fig. 3. To ensure we are able to perform detailed comparisons between pulses beyond the limits imposed by so-called state preparation and measurement (SPAM) errors, we employ a range of “error-amplifying” protocols that are SPAM insensitive; using these, it is possible to extract an effective error rate much smaller than the SPAM threshold.

The first sequence we employ consists of repeated application of a target rotation. This sequence is maximally sensitive to coherent over or under rotations arising from various noise sources. All repetitions are constructed to comprise an odd-integer multiple of $R_x(\pi)$. As the repetition number grows, we observe a gradual deviation of the ideal state-transfer probability to the excited state that differs substantially between different pulse implementations [Fig. 4(a)]. Deviation from the ideal state transfer is slowest for optimized pulses designed to suppress terms in H_{noise} , and worst for the default U_3 gate, across all qubits [Figs. 4(a)–4(e)]. This indicates that the numerically optimized pulses are more robust to naturally occurring noise, drifts, and miscalibrations in either pulse amplitude, duration, or frequency detuning.

Data are fit [Fig. 4(a)] with a decay curve modelling simultaneous coherent rotation errors and incoherent T_1 -type decay, in line with published error models [34]

$$\frac{P(1)}{2}[1 - \cos(\pi + n\epsilon_r)e^{-\beta n}]. \quad (13)$$

Here, $P(1)$ corresponds to the value of the first point in the data set and represents a constant (but variable between data sets) offset associated with measurement error. We allow both the angular rotation error, ϵ_r , and the population-decay error per gate, β , to serve as fit parameters and observe excellent agreement with the data. The total effective gate error, ϵ is determined from the quadrature sum of coherent and incoherent error terms $\epsilon = \sqrt{\epsilon_r^2 + \beta^2}$.

Typical values of ϵ are in the range 10^{-3} – 10^{-2} , somewhat larger than tabulated randomized-benchmarking results (see Sec. V for a full discussion). The extracted incoherent decay is also somewhat stronger than the backend tabulated values reported by IBM for the default pulses. By contrast, the extracted values of $T_1 \sim \tau_g/\beta$ for optimized pulses on each qubit compare well against tabulated values, and exhibit fluctuations between data sets comparable to the statistical deviation recorded by IBM’s internal calibration procedures (see Fig. 8 in Appendix B).

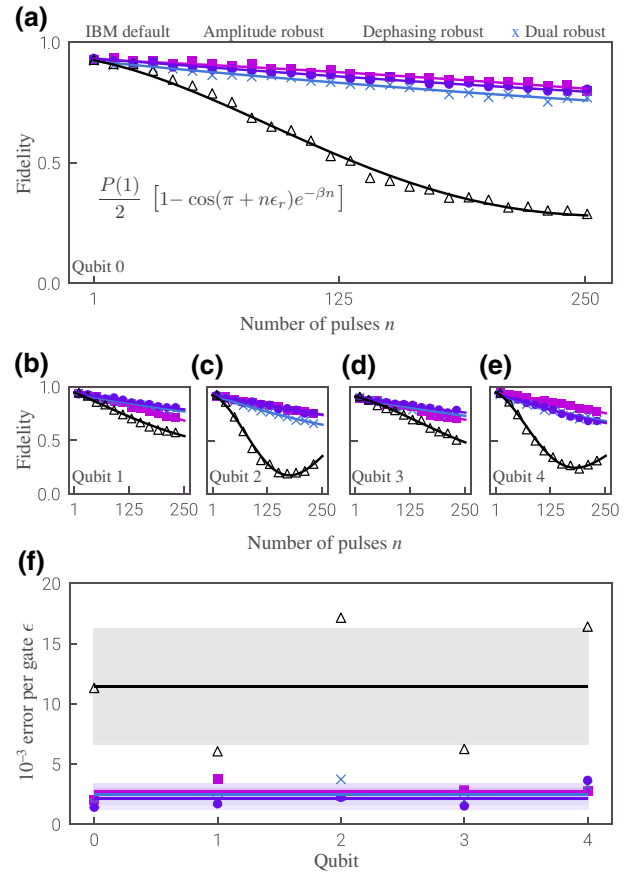


FIG. 4. Gate performance for all five qubits on the IBM *Valencia* device on 5 August 2020. Full details of the pulses in use and their durations are presented in Fig. 9, but operations are approximately 2–3 times shorter than on *Armonk*. (a)–(e) Single-qubit data sets for repeated pulses applied independently to each qubit, overlaid with theoretical fits. (f) The effective error rate extracted from the fit parameters, ϵ , is shown for four different pulses: IBM default U_3 (triangles), amplitude robust (circles), dephasing robust (squares), and dual robust (crosses). Solid lines represent the average across the device, while the shaded region corresponds to the average \pm the standard deviation σ_q . All robust pulses outperform the default for every qubit.

Alternate fitting approaches, such as fixing all values of $P(1)$ across a device or fixing T_1 based on measured values, visibly degrade fit quality, but do not materially alter the best-fit value of ϵ . We note that qubits two and four routinely show substantial deviation in performance from tabulated values (possibly due to transient two-level fluctuators [42,43]).

The effective gate error ϵ for multiple optimized pulses extracted from data collected within a single-day calibration period on the IBM Quantum device is shown in Fig. 4(f). Across all qubits we observe superior performance from numerically optimized robust pulses, with marginal difference between them on this day. In these representative data the observed infidelity for numerically

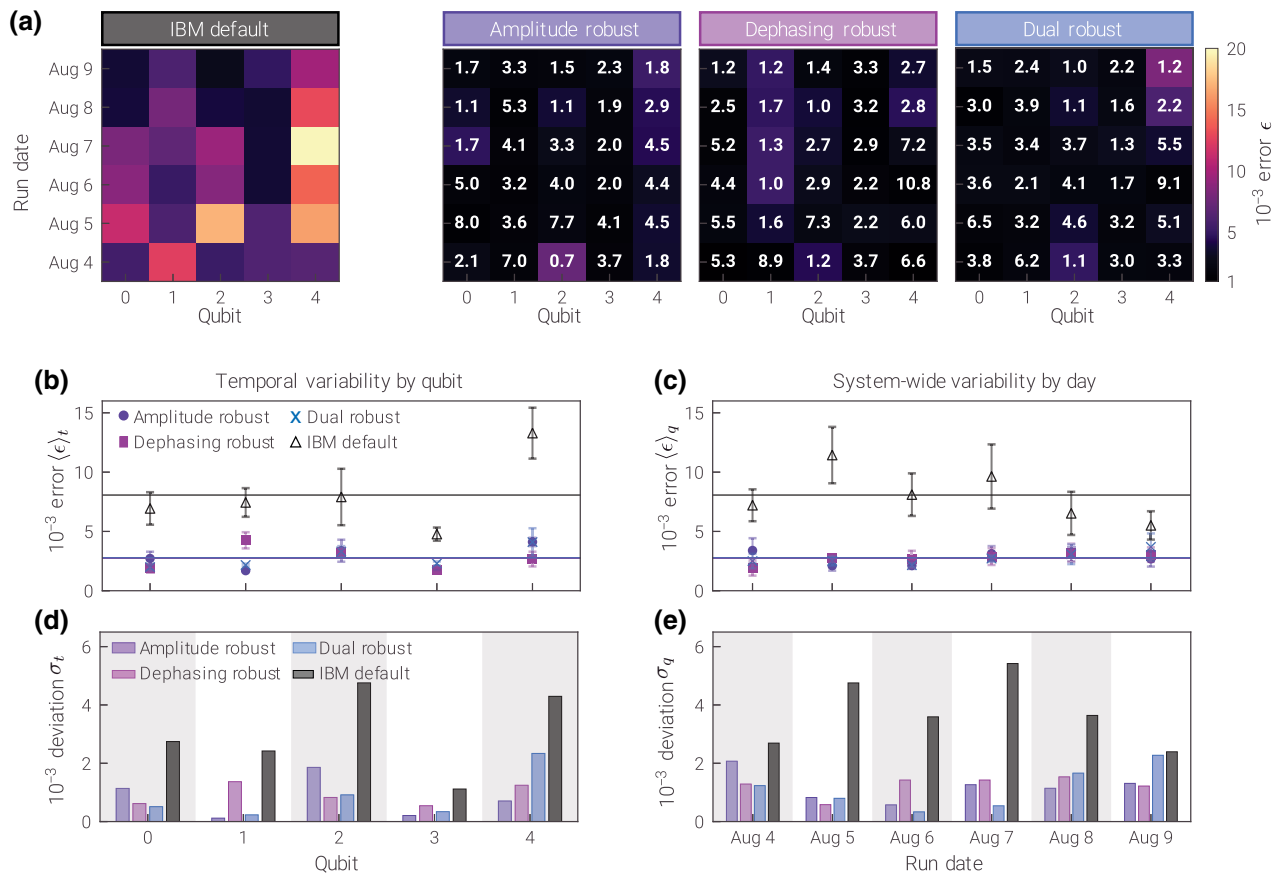


FIG. 5. (a) Effective error ϵ across qubits and measurement days expressed as a color scale. All experiments with robust pulses were carried out using a calibration performed on 3 August 2020, while the left plot used the IBM default pulse from the most recent daily hardware calibration. The gate improvement from robust pulses for each qubit and day is quantified by the error reduction ($\epsilon_{\text{default}}/\epsilon_{\text{robust}}$), given explicitly by the numbers overlaid on the colorplot. The temporal and system-wide variabilities are respectively summarized in panels (b),(d) and (c),(e). The effective error averaged over different days, $\langle \epsilon \rangle_t$, for each qubit is shown in (b), while (c) displays the average over all qubits in the device, $\langle \epsilon \rangle_q$, for different days. The standard deviations for each average error are shown as error bars in (b),(c) and as bar plots in (d),(e) for ease of comparison between the different pulses. All robust pulses strongly reduce both temporal and system-wide variabilities.

optimized pulses is dominated by the effective exponential decay that we attribute to T_1 . Quantitatively, in this single-day data set we observe up to an 8 times improvement in the reduction of single qubit ϵ . Similar results were achieved over the six-day measurement window 4–9 August 2020 shown in Fig. 5, and over an additional window of 18–22 May 2020. We observed a maximum of 10.8 times improvement in single qubit ϵ .

V. REDUCING SPACE-TIME ERROR FLUCTUATIONS AND CORRELATIONS

IBM Quantum hardware exhibits natural performance variation across qubits on a device and over time. For instance, the tabulated T_1 and T_2 values show typical daily variability of several tens of percent and up to over 2 times for individual qubits [44]. Variation of a similar scale is also observed between qubits on a single device. This

spatiotemporal inhomogeneity is reflected in the extracted ϵ for the IBM default DRAG pulse, displayed as a color scale for the different pulses, across both qubits on a device and measurement days [Fig. 5(a)]. For the August 2020 data acquisition period shown here, we observe more than an 8 times difference in ϵ , from the best to the worst recorded single-device performance [Fig. 5(a)].

This hardware variability is suppressed through use of numerically optimized pulses. Beginning with Fig. 4(f), we observe that the system-wide variability on *Valencia* for each pulse, expressed as the standard deviation of ϵ across qubits, σ_q , is reduced for all of the optimized error-robust optimized pulses (indicated by colored shading). For this day, the dephasing-robust pulse provides an 8.2 times reduction of σ_q relative to the IBM default pulses. A similar reduction in performance variability across devices is observed on multiple measurement days. A more explicit quantitative comparison of the system-wide variability for

IBM default and robust pulses is shown in panels (c) and (e) of Fig. 5, where the effective error averaged over all qubits (c), $\langle \epsilon \rangle_q$, and the corresponding standard deviations (e) are presented for the different days of data acquisition. For all optimized pulses, the variability across devices is reduced; daily average “error homogenization” ranges from about 1.1–10.8 times across this week.

Temporal variation in gate performance for each qubit is also suppressed through use of optimized pulses. Improvement on individual-qubit stability over time using optimized error-robust pulses is summarized in Figs. 5(b) and 5(d), which respectively show the time-averaged errors $\langle \epsilon \rangle_t$ for each qubit and their deviations σ_t . These plots enable direct comparison of the temporal variability for different pulses on each qubit, with robust pulses reducing the deviation in the time-averaged error from 1.8 times (qubit 1, dephasing robust) to over 20 times (qubit 1, amplitude robust) as compared to the IBM default pulse. Full tabulation of all extracted errors and calculated statistical moments for each pulse type are presented in Appendix D.

These measurements highlight an additional practical benefit; in the data displayed in Fig. 5, the IBM default pulses are recalibrated daily, while the numerically optimized pulses are calibrated only once on 3 August and used with the same setting over the course of the week. Measurements of device performance within a single 24 h period indicate that calibrations of default pulses begin to go stale after about 12 h, while the calibration for numerically optimized robust pulses is extended by over 10 times to at least six days. Even at the end of the six-day measurement period, each individual optimized pulse type on each qubit still outperforms the default pulse [Fig. 5(a)].

Single-qubit measurements have been cross-referenced against Clifford randomized benchmarking performed using either U_3 gate constructions composed from default $R_x(\pi/2)$ pulses, or combinations of error robust $R_x(\pi)$ and $R_x(\pi/2)$ pulses. We observe that numerically optimized error-robust pulses can outperform the default pulses in the returned p_{RB} despite the extended gate duration relative to the default, but now by a smaller margin of approximately 10%–20% [Fig. 6(a)], relative to the experiments presented above. In general, we find that the extracted p_{RB} for both gate classes vary by date [Figs. 6(a) and 6(d)], and that differences in performance fall within statistical variations between data sets. In addition, we have observed that the best results are achieved using a compilation involving both driven, optimized $R_x(\pi)$ and $R_x(\pi/2)$ pulses, rather than constructing all gates from the U_3 decomposition. We believe this is due to the reduced execution time and the presence of limiting T_1 processes [combining two optimized $R_x(\pi/2)$ pulses requires about 6% longer than a single optimized $R_x(\pi)$ pulse].

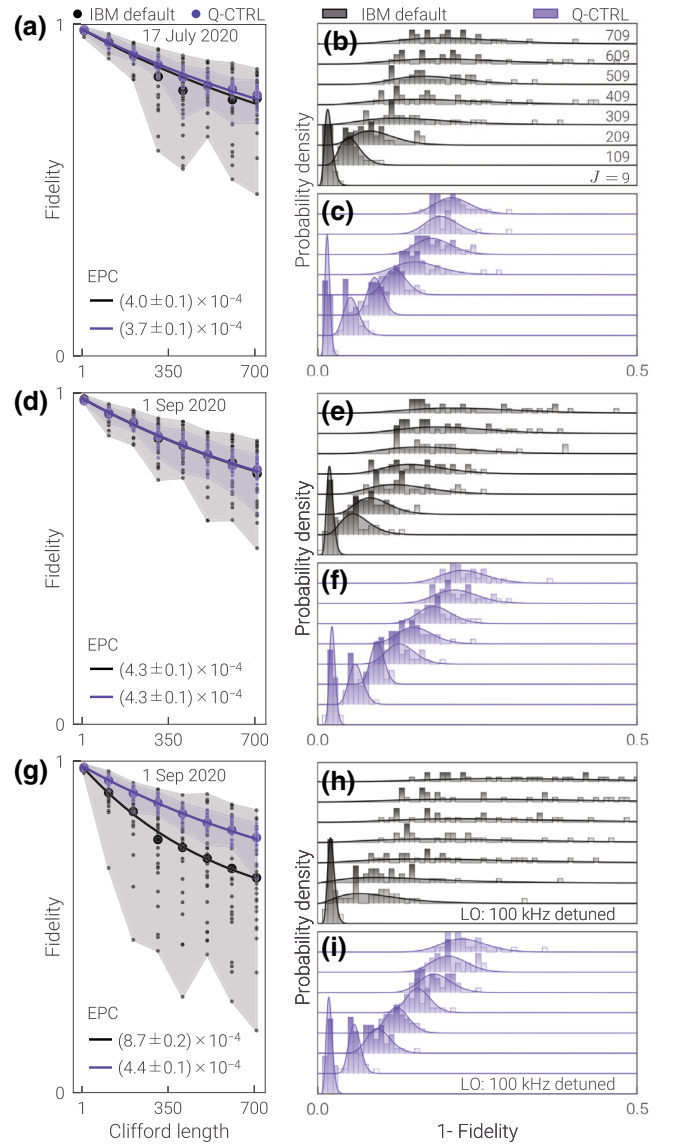


FIG. 6. Randomized benchmarking comparison between IBM default pulses and optimized dephasing-robust pulses. In these experiments all 24 single-qubit Clifford gates are transpiled using only $R_x(\pi/2)$ and $R_x(\pi)$ driven pulses, interleaved with virtual R_z gates as needed. For the IBM default case (gray), $R_x(\pi) = R_x(\pi/2)R_x(\pi/2)$, where $R_x(\pi/2)$ is a DRAG pulse, as shown in Fig. 9(b). For the Q-CTRL case (purple), $R_x(\pi/2, \pi)$ gates are independently optimized pulses, as shown in Figs. 9(f) and 9(j), respectively. Panel (b) shows histograms of measured fidelity for each sequence length, J , using IBM default pulses, overlaid with fits to gamma distributions. Large variance and skew in distributions for default pulses is indicative of coherent errors [23,35,45]. Panel (c) reveals both the narrower distribution and shape better approximated by a Gaussian, indicating decorrelation of coherent errors [23]. Panels (d)–(f) show similar data, repeating the experiment on a later date. Panels (g)–(i) show similar data, but imposing a 100-kHz detuning on the LO to reveal the impact of a deliberate (large) frequency miscalibration error, changing the extracted error per Clifford (EPC) p_{RB} by about 2 times.

The extracted values of p_{RB} are approximately 8 times smaller for optimized pulses and nearly 20 times smaller for default pulses than the direct probing experiments presented in previous figures. This discrepancy is consistent with the previously reported reduced sensitivity of randomized benchmarking to coherent errors arising from, e.g., calibration errors and drifts [36]. This insensitivity can lead to large deviations between expected and actual hardware performance at the algorithmic level [46]. Our measurements nonetheless reveal the action of robust optimized pulses in modifying the characteristics of gate errors through suppression of error correlations.

The distribution of randomized-benchmarking outcomes over sequences contains a key signature of coherent errors in the form of a skewed and broadened distribution [23,35,45]. For the IBM default pulses, this skew and broad variance is apparent, showing a large spread of measured return probabilities over different measured sequences, as illustrated in Fig. 6(a). By contrast, the numerically optimized pulses show a narrow and symmetric distribution around the mean.

The form of these distributions is quantitatively matched to a gamma distribution with a shape varying with the form of the error model, ranging from approximately Gaussian (incoherent errors) to approximately log normal (coherent errors) [35,45]. In Figs. 6(b) and 6(c), histograms of measured infidelities over different randomized sequences are overlaid with fits to gamma distributions. Agreement is good in both cases, and highlights the differences between default and optimized pulses. These observations are consistent with the interpretation that error-robust optimized pulses suppress coherent errors, leaving residual errors dominated by incoherent T_1 processes, while default pulses remain susceptible to both error processes. We consider this a form of “error virtualization” in which the statistical characteristics of the gates change, even if the observed p_{RB} is only minimally affected [23].

Additional experiments performed with pulses designed to suppress both coherent error processes and leakage out of the qubit subspace show similar performance, with differences laying within the statistical variations in the IBM hardware. The range of randomized-benchmarking studies and comparisons we have undertaken suggests that performance may be limited by Hamiltonian terms that are not accounted for either in published system documentation or associated numerical optimizations, in addition to incoherent T_1 processes.

VI. SUPPRESSION OF CROSSTALK ERRORS

The measurements presented thus far focus on executing pulses on each qubit sequentially, keeping the remaining qubits idle. This matches the serialization approach taken in standard Qiskit compilers [47], but misses an opportunity to reduce circuit-execution time and the associated

incoherent errors. Understandably, parallel execution is not favored in general as crosstalk errors degrade performance relative to tabulated single-gate metrics without further mitigation strategies [48].

We test the benefits of numerically optimized error-robust pulse design in the presence of crosstalk error by studying gate performance under full parallel execution across all five qubits on *Valencia* [38]. Again, we measure deviation of a sequence of repeated pulses from unity fidelity, and extract an effective error per gate. In this case the best fit to the experimental data using IBM default pulses is consistent with the assumption of a dephasing noise Hamiltonian, leading to a fit function

$$\frac{P(1)}{2} [1 - e^{-(\beta^{\parallel} n)^2} \cos(\pi + n\epsilon_r^{\parallel})]. \quad (14)$$

Fit quality for the default pulses is improved by appropriate selection of the functional form, now including a Gaussian decay, but quantitative measures for effective rotation error, here captured through ϵ_r^{\parallel} , are minimally affected. By contrast, we continue to find that best-fit performance for optimized error-robust pulses is realized using Eq. (13).

Extracting an effective error per gate, ϵ , we first observe that performance of the default pulse averaged across the device has degraded by an average of approximately 2 times relative to serial execution [Fig. 7(a)]. Numerically optimized pulses in parallel execution continue to outperform the default pulse, and even outperform the default pulse using serial execution. Surprisingly, numerically optimized pulses designed to exhibit robustness to both error quadratures perform the worst among the optimized solutions. Under parallel execution, amplitude-robust pulses marginally outperform dephasing-robust pulses across all days and both in device-wide averages and individual-qubit performance, showing up to a 9.5 times improvement in reduction of ϵ for an individual qubit. System-wide and temporally averaged performance for both the dephasing-robust and amplitude-robust pulses is unchanged or modestly improved (consistent with daily experimental fluctuations) relative to serial pulse application [Figs. 7(b)–7(c)].

Examining the effective incoherent decay constants, β and β^{\parallel} , yields results consistent with the appearance of an additional decay process beyond T_1 under parallel pulse application. Despite good fit agreement, conversion of the decay constants for the (poorly performing) default and dual-robust pulses to a time constant now substantially underestimates T_1 relative to tabulated values (see Fig. 8 in Appendix B). By contrast, the dephasing- and amplitude-robust pulses, which perform well, exhibit extracted values of β that still agree approximately with the tabulated T_1 from the hardware backend.

These performance differences are striking when we examine spatiotemporal variability in gate performance. In

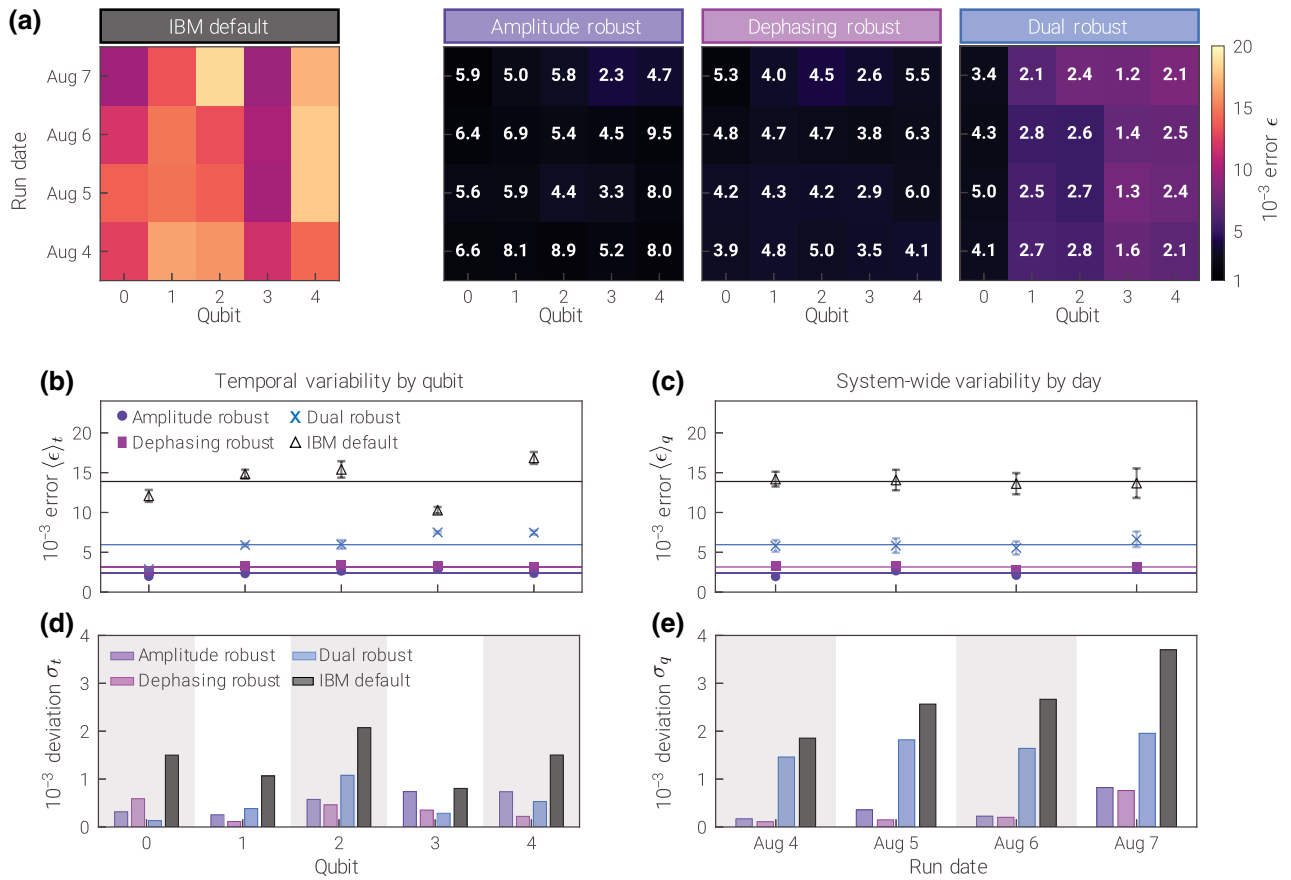


FIG. 7. Device-wide performance, as in Fig. 5, but now for parallelized operations (pulses applied simultaneously on all five qubits). Comparison between panels (a) in this figure and in Fig. 5 shows the deleterious effect of crosstalk when the IBM default pulses are applied in parallel. In contrast, dephasing- and amplitude-robust pulses perform as well, or even slightly better, than in the serial pulse case. The standard deviations in (d) and (e) show that robust pulses can suppress temporal variability by more than 11 times and system-wide variability by more than 17 times.

addition to overall error reduction, relative to the default pulse we see up to an average of about an 8.6 times reduction of variance across all measurement runs for the best-performing amplitude-robust pulses [Fig. 7(d)], and up to about a 12 times reduction of variance across qubits on a single day [Fig. 7(e)]. The absolute magnitude of the temporal and system-wide variability appears somewhat smaller than in serial execution for these pulses [Figs. 7(b)–7(e)], which we believe arises due to crosstalk becoming a single dominant error process. We emphasize that these data sets were acquired in an interleaved fashion with the data in Fig. 5, over the window of 4–7 August 2020.

From these data sets, we surmise that crosstalk in these devices likely occurs as both a term in $H_{\text{noise}} \propto \hat{\sigma}_x$ contributing to an effective amplitude error, and an additive term $H_{\text{noise}} \propto \hat{\sigma}_z$ due to what appears to be an effective ac-Stark shift. We draw this conclusion for the following reasons. First, the superior performance of amplitude-robust versus dephasing-robust optimized pulses and the device-wide performance degradation of

the dephasing-robust pulses suggest that the error process induced by parallel operation is not pure dephasing. Next, the difference in best-fit form for the Gaussian decay of default pulses (while the dephasing-robust and amplitude-robust pulses maintain a simple exponential decay form) suggests the presence of a quasistatic dephasing term that is suppressed by the robust pulses until their performance is limited by an incoherent error process. We do not have a full explanation for the poor performance of the “dual-robust” pulses, but note that they are 170 ns in duration, 70% longer than the dephasing-robust pulses, providing an additional potential source of error.

VII. CONCLUSION

In this work we have employed the Qiskit Pulse programming API to experimentally test numerically optimized error-robust pulses implementing single-qubit gates on IBM cloud quantum computers. We have performed a detailed comparison of different pulse designs

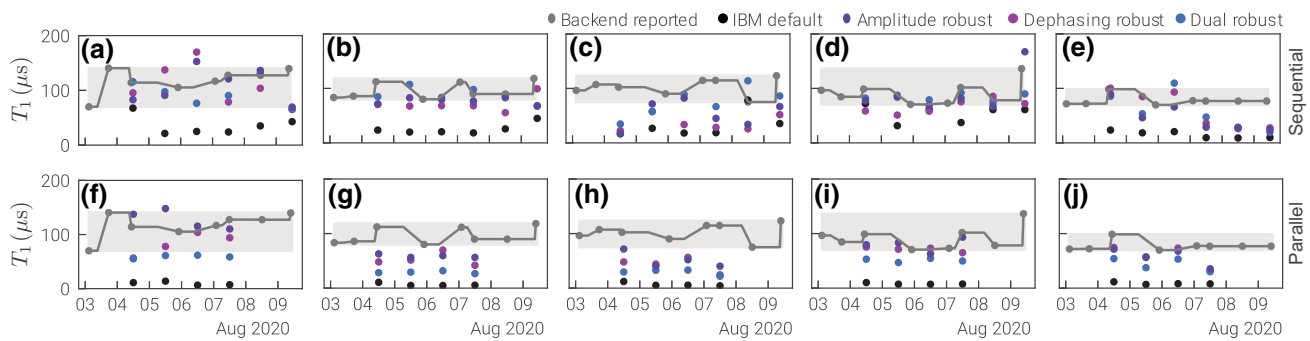


FIG. 8. Tabulated (gray circles) and fitted (colored circles) T_1 during the week of the experiments on *Valencia* for qubits 0 to 4 (from left to right). The shaded area corresponds to the range between the minimum and maximum T_1 reported in that week. The fitted decay time is extracted from $T_1 = \tau_g/\beta$ using the gate durations and the tabulated β or β^{\parallel} values in Appendix D for the serial (top row) and parallel (bottom row) experiments.

incorporating various optimization constraints and objectives. We find that in realistic circuits [46], coherent errors dominate default-pulse performance and are manifested as circuit errors that can be an order of magnitude larger than randomized benchmarking results would suggest. These errors are suppressed by up to about 10 times for an individual qubit through use of optimized error-robust pulses under both serial and parallel execution on the five-qubit *Valencia* [38] hardware backend. Our results reveal that the best performing single-qubit pulses are those designed to be robust against either control-amplitude or dephasing noise, and incorporating a sinc-smoothing function to account for band limits on transmission. Our investigations further suggest the presence of crosstalk errors due to simultaneous dephasing and coherent rotation errors, both of which are suppressed through an appropriate choice of robust pulses.

In addition, we demonstrate that optimized robust pulses virtualize errors by changing their statistical properties in addition to error magnitudes [23]. We use error-robust optimized pulses to demonstrate mitigation of hardware variability across qubits and time, reducing the standard deviation of measured qubit errors by about 8.6 times over a four-day measurement window. During this window, the robust pulses are never recalibrated, while IBM default pulses are observed to exhibit performance variability on about a 12 h timescale and require daily recalibration; this corresponds to a net improvement in the calibration window by over about 10 times. Additional randomized-benchmarking measurements reveal key signatures of coherent-error suppression using optimized pulses, even as returned values of p_{RB} appear consistent with T_1 -limited performance.

The techniques demonstrated here are transferable to any system facing strict timing and bandlimited controls, and consequently have major implications for all superconducting platforms. Although the focus in this paper has been on single-qubit gates, and on providing a comprehensive description of the implementation steps,

we note that these techniques are not limited to single-qubit gates. Similar experiments demonstrating optimized two-qubit gates are the subject of a forthcoming publication. Our work also focuses on the specific execution of these techniques using analog-layer programming on a cloud quantum computer. To date, researchers have been limited primarily to fixed gate sets used to compose algorithms. Using error-robust pulses as a case study, our work demonstrates that a wide range of tunable Hamiltonians may be successfully engineered, indicating alternative avenues of research.

Moreover, these demonstrations highlight the potential for error-robust quantum control to have impact on the performance and design of quantum computers, and flow through to impact higher-level abstractions such as circuit compilation and quantum error correction [18]. For instance, appropriately optimized pulses may permit relaxation of classical-crosstalk engineering tolerances, and permit full parallelization of all single-qubit unitaries as a default mode of operation without performance degradation. Similarly, the error-rate homogenization we have demonstrated here may simplify noise-aware compilation [44,49,50] that nominally trades circuit complexity for improved gate errors.

We note that the I and Q waveforms for the default DRAG pulses available through the IBM backends are in fact heavily distorted from the analytic forms expected to produce leakage suppression. Indeed, simulations using these pulses show no leakage suppression at the claimed transmon anharmonicities. However, the durations of the default pulses (36 ns for *Valencia*, 284 ns for *Armonk*) are also sufficiently long that leakage errors are in fact small relative to T_1 or T_2 errors, or errors arising from amplitude or detuning instabilities. From these observations we conjecture that the default DRAG pulses have been modified under the daily IBM calibration procedure, in which the pulse shape is heavily distorted to compensate for the system's primary errors—daily drifts in amplitude or frequency calibrations. For this reason, the

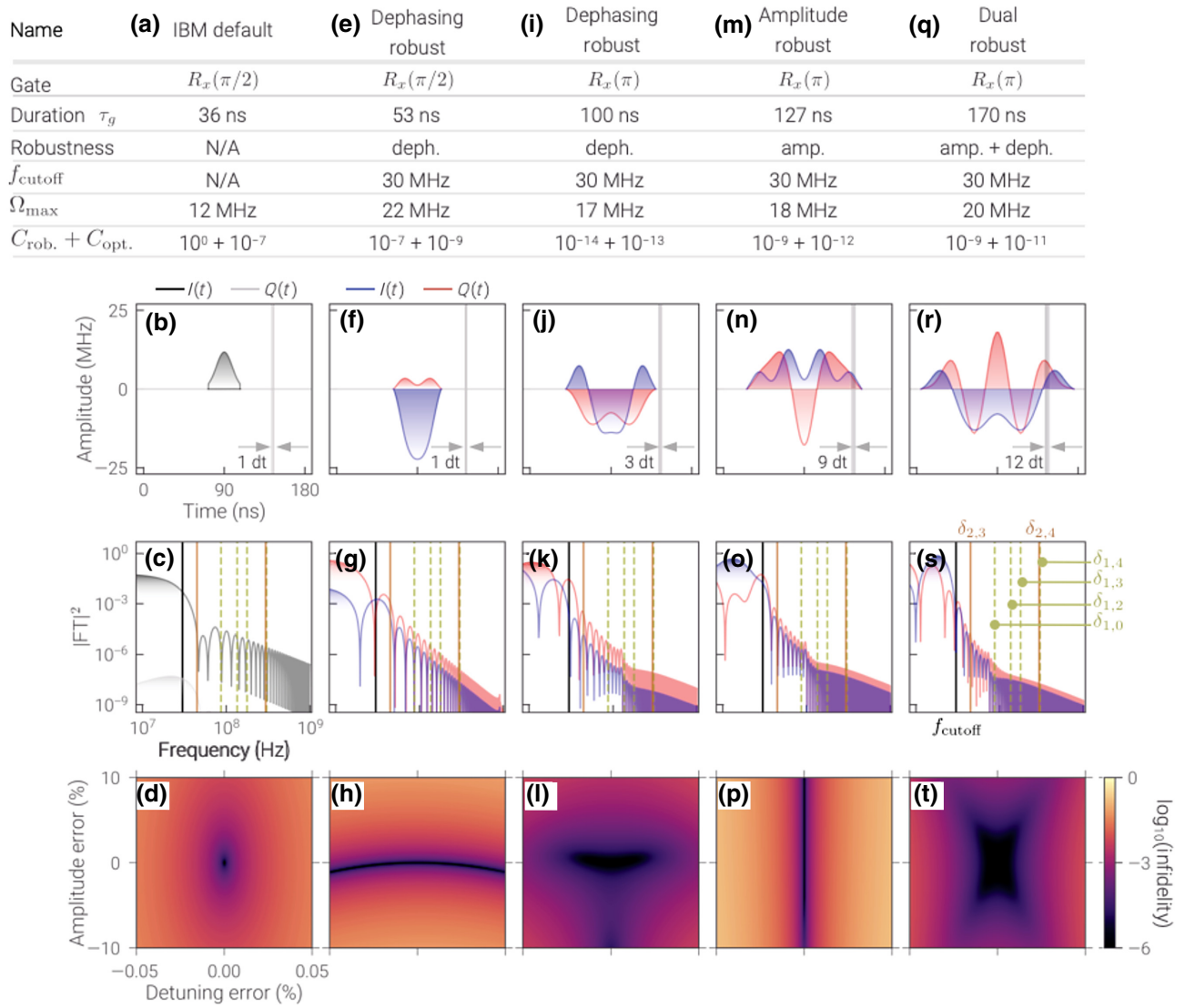


FIG. 9. Description and comparative analysis of pulse waveforms identified in the main text implemented on the *Valencia* backend. (a)–(d) IBM default $R_x(\pi/2)$ pulse. (e)–(h) Q-CTRL dephasing-robust $R_x(\pi/2)$ pulse. (i)–(l) Q-CTRL dephasing-robust $R_x(\pi)$ pulse. (m)–(p) Q-CTRL amplitude-robust $R_x(\pi)$ pulse. (q)–(t) Q-CTRL dual dephasing- and amplitude-robust $R_x(\pi)$ pulse. (a), (e), (i), (m), (q) Tabulated characteristics of IBM default DRAG compared with optimized Q-CTRL pulses. Q-CTRL pulses are optimized for robustness against dephasing and/or amplitude errors (row 3) subject to bandlimited (row 4) and bound-amplitude (row 5) constraints. Here f_{cutoff} refers to the 3 dB cutoff frequency associated with a low-pass sinc filter, and Ω_{max} is the maximum Rabi rate over the pulse duration. Optimization costs are shown in row 6, with cost functions $C_{\text{opt.}}$ and $C_{\text{rob.}}$ defined in Ball *et al.* [18]. For comparison, panel (a) shows these metrics computed for the default pulse. (b), (f), (j), (n), (r) I and Q waveforms for IBM default DRAG pulse (gray) compared with optimized Q-CTRL pulses (red and blue). Durations for each pulse are tabulated in row 2 of the table. Vertical gray shading indicates the segment duration for each pulse, where $dt = 0.22$ ns is the minimum sample time permitted by the IBM *Valencia* backend. (c), (g), (k), (o), (s) Fourier analysis of the I and Q waveforms for each pulse plotted above. Decay in pulse spectral features is consistent with low-pass sinc filter with $f_{\text{cutoff}} = 30$ MHz (vertical black lines). Spectra are overlaid with leakage transitions associated with detunings from higher-level states of neighboring qubits [52], showing spectral overlap comparable with the default pulse. For qubit indexes $i \in \{0, 1, 2, 3, 4\}$ on *Valencia*, these transitions are defined as $\delta_{1,i} \equiv (\omega_i - \omega_1)/2\pi$ and $\delta_{2,i} \equiv [2(\omega_i - \omega_1) - \chi_i]/2\pi$, where ω_i and χ_i are respectively the resonant frequency and anharmonicities for qubit i . (d), (h), (l), (p), (t) Log-scale density plots showing robustness properties for each pulse. Dephasing and amplitude robustness is indicated by dark regions in the two-dimensional domain corresponding to low infidelities while scanning over these errors. Infidelities are calculated assuming a two-level system model incorporating offset detuning and amplitude errors, plotted as percentages of the nominal qubit frequency (about 5 GHz) and ideal per-segment I and Q amplitudes, respectively. Q-CTRL solutions suppress these errors by orders of magnitude compared to the default pulse. Dephasing and amplitude robustness is seen by the extended regions of low infidelity along these axes. The dual-robust solution plotted in panel (r) shows robustness in both directions. These are to be compared with the IBM default DRAG pulse in panel (d), showing a sharp increase in infidelity with both these errors.

TABLE I. Rotation error (ϵ_r), population-decay error per gate (β), and total effective error per gate (ϵ) for all qubits and experimental runs from 4 to 9 August 2020 with serial application of the $R_x(\pi)$ gate using the IBM default pulse. The last two columns respectively contain the total error averaged over the qubits $\langle \epsilon \rangle_q$ and their deviations σ_q for each day. The bottom two rows show the time-averaged errors $\langle \epsilon \rangle_t$ for each qubit and their deviations σ_t , respectively. The total average, $\langle \epsilon \rangle = \langle \epsilon \rangle_{q,t}$, as well as the averages of the deviations, $\langle \sigma_t \rangle_q$ and $\langle \sigma_q \rangle_t$, are shown at the bottom right of the table.

		Qubit					System-averaged data	
		0	1	2	3	4	$\langle \epsilon \rangle_q$	σ_q
August 4	ϵ_r	5.47×10^{-3}	1.21×10^{-2}	2.96×10^{-3}	6.14×10^{-3}	5.78×10^{-3}	7.21×10^{-3}	2.69×10^{-3}
	β	1.05×10^{-3}	3.03×10^{-3}	4.28×10^{-3}	9.88×10^{-4}	3.03×10^{-3}		
	ϵ	5.57×10^{-3}	1.25×10^{-2}	5.20×10^{-3}	6.22×10^{-3}	6.52×10^{-3}		
August 5	ϵ_r	1.08×10^{-2}	4.89×10^{-3}	1.69×10^{-2}	5.84×10^{-3}	1.60×10^{-2}	1.14×10^{-2}	4.76×10^{-3}
	β	3.34×10^{-3}	3.58×10^{-3}	2.65×10^{-3}	2.25×10^{-3}	3.84×10^{-3}		
	ϵ	1.13×10^{-2}	6.06×10^{-3}	1.71×10^{-2}	6.25×10^{-3}	1.64×10^{-2}		
August 6	ϵ_r	8.36×10^{-3}	3.80×10^{-3}	7.71×10^{-3}	3.58×10^{-3}	1.37×10^{-2}	8.10×10^{-3}	3.60×10^{-3}
	β	2.85×10^{-3}	3.48×10^{-3}	3.94×10^{-3}	1.15×10^{-3}	3.44×10^{-3}		
	ϵ	8.83×10^{-3}	5.15×10^{-3}	8.66×10^{-3}	3.76×10^{-3}	1.41×10^{-2}		
August 7	ϵ_r	7.62×10^{-3}	5.68×10^{-3}	8.71×10^{-3}	3.29×10^{-3}	1.83×10^{-2}	9.63×10^{-3}	5.42×10^{-3}
	β	2.96×10^{-3}	3.88×10^{-3}	3.80×10^{-3}	1.91×10^{-3}	7.47×10^{-3}		
	ϵ	8.18×10^{-3}	6.88×10^{-3}	9.50×10^{-3}	3.80×10^{-3}	1.98×10^{-2}		
August 8	ϵ_r	3.36×10^{-3}	7.46×10^{-3}	3.87×10^{-3}	3.49×10^{-3}	1.04×10^{-2}	6.52×10^{-3}	3.64×10^{-3}
	β	2.03×10^{-3}	2.77×10^{-3}	9.05×10^{-4}	1.16×10^{-3}	7.90×10^{-3}		
	ϵ	3.93×10^{-3}	7.95×10^{-3}	3.98×10^{-3}	3.68×10^{-3}	1.31×10^{-2}		
August 9	ϵ_r	3.41×10^{-3}	5.89×10^{-3}	2.16×10^{-3}	4.77×10^{-3}	6.62×10^{-3}	5.51×10^{-3}	2.40×10^{-3}
	β	1.66×10^{-3}	1.58×10^{-3}	2.00×10^{-3}	1.16×10^{-3}	7.23×10^{-3}		
	ϵ	3.79×10^{-3}	6.10×10^{-3}	2.94×10^{-3}	4.91×10^{-3}	9.80×10^{-3}		
Time averaged	$\langle \epsilon \rangle_t$	6.94×10^{-3}	7.44×10^{-3}	7.90×10^{-3}	4.77×10^{-3}	1.33×10^{-2}	Total averages	
	σ_t	2.74×10^{-3}	2.42×10^{-3}	4.76×10^{-3}	1.12×10^{-3}	4.30×10^{-3}	$\langle \epsilon \rangle = 8.07 \times 10^{-3}$	$\langle \sigma_t \rangle_q = 3.07 \times 10^{-3}$
							$\langle \sigma_q \rangle_t = 3.75 \times 10^{-3}$	

optimized pulses demonstrated here target robustness to these primary errors and do not target leakage suppression, although this could be easily included in our optimization framework. However, owing to hardware limitations in pulse amplitude, optimizing pulses with robustness to amplitude and/or detuning errors that simultaneously suppress leakage would come at the cost of increased pulse duration, further offsetting the benefit of leakage suppression against the overhead of increased T_2 errors.

Since completing these experiments we have identified $\hat{\sigma}_z \hat{\sigma}_z$ couplings between certain qubit pairs that are not tabulated in backend properties and are accordingly not taken into account for our model-based approach to pulse optimization. We have also identified other minor discrepancies between IBM documentation for backend hardware parameters and observed values determined through system identification routines (e.g., sign and magnitude of tabulated qubit anharmonicities). In response, we have deployed autonomous reinforcement learning via Qiskit Pulse to discover optimized pulses without the need for complete models of the system Hamiltonian. Initial results finding high-fidelity optimized pulses are exceptionally promising and will be the subject of a future publication. Combining complete system identification and online

optimization to compensate for unaccounted Hamiltonian terms provides a pathway to dramatically expand the performance of these solutions. Overall, we hope that these demonstrations open opportunities for complex low-level control strategies to augment the performance of near-term quantum computers.

ACKNOWLEDGMENTS

We acknowledge the use of IBM Quantum services for this work. The views expressed are those of the authors, and do not reflect the official policy or position of IBM or the IBM Quantum team. The authors also acknowledge Nathan Earnest-Noble for technical discussions and his support on using Qiskit Pulse on IBM Quantum devices.

APPENDIX A: FROBENIUS INNER PRODUCT AND FROBENIUS NORM

For matrices $A, B \in \mathbb{C}^{m \times n}$, the Frobenius inner product is defined as

$$\langle A, B \rangle_F = \sum_{ij} A_{ij}^* B_{ij} = \text{Tr}(A^\dagger B). \quad (\text{A1})$$

TABLE II. Same as in Table I but for the amplitude-robust pulse.

		Qubit					System-averaged data	
		0	1	2	3	4	$\langle \epsilon \rangle_q$	σ_q
August 4	ϵ_r	2.13×10^{-3}	2.58×10^{-8}	2.25×10^{-8}	2.78×10^{-4}	3.28×10^{-3}	3.39×10^{-3}	2.07×10^{-3}
	β	1.54×10^{-3}	1.79×10^{-3}	7.30×10^{-3}	1.65×10^{-3}	1.43×10^{-3}		
	ϵ	2.63×10^{-3}	1.79×10^{-3}	7.30×10^{-3}	1.67×10^{-3}	3.58×10^{-3}		
August 5	ϵ_r	1.16×10^{-7}	7.11×10^{-4}	1.33×10^{-3}	2.14×10^{-8}	2.38×10^{-3}	2.10×10^{-3}	8.25×10^{-4}
	β	1.41×10^{-3}	1.55×10^{-3}	1.79×10^{-3}	1.52×10^{-3}	2.77×10^{-3}		
	ϵ	1.41×10^{-3}	1.70×10^{-3}	2.23×10^{-3}	1.52×10^{-3}	3.66×10^{-3}		
August 6	ϵ_r	1.57×10^{-3}	1.52×10^{-8}	1.52×10^{-3}	1.17×10^{-8}	2.55×10^{-3}	2.13×10^{-3}	5.71×10^{-4}
	β	8.38×10^{-4}	1.61×10^{-3}	1.55×10^{-3}	1.86×10^{-3}	1.94×10^{-3}		
	ϵ	1.78×10^{-3}	1.61×10^{-3}	2.18×10^{-3}	1.86×10^{-3}	3.21×10^{-3}		
August 7	ϵ_r	4.63×10^{-3}	1.09×10^{-8}	6.58×10^{-8}	1.14×10^{-3}	1.47×10^{-8}	3.13×10^{-3}	1.26×10^{-3}
	β	1.06×10^{-3}	1.67×10^{-3}	2.84×10^{-3}	1.57×10^{-3}	4.45×10^{-3}		
	ϵ	4.75×10^{-3}	1.67×10^{-3}	2.84×10^{-3}	1.94×10^{-3}	4.45×10^{-3}		
August 8	ϵ_r	3.45×10^{-3}	7.26×10^{-8}	1.09×10^{-7}	4.40×10^{-8}	1.21×10^{-7}	3.05×10^{-3}	1.14×10^{-3}
	β	9.39×10^{-4}	1.51×10^{-3}	3.77×10^{-3}	1.92×10^{-3}	4.50×10^{-3}		
	ϵ	3.57×10^{-3}	1.51×10^{-3}	3.77×10^{-3}	1.92×10^{-3}	4.50×10^{-3}		
August 9	ϵ_r	1.16×10^{-3}	9.28×10^{-8}	8.17×10^{-9}	2.04×10^{-3}	6.32×10^{-8}	2.69×10^{-3}	1.31×10^{-3}
	β	1.84×10^{-3}	1.87×10^{-3}	1.92×10^{-3}	7.65×10^{-4}	5.30×10^{-3}		
	ϵ	2.18×10^{-3}	1.87×10^{-3}	1.92×10^{-3}	2.18×10^{-3}	5.30×10^{-3}		
Time averaged	$\langle \epsilon \rangle_t$	2.72×10^{-3}	1.69×10^{-3}	3.37×10^{-3}	1.85×10^{-3}	4.11×10^{-3}	Total averages	
	σ_t	1.14×10^{-3}	1.18×10^{-4}	1.86×10^{-3}	2.07×10^{-4}	7.04×10^{-4}	$\langle \epsilon \rangle = 2.75 \times 10^{-3}$	
							$\langle \sigma_t \rangle_q = 8.05 \times 10^{-4}$	
							$\langle \sigma_q \rangle_t = 1.20 \times 10^{-3}$	

TABLE III. Same as in Table I but for the dephasing-robust pulse.

		Qubit					System-averaged data	
		0	1	2	3	4	$\langle \epsilon \rangle_q$	σ_q
August 4	ϵ_r	8.53×10^{-8}	3.16×10^{-8}	1.27×10^{-10}	8.30×10^{-8}	7.40×10^{-8}	1.92×10^{-3}	1.29×10^{-3}
	β	1.04×10^{-3}	1.41×10^{-3}	4.43×10^{-3}	1.70×10^{-3}	9.92×10^{-4}		
	ϵ	1.04×10^{-3}	1.41×10^{-3}	4.43×10^{-3}	1.70×10^{-3}	9.92×10^{-4}		
August 5	ϵ_r	1.92×10^{-3}	3.47×10^{-3}	1.60×10^{-3}	2.07×10^{-3}	2.49×10^{-3}	2.75×10^{-3}	5.81×10^{-4}
	β	7.25×10^{-4}	1.47×10^{-3}	1.73×10^{-3}	1.95×10^{-3}	1.17×10^{-3}		
	ϵ	2.06×10^{-3}	3.77×10^{-3}	2.36×10^{-3}	2.84×10^{-3}	2.75×10^{-3}		
August 6	ϵ_r	1.93×10^{-3}	5.08×10^{-3}	2.44×10^{-8}	3.04×10^{-9}	7.51×10^{-4}	2.66×10^{-3}	1.42×10^{-3}
	β	5.86×10^{-4}	1.46×10^{-3}	2.96×10^{-3}	1.71×10^{-3}	1.07×10^{-3}		
	ϵ	2.02×10^{-3}	5.29×10^{-3}	2.96×10^{-3}	1.71×10^{-3}	1.31×10^{-3}		
August 7	ϵ_r	9.60×10^{-4}	5.05×10^{-3}	4.79×10^{-8}	3.81×10^{-8}	1.26×10^{-8}	2.88×10^{-3}	1.42×10^{-3}
	β	1.26×10^{-3}	1.46×10^{-3}	3.49×10^{-3}	1.32×10^{-3}	2.76×10^{-3}		
	ϵ	1.59×10^{-3}	5.26×10^{-3}	3.49×10^{-3}	1.32×10^{-3}	2.76×10^{-3}		
August 8	ϵ_r	1.25×10^{-3}	4.44×10^{-3}	3.71×10^{-7}	2.81×10^{-8}	2.55×10^{-3}	3.20×10^{-3}	1.53×10^{-3}
	β	9.60×10^{-4}	1.79×10^{-3}	3.88×10^{-3}	1.16×10^{-3}	3.83×10^{-3}		
	ϵ	1.58×10^{-3}	4.79×10^{-3}	3.88×10^{-3}	1.16×10^{-3}	4.60×10^{-3}		
August 9	ϵ_r	2.63×10^{-3}	4.89×10^{-3}	7.83×10^{-4}	6.03×10^{-4}	5.80×10^{-8}	3.04×10^{-3}	1.22×10^{-3}
	β	1.53×10^{-3}	1.00×10^{-3}	1.92×10^{-3}	1.38×10^{-3}	3.61×10^{-3}		
	ϵ	3.04×10^{-3}	4.99×10^{-3}	2.07×10^{-3}	1.51×10^{-3}	3.61×10^{-3}		
Time averaged	$\langle \epsilon \rangle_t$	1.89×10^{-3}	4.25×10^{-3}	3.20×10^{-3}	1.71×10^{-3}	2.67×10^{-3}	Total averages	
	σ_t	6.16×10^{-4}	1.37×10^{-3}	8.27×10^{-4}	5.44×10^{-4}	1.25×10^{-3}	$\langle \epsilon \rangle = 2.74 \times 10^{-3}$	
							$\langle \sigma_t \rangle_q = 9.20 \times 10^{-4}$	
							$\langle \sigma_q \rangle_t = 1.24 \times 10^{-3}$	

TABLE IV. Same as in Table I but for the dual-robust pulse.

		Qubit					System-averaged data	
		0	1	2	3	4	$\langle \epsilon \rangle_q$	σ_q
August 4	ϵ_r	2.07×10^{-8}	6.62×10^{-8}	5.62×10^{-8}	4.81×10^{-8}	3.61×10^{-8}	2.50×10^{-3}	1.23×10^{-3}
	β	1.47×10^{-3}	2.01×10^{-3}	4.92×10^{-3}	2.08×10^{-3}	2.00×10^{-3}		
	ϵ	1.47×10^{-3}	2.01×10^{-3}	4.92×10^{-3}	2.08×10^{-3}	2.00×10^{-3}		
August 5	ϵ_r	5.13×10^{-8}	1.08×10^{-3}	2.25×10^{-3}	5.13×10^{-8}	1.16×10^{-7}	2.51×10^{-3}	7.98×10^{-4}
	β	1.75×10^{-3}	1.59×10^{-3}	2.97×10^{-3}	1.95×10^{-3}	3.19×10^{-3}		
	ϵ	1.75×10^{-3}	1.92×10^{-3}	3.73×10^{-3}	1.95×10^{-3}	3.19×10^{-3}		
August 6	ϵ_r	1.08×10^{-3}	1.33×10^{-3}	9.33×10^{-4}	2.75×10^{-8}	3.11×10^{-8}	2.15×10^{-3}	3.31×10^{-4}
	β	2.23×10^{-3}	2.04×10^{-3}	1.91×10^{-3}	2.16×10^{-3}	1.56×10^{-3}		
	ϵ	2.48×10^{-3}	2.44×10^{-3}	2.13×10^{-3}	2.16×10^{-3}	1.56×10^{-3}		
August 7	ϵ_r	1.44×10^{-3}	1.06×10^{-3}	3.49×10^{-8}	2.36×10^{-3}	7.29×10^{-9}	2.71×10^{-3}	5.42×10^{-4}
	β	1.89×10^{-3}	1.74×10^{-3}	2.56×10^{-3}	1.86×10^{-3}	3.60×10^{-3}		
	ϵ	2.37×10^{-3}	2.04×10^{-3}	2.56×10^{-3}	3.01×10^{-3}	3.60×10^{-3}		
August 8	ϵ_r	3.92×10^{-8}	5.64×10^{-8}	3.30×10^{-3}	3.97×10^{-9}	5.07×10^{-8}	3.07×10^{-3}	1.66×10^{-3}
	β	1.30×10^{-3}	2.06×10^{-3}	1.50×10^{-3}	2.35×10^{-3}	6.04×10^{-3}		
	ϵ	1.30×10^{-3}	2.06×10^{-3}	3.62×10^{-3}	2.35×10^{-3}	6.04×10^{-3}		
August 9	ϵ_r	1.57×10^{-8}	2.30×10^{-8}	2.07×10^{-3}	1.21×10^{-3}	1.96×10^{-7}	3.70×10^{-3}	2.27×10^{-3}
	β	2.60×10^{-3}	2.53×10^{-3}	1.98×10^{-3}	1.90×10^{-3}	8.74×10^{-3}		
	ϵ	2.60×10^{-3}	2.53×10^{-3}	2.86×10^{-3}	2.25×10^{-3}	8.74×10^{-3}		
Time averaged	$\langle \epsilon \rangle_t$	2.00×10^{-3}	2.17×10^{-3}	3.30×10^{-3}	2.30×10^{-3}	4.10×10^{-3}	Total averages	
	σ_t	5.10×10^{-4}	2.31×10^{-4}	9.15×10^{-4}	3.40×10^{-4}	2.34×10^{-3}	$\langle \epsilon \rangle = 2.77 \times 10^{-3}$	
							$\langle \sigma_t \rangle_q = 8.67 \times 10^{-4}$ $\langle \sigma_q \rangle_t = 1.14 \times 10^{-3}$	

The inner product in Eq. (A1) induces a matrix norm. For a matrix $A \in \mathbb{C}^{m \times n}$, the Frobenius norm is defined by

$$\|A\|_F = \sqrt{\langle A, A \rangle_F} = \sqrt{\sum_{ij} |A_{ij}|^2} = \sqrt{\text{Tr}(A^\dagger A)}. \quad (\text{A2})$$

APPENDIX B: COMPARISON BETWEEN FITTED AND HARDWARE REPORTED T_1

At every daily hardware calibration, the decay time T_1 for each qubit in the backend is reported. These values

TABLE V. Same as in Table I but for the case of parallel gate implementation. All fittings used Eq. (14) except for qubit 2 on 7 August 2020 where Eq. (13) provided a better fit.

		Qubit					System-averaged data	
		0	1	2	3	4	$\langle \epsilon \rangle_q$	σ_q
August 4	ϵ_r	1.92×10^{-3}	2.04×10^{-3}	1.79×10^{-3}	2.24×10^{-3}	1.79×10^{-3}	1.42×10^{-2}	1.86×10^{-3}
	β	6.37×10^{-3}	6.69×10^{-3}	5.87×10^{-3}	6.99×10^{-3}	6.17×10^{-3}		
	ϵ	1.26×10^{-2}	1.64×10^{-2}	1.59×10^{-2}	1.16×10^{-2}	1.43×10^{-2}		
August 5	ϵ_r	2.50×10^{-3}	2.46×10^{-3}	3.16×10^{-3}	3.04×10^{-3}	2.24×10^{-3}	1.41×10^{-2}	2.57×10^{-3}
	β	5.13×10^{-3}	1.46×10^{-2}	1.39×10^{-2}	9.91×10^{-3}	1.06×10^{-2}		
	ϵ	1.39×10^{-2}	1.46×10^{-2}	1.39×10^{-2}	9.91×10^{-3}	1.80×10^{-2}		
August 6	ϵ_r	1.86×10^{-3}	2.15×10^{-3}	2.46×10^{-3}	2.23×10^{-3}	1.89×10^{-3}	1.36×10^{-2}	2.67×10^{-3}
	β	1.09×10^{-2}	1.48×10^{-2}	1.32×10^{-2}	1.01×10^{-2}	8.76×10^{-3}		
	ϵ	1.09×10^{-2}	1.48×10^{-2}	1.32×10^{-2}	1.01×10^{-2}	1.80×10^{-2}		
August 7	ϵ_r	1.66×10^{-3}	2.68×10^{-3}	3.18×10^{-3}	4.03×10^{-3}	3.62×10^{-3}	1.37×10^{-2}	3.70×10^{-3}
	β	9.80×10^{-3}	1.35×10^{-2}	1.67×10^{-2}	9.47×10^{-3}	9.14×10^{-3}		
	ϵ	9.80×10^{-3}	1.35×10^{-2}	1.86×10^{-2}	9.47×10^{-3}	1.71×10^{-2}		
Time averaged	$\langle \epsilon \rangle_t$	1.21×10^{-2}	1.48×10^{-2}	1.54×10^{-2}	1.03×10^{-2}	1.68×10^{-2}	Total averages	
	σ_t	1.50×10^{-3}	1.07×10^{-3}	2.07×10^{-3}	8.06×10^{-4}	1.50×10^{-3}	$\langle \epsilon \rangle = 1.39 \times 10^{-2}$	
							$\langle \sigma_t \rangle_q = 1.39 \times 10^{-3}$ $\langle \sigma_q \rangle_t = 2.70 \times 10^{-3}$	

TABLE VI. Same as in Table I but for the case of parallel gate implementation of the amplitude-robust pulse.

		Qubit					System-averaged data	
		0	1	2	3	4	$\langle\epsilon\rangle_q$	σ_q
August 4	ϵ_r	1.68×10^{-3}	4.04×10^{-9}	7.36×10^{-8}	1.55×10^{-3}	9.43×10^{-8}	1.95×10^{-3}	1.69×10^{-4}
	β	9.31×10^{-4}	2.04×10^{-3}	1.79×10^{-3}	1.61×10^{-3}	1.79×10^{-3}		
	ϵ	1.92×10^{-3}	2.04×10^{-3}	1.79×10^{-3}	2.24×10^{-3}	1.79×10^{-3}		
August 5	ϵ_r	2.35×10^{-3}	9.54×10^{-4}	7.52×10^{-9}	2.63×10^{-3}	1.62×10^{-9}	2.68×10^{-3}	3.57×10^{-4}
	β	8.65×10^{-4}	2.27×10^{-3}	3.16×10^{-3}	1.54×10^{-3}	2.24×10^{-3}		
	ϵ	2.50×10^{-3}	2.46×10^{-3}	3.16×10^{-3}	3.04×10^{-3}	2.24×10^{-3}		
August 6	ϵ_r	1.49×10^{-3}	1.58×10^{-7}	7.48×10^{-8}	9.33×10^{-4}	1.97×10^{-8}	2.12×10^{-3}	2.25×10^{-4}
	β	1.11×10^{-3}	2.15×10^{-3}	2.46×10^{-3}	2.03×10^{-3}	1.89×10^{-3}		
	ϵ	1.86×10^{-3}	2.15×10^{-3}	2.46×10^{-3}	2.23×10^{-3}	1.89×10^{-3}		
August 7	ϵ_r	1.18×10^{-3}	1.41×10^{-3}	1.02×10^{-7}	3.79×10^{-3}	1.72×10^{-7}	3.03×10^{-3}	8.23×10^{-4}
	β	1.16×10^{-3}	2.28×10^{-3}	3.18×10^{-3}	1.37×10^{-3}	3.62×10^{-3}		
	ϵ	1.66×10^{-3}	2.68×10^{-3}	3.18×10^{-3}	4.03×10^{-3}	3.62×10^{-3}		
Time averaged	$\langle\epsilon\rangle_t$	1.99×10^{-3}	2.33×10^{-3}	2.65×10^{-3}	2.89×10^{-3}	2.39×10^{-3}	Total averages	
	σ_t	3.15×10^{-4}	2.53×10^{-4}	5.75×10^{-4}	7.40×10^{-4}	7.34×10^{-4}	$\langle\epsilon\rangle = 2.45 \times 10^{-3}$	
							$\langle\sigma_t\rangle_q = 5.23 \times 10^{-4}$	
							$\langle\sigma_q\rangle_t = 3.93 \times 10^{-4}$	

are shown as gray circles in Fig. 8 for the whole period of our August experiments on *Valencia*. For the IBM default pulse, the fitted decay times substantially underestimate the reported backend values across the device. By contrast, the fitted values extracted with robust pulses approximately agree with tabulated T_1 for most qubits and experiments (Fig. 8, top row). Agreement gets worse with parallel implementation of pulses, indicating the existence of an extra decaying process.

APPENDIX C: OPTIMIZED PULSES IMPLEMENTED ON VALENCIA

The default high-fidelity gates on the IBM-Q hardware are obtained through a combination of DRAG pulses and virtual Z rotations [51]. In Qiskit, $R_x(\pi)$ and $R_x(\pi/2)$ pulses of this kind are called with the following commands.

TABLE VII. Same as in Table I but for the case of parallel gate implementation of the dephasing-robust pulse.

		Qubit					System-averaged data	
		0	1	2	3	4	$\langle\epsilon\rangle_q$	σ_q
August 4	ϵ_r	2.68×10^{-3}	2.72×10^{-3}	2.43×10^{-3}	3.03×10^{-3}	3.20×10^{-3}	3.32×10^{-3}	1.09×10^{-4}
	β	1.76×10^{-3}	2.07×10^{-3}	2.07×10^{-3}	1.32×10^{-3}	1.34×10^{-3}		
	ϵ	3.20×10^{-3}	3.42×10^{-3}	3.20×10^{-3}	3.30×10^{-3}	3.46×10^{-3}		
August 5	ϵ_r	3.04×10^{-3}	2.76×10^{-3}	2.47×10^{-3}	3.13×10^{-3}	2.43×10^{-3}	3.29×10^{-3}	1.49×10^{-4}
	β	1.28×10^{-3}	1.94×10^{-3}	2.26×10^{-3}	1.40×10^{-3}	1.77×10^{-3}		
	ϵ	3.30×10^{-3}	3.38×10^{-3}	3.34×10^{-3}	3.43×10^{-3}	3.00×10^{-3}		
August 6	ϵ_r	2.32×10^{-3}	2.78×10^{-3}	2.21×10^{-3}	2.33×10^{-3}	2.53×10^{-3}	2.81×10^{-3}	2.01×10^{-4}
	β	9.59×10^{-4}	1.42×10^{-3}	1.77×10^{-3}	1.36×10^{-3}	1.36×10^{-3}		
	ϵ	2.51×10^{-3}	3.12×10^{-3}	2.83×10^{-3}	2.70×10^{-3}	2.87×10^{-3}		
August 7	ϵ_r	1.50×10^{-3}	2.37×10^{-3}	2.84×10^{-8}	3.32×10^{-3}	1.74×10^{-8}	3.21×10^{-3}	7.62×10^{-4}
	β	1.06×10^{-3}	2.38×10^{-3}	4.10×10^{-3}	1.53×10^{-3}	3.11×10^{-3}		
	ϵ	1.84×10^{-3}	3.36×10^{-3}	4.10×10^{-3}	3.66×10^{-3}	3.11×10^{-3}		
Time averaged	$\langle\epsilon\rangle_t$	2.71×10^{-3}	3.32×10^{-3}	3.37×10^{-3}	3.27×10^{-3}	3.11×10^{-3}	Total averages	
	σ_t	5.88×10^{-4}	1.15×10^{-4}	4.61×10^{-4}	3.54×10^{-4}	2.21×10^{-4}	$\langle\epsilon\rangle = 3.16 \times 10^{-3}$	
							$\langle\sigma_t\rangle_q = 3.48 \times 10^{-4}$	
							$\langle\sigma_q\rangle_t = 3.05 \times 10^{-4}$	

TABLE VIII. Same as in Table I but for the case of parallel gate implementation of the dual-robust pulse.

		Qubit					System-averaged data	
		0	1	2	3	4	$\langle \epsilon \rangle_q$	σ_q
August 4	ϵ_r	3.79×10^{-8}	7.85×10^{-8}	6.93×10^{-8}	6.53×10^{-3}	6.07×10^{-3}	5.81×10^{-3}	1.46×10^{-3}
	β	3.10×10^{-3}	6.03×10^{-3}	5.77×10^{-3}	3.22×10^{-3}	3.16×10^{-3}		
	ϵ	3.10×10^{-3}	6.03×10^{-3}	5.77×10^{-3}	7.28×10^{-3}	6.84×10^{-3}		
August 5	ϵ_r	5.43×10^{-9}	1.04×10^{-7}	4.16×10^{-8}	6.98×10^{-3}	5.96×10^{-3}	5.84×10^{-3}	1.82×10^{-3}
	β	2.79×10^{-3}	5.82×10^{-3}	5.22×10^{-3}	3.64×10^{-3}	4.56×10^{-3}		
	ϵ	2.79×10^{-3}	5.82×10^{-3}	5.22×10^{-3}	7.87×10^{-3}	7.50×10^{-3}		
August 6	ϵ_r	4.66×10^{-8}	1.13×10^{-7}	2.56×10^{-8}	6.45×10^{-3}	6.50×10^{-3}	5.54×10^{-3}	1.64×10^{-3}
	β	2.76×10^{-3}	5.37×10^{-3}	5.16×10^{-3}	3.13×10^{-3}	3.21×10^{-3}		
	ϵ	2.76×10^{-3}	5.37×10^{-3}	5.16×10^{-3}	7.17×10^{-3}	7.25×10^{-3}		
August 7	ϵ_r	1.96×10^{-7}	9.77×10^{-9}	3.37×10^{-8}	6.87×10^{-3}	6.05×10^{-3}	6.63×10^{-3}	1.96×10^{-3}
	β	2.91×10^{-3}	6.43×10^{-3}	7.81×10^{-3}	3.43×10^{-3}	5.68×10^{-3}		
	ϵ	2.91×10^{-3}	6.43×10^{-3}	7.81×10^{-3}	7.68×10^{-3}	8.30×10^{-3}		
Time averaged	$\langle \epsilon \rangle_t$	2.89×10^{-3}	5.91×10^{-3}	5.99×10^{-3}	7.50×10^{-3}	7.47×10^{-3}	Total averages	
	σ_t	1.32×10^{-4}	3.84×10^{-4}	1.08×10^{-3}	2.85×10^{-4}	5.31×10^{-4}	$\langle \epsilon \rangle = 5.95 \times 10^{-3}$	
							$\langle \sigma_t \rangle_q = 4.82 \times 10^{-4}$ $\langle \sigma_q \rangle_t = 1.72 \times 10^{-3}$	

```

defaults = backend.defaults()
inst_map = defaults.instruction_
schedule_map
R_x_pi = inst_map.get('u3', [qubit],
P0=np.pi, P1=-np.pi/2, P2=np.pi/2)
R_x_pi_2 = inst_map.get('u3', [qubit],
P0=np.pi/2, P1=-np.pi/2, P2=np.pi/2)

```

We note that the IBM backend performs modifications of analytically defined DRAG pulses for each qubit in a manner that is undisclosed. Throughout this work “default” pulses are called in this way without further calibration or modification. Optimized pulses used throughout this work are described in detail in Fig. 9.

APPENDIX D: SUPPLEMENTARY DATA SETS

In this appendix we provide tabulated data for the rotation error ϵ_r , decay constant β , and total error ϵ extracted from experiments summarized in Figs. 5 and 7. For each day on which measurements are performed, and for each qubit, we additionally calculate average device performance and variance over both time and across devices. In Tables I–VIII we present the experimental data for both serialized operations and parallelized operations subject to crosstalk.

- [1] F. Arute *et al.*, Quantum supremacy using a programmable superconducting processor, *Nature* **574**, 505 (2019).
- [2] J. Preskill, Quantum computing in the NISQ era and beyond, *Quantum* **2**, 79 (2018).
- [3] M. H. Devoret and R. J. Schoelkopf, Superconducting circuits for quantum information: An outlook, *Science* **339**, 1169 (2013).

- [4] G. Wendin, Quantum information processing with superconducting circuits: A review, *Rep. Progress Phys.* **80**, 106001 (2017).
- [5] 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).
- [6] G. Aleksandrowicz *et al.*, Qiskit: An Open-source Framework for Quantum Computing (2019).
- [7] Rigetti forest, GitHub (accessed: 26-May-2020).
- [8] A python framework for creating, editing, and invoking Noisy Intermediate Scale Quantum (NISQ) circuits, GitHub (accessed: 26-May-2020).
- [9] Microsoft quantum development kit, GitHub (accessed: 26-May-2020).
- [10] 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, [arXiv:1809.03452 \[quant-ph\]](https://arxiv.org/abs/1809.03452) (2018).
- [11] P. Gokhale, A. Javadi-Abhari, N. Earnest, Y. Shi, and F. T. Chong, Optimized quantum compilation for near-term algorithms with openpulse, [arXiv:2004.11205 \[quant-ph\]](https://arxiv.org/abs/2004.11205) (2020).
- [12] L. Viola and S. Lloyd, Dynamical suppression of decoherence in two-state quantum systems, *Phys. Rev. A* **58**, 2733 (1998).
- [13] R. H. Byrd, P. Lu, J. Nocedal, and C. Zhu, A limited memory algorithm for bound constrained optimization, *SIAM J. Sci. Comput.* **16**, 1190 (1995).
- [14] G. Gordon, G. Kurizki, and D. A. Lidar, Optimal Dynamical Decoherence Control of a Qubit, *Phys. Rev. Lett.* **101**, 010403 (2008).
- [15] K. Khodjasteh and L. Viola, Dynamically Error-Corrected Gates for Universal Quantum Computation, *Phys. Rev. Lett.* **102**, 080501 (2009).

- [16] A. Soare, H. Ball, D. Hayes, J. Sastrawan, M. C. Jarratt, J. J. McLoughlin, X. Zhen, T. J. Green, and M. J. Biercuk, Experimental noise filtering by quantum control, *Nat. Phys.* **10**, 825 (2014).
- [17] F. K. Wilhelm, S. Kirchhoff, S. Machnes, N. Wittler, and D. Sugny, An introduction into optimal control for quantum technologies, [arXiv:2003.10132 \[quant-ph\]](https://arxiv.org/abs/2003.10132) (2020).
- [18] H. Ball, M. Biercuk, A. Carvalho, J. Chen, M. R. Hush, L. A. de Castro, L. Li, P. J. Liebermann, H. 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 Science and Technology* (2021).
- [19] H. K. Ng, D. A. Lidar, and J. Preskill, Combining dynamical decoupling with fault-tolerant quantum computation, *Phys. Rev. A* **84**, 012305 (2011).
- [20] K. Khodjasteh and D. A. Lidar, Fault-Tolerant Quantum Dynamical Decoupling, *Phys. Rev. Lett.* **95**, 180501 (2005).
- [21] K. Khodjasteh and D. A. Lidar, Rigorous bounds on the performance of a hybrid dynamical-decoupling quantum-computing scheme, *Phys. Rev. A* **78**, 012355 (2008).
- [22] N. C. Jones, R. Van Meter, A. G. Fowler, P. L. McMahon, J. Kim, T. D. Ladd, and Y. Yamamoto, Layered Architecture for Quantum Computing, *Phys. Rev. X* **2**, 031007 (2012).
- [23] C. L. Edmunds, C. Hempel, R. J. Harris, V. Frey, T. M. Stace, and M. J. Biercuk, Dynamically corrected gates suppressing spatiotemporal error correlations as measured by randomized benchmarking, *Phys. Rev. Res.* **2**, 013156 (2020).
- [24] J. T. Merrill and K. R. Brown, in *Quantum Information and Computation for Chemistry* (John Wiley & Sons, Ltd, Hoboken, New Jersey, 2014), p. 241.
- [25] 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).
- [26] 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).
- [27] 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, [arXiv:2003.05952 \[quant-ph\]](https://arxiv.org/abs/2003.05952) (2020).
- [28] J. L. Allen, R. Kosut, and E. Ginossar, Minimal time robust two qubit gates in circuit qed, [arXiv:1902.08056 \[quant-ph\]](https://arxiv.org/abs/1902.08056) (2019).
- [29] A. W. Cross and J. M. Gambetta, Optimized pulse shapes for a resonator-induced phase gate, *Phys. Rev. A* **91**, 032325 (2015).
- [30] Optimization, *Q-CTRL User Guide* (accessed: 30-Sep-2020).
- [31] Qiskit pulse api reference, *Qiskit Reference Documentation* (accessed: 30-Sep-2020).
- [32] 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).
- [33] Qiskit terra api reference: Drag, *Qiskit Reference Documentation* (accessed: 9-Oct-2020).
- [34] W. D. Oliver, in *Quantum information processing : lecture notes of the 44th IFF Spring School 2013*, edited by D. P. DiVincenzo (Forschungszentrum Jülich, 2013).
- [35] H. Ball, T. M. Stace, S. T. Flammia, and M. J. Biercuk, Effect of noise correlations on randomized benchmarking, *Phys. Rev. A* **93**, 022303 (2016).
- [36] T. Proctor, K. Rudinger, K. Young, M. Sarovar, and R. Blume-Kohout, What Randomized Benchmarking Actually Measures, *Phys. Rev. Lett.* **119**, 130502 (2017).
- [37] (2020), [ibmq_armonk v1.1.1](https://ibmq_armonk.v1.1.1.ibm.com), IBM Quantum team. Retrieved from <https://quantum-computing.ibm.com>.
- [38] (2020), [ibmq_valencia v1.3.1](https://ibmq_valencia.v1.3.1.ibm.com), IBM Quantum team. Retrieved from <https://quantum-computing.ibm.com>.
- [39] Control hardware: pulse optimization under realistic experimental constraints, *Q-CTRL Application Note* (accessed: 22-Jun-2020).
- [40] Superconducting qubits: improving the performance of single qubit gates, *Q-CTRL Application Note* (accessed: 11-May-2020).
- [41] Qiskit ignis api reference, *Qiskit Reference Documentation* (accessed: 30-Sep-2020).
- [42] P. V. Klimov *et al.*, Fluctuations of Energy-Relaxation Times in Superconducting Qubits, *Phys. Rev. Lett.* **121**, 090502 (2018).
- [43] S. Schlör, J. Lisenfeld, C. Müller, A. Bilmes, A. Schneider, D. P. Pappas, A. V. Ustinov, and M. Weides, Correlating Decoherence in Transmon Qubits: Low Frequency Noise by Single Fluctuators, *Phys. Rev. Lett.* **123**, 190502 (2019).
- [44] P. Murali, J. M. Baker, A. J. Abhari, F. T. Chong, and M. Martonosi, Noise-adaptive compiler mappings for noisy intermediate-scale quantum computers, [arXiv:1901.11054 \[quant-ph\]](https://arxiv.org/abs/1901.11054) (2019)..
- [45] S. Mavadia, C. L. Edmunds, C. Hempel, H. Ball, F. Roy, T. M. Stace, and M. J. Biercuk, Experimental quantum verification in the presence of temporally correlated noise, *Npj Quantum Inf.* **4**, 7 (2018).
- [46] T. Proctor, K. Rudinger, K. Young, E. Nielsen, and R. Blume-Kohout, Measuring the capabilities of quantum computers, [arXiv:2008.11294 \[quant-ph\]](https://arxiv.org/abs/2008.11294) (2020).
- [47] Qiskit terra api reference, *Qiskit Reference Documentation* (accessed: 30-Sep-2020).
- [48] Y. Ding, P. Gokhale, S. F. Lin, R. Rines, T. Proppson, and F. T. Chong, Systematic crosstalk mitigation for superconducting qubits via frequency-aware compilation, [arXiv:2008.09503 \[quant-ph\]](https://arxiv.org/abs/2008.09503) (2020).
- [49] S. Nishio, Y. Pan, T. Satoh, H. Amano, and R. V. Meter, Extracting success from ibm's 20-qubit machines using error-aware compilation, *ACM. J. Emerg. Technol. Comput. Syst.* **16**, 1 (2020).
- [50] S. S. Tannu and M. K. Qureshi, in *Proceedings of the Twenty-Fourth International Conference on Architectural Support for Programming Languages and Operating Systems*, ASPLOS '19 (Association for Computing Machinery, New York, NY, USA, 2019), p. 987.
- [51] 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).
- [52] 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).