Resonator reset in circuit QED by optimal control for large open quantum systems

Samuel Boutin,[1] Christian Kraglund Andersen,[1,2,3,*] Jayameenakshi Venkatraman,[4] Andrew J. Ferris,[5] and Alexandre Blais[1,6]

[1]*Institut Quantique and Département de Physique, Université de Sherbrooke, Sherbrooke, Québec, Canada J1K 2R1*
[2]*Department of Physics and Astronomy, Aarhus University, DK-8000 Aarhus C, Denmark*
[3]*Department of Physics, ETH Zurich, CH-8093 Zurich, Switzerland*
[4]*Department of Physics, Indian Institute of Technology Kanpur, Kanpur 208016, India*
[5]*ICFO-Institut de Ciencies Fotoniques, The Barcelona Institute of Science and Technology, 08860 Castelldefels (Barcelona), Spain*
[6]*Canadian Institute for Advanced Research, Toronto, Canada*

We study an implementation of the open GRAPE (gradient ascent pulse engineering) algorithm well suited for large open quantum systems. While typical implementations of optimal control algorithms for open quantum systems rely on explicit matrix exponential calculations, our implementation avoids these operations, leading to a polynomial speedup of the open GRAPE algorithm in cases of interest. This speedup, as well as the reduced memory requirements of our implementation, are illustrated by comparison to a standard implementation of open GRAPE. As a practical example, we apply this open-system optimization method to active reset of a readout resonator in circuit QED. In this problem, the shape of a microwave pulse is optimized such as to empty the cavity from measurement photons as fast as possible. Using our open GRAPE implementation, we obtain pulse shapes, leading to a reset time over 4 times faster than passive reset.

I. INTRODUCTION

Optimal control, which aims at devising ideal control pulses to optimize a given physical process, is finding wide applications in the fields of theoretical quantum information science [1–9], quantum optics [10], and quantum chemistry [11], among other quantum fields [12]. Quantum optimal control theory has also found applications in the laboratory, in particular with nuclear magnetic resonance [13], trapped ions [14], and superconducting qubits [15,16]. In most instances, optimal control is applied to unitary processes where dissipation is a nuisance and is considered to be detrimental to the desired process. If properly engineered, dissipation can, however, be a useful resource for tasks ranging from quantum state preparation in circuit QED [17,18] to universal quantum computation [19]. While not as widespread as its dissipationless version, open quantum optimal control has also been studied [20–25], with the most widely used algorithms being the open system versions of the GRAPE (gradient ascent pulse engineering) [22,26] and Krotov [27,28] algorithms, while other optimization algorithms [29–31] may also prove useful in the context of open systems.

An important difficulty when dealing with open quantum systems is that the Schrödinger equation is replaced by a master equation and the wave function by a density matrix [32]. For a system of dimension d, described by the master equation $\dot{\rho} = \hat{\mathcal{L}}\rho$, a standard approach for optimal control is then to express the density matrix ρ as a vector ρ_L of dimension $d^2 \times 1$ and the superoperator $\hat{\mathcal{L}}\cdot$ representing the master equation as a matrix L of size $d^2 \times d^2$ [22]. In this representation, the time evolution can be obtained by direct matrix exponentiation which, given the large size of L even for moderate d, rapidly becomes numerically intensive. While alternative implementations with optimized time propagators, for example, using expansion in

Newton polynomials [33,34] or by projection onto Krylov subspace [35,36] can be used, they lack the simplicity of the direct matrix exponentials and are thus not as widespread. Optimal control in open quantum systems has therefore been mostly limited to systems with small Hilbert space size. Here, we present an alternative implementation of the open GRAPE algorithm that eliminates the need to generate the large matrix L. This implementation is well suited for large open quantum systems and avoids explicit matrix exponentiation by rather relying on simple and standard Runge-Kutta time integration of the master equation.

As an example, we apply this open GRAPE implementation to a problem of current experimental interest: resonator reset in circuit QED. In this architecture, qubit readout is realized by injecting microwave photons in a resonator, which is dipole coupled to qubits, and by measuring the photons reflected or transmitted by the resonator. With excess photons in readout resonators having been shown to be a source of unwanted coherent [37–39] and incoherent [40–42] qubit transitions, it is essential to reset the system by removing the measurement photons from the resonator after readout, and before further coherent manipulations or subsequent readout of the qubit can be performed. The usual approach is to wait for several photon decay times $T_\kappa = 1/\kappa$, with κ the resonator decay rate, for the photons to leak out of the resonator [43,44]. In practice, this is, however, often too slow, as a fast repetition time of qubit measurements is critical, e.g., for quantum error correction [45]. With this standard passive approach, this need for fast decay is in contradiction with the necessity to use high-Q resonators to avoid qubit Purcell decay [46]. Alternatively, active reset can be performed, where a microwave tone is used to empty the resonator in a shorter time. Such a reset tone can be either conditional on the readout result [44] or unconditional [43,44], using no knowledge of the resonator and qubit states. Devising an active unconditional reset protocol is an ideal test problem for our open GRAPE implementation, since it is an intrinsically dissipative process requiring a large Hilbert

*chanders@phys.ethz.ch

space size due to the many resonator photons used for qubit measurement. Moreover, active resonator reset in circuit QED was recently explored experimentally [43,44], giving us the opportunity to consider parameters of current practical interest. In addition to this example, the numerical approach presented here has also recently been successfully applied by some of us to optimize a fast initialization of cat states in a Kerr resonator based on two-photon driving [47].

The paper is organized as follows: We first present a brief overview of open GRAPE in Sec. II. We then discuss our implementation of this algorithm in Sec. III. Section IV is devoted to the application of the algorithm to active resonator reset. Finally, Sec. V summarizes our work.

II. OPTIMAL CONTROL FOR OPEN QUANTUM SYSTEMS

Before discussing our implementation of the algorithm, we first present an overview of the problem solved by the GRAPE algorithm [26] and of open GRAPE [22]. The reader familiar with these concepts can immediately skip to Sec. III.

A. The control problem

Consider a system with the free Hamiltonian H_0 and subject to R independent control fields, each described by the Hamiltonians H_k such that the full system Hamiltonian reads [9,26]

$$H(t) = H_0 + \sum_{k=1}^{R} u_k(t) H_k. \tag{1}$$

The classical parameters $u_k(t)$ in the above expression can be continuously adjusted to change the strength of the control fields on the system. In the context of circuit QED, these $u_k(t)$ can, for example, correspond to the time-dependent amplitude of different microwave drives on the resonator or the qubit.

The objective of the control problem is to find the optimal set $\{u_k(t)\}$ to accomplish a specific task, most typically implementing quantum gates [5,6]. This can be expressed as an optimization problem where the goal is to maximize the performance index $\Phi[\{u_k\}]$, a measure for the success of the desired task and a functional of the control parameters. As the optimization problem must be of finite dimension, the control amplitudes, $u_k(t)$, are taken to be piecewise constant. For a process of duration T, each $u_k(t)$ is divided into N time steps of duration $\Delta t = T/N$ as illustrated in Fig. 1(a). In this way, for the jth step, i.e., for $t \in [(j-1)\Delta t; j\Delta t]$, the function $u_k(t)$ is a constant of amplitude $u_k(j)$ with $j \in \{1,2,\ldots,N\}$. The elements of the set $\{u_k(j)\}$ are referred to as the controls.

In practice, these sharp controls are smoothed out by the finite bandwidth of the control lines. Following Ref. [9] and as illustrated in Fig. 1(b), this important experimental consideration can be taken into account by filtering the controls in the evaluation of the performance index and its gradient. This filtering procedure maps the piecewise constant functions described by the set $\{u_k(j)\}$ to smoother piecewise constant functions defined by the larger set $\{s_k(l)\}$ with $l = 1,2,\ldots M$ and $M = T/\delta t \gg N$. For completeness, details of this filtering procedure can be found in Appendix A.

An approach to optimize the performance index is to update the controls by using a gradient-based optimization algorithm
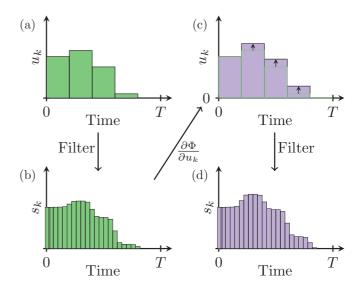

FIG. 1. Schematic of a gradient-based optimization step with GRAPE-type controls update and Gaussian filtering to account for experimental constraints. Starting from the initial controls shown in (a), we calculate the filtered experimental pulse shape in (b). From this filtered field, the gradient $\partial\Phi/\partial u_k$ is calculated using the chain rule (see Ref. [9] or Appendix A for details) and the controls are updated in (c), which leads to a new filtered field in (d). The boundary conditions of the field are taken into account by fixing the first and last control.

such that [26]

$$u_k(j) \rightarrow u_k(j) + \sum_{lm} B_{kj,lm} \frac{\partial \Phi}{\partial u_l(m)}, \tag{2}$$

where $B_{kj,lm}$ are the elements of a step matrix which depends on the details of the chosen optimization algorithm. Simple gradient descent optimization corresponds to the choice $B_{kj,lm} \propto \delta_{kl}\delta_{jm}$, while for more sophisticated methods, such as the Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm, $B_{kj,lm}$ is related to the inverse of the Hessian matrix [48]. Since the BFGS algorithm leads to improved convergence [49], it will be used in the numerical computations presented below. A nontrivial step in the update rule Eq. (2) is the evaluation of the gradient of the performance index. While this can be done by numerical derivatives, this approach becomes intractable for problems with a large set of controls. Using an analytical result described below for open systems, the GRAPE algorithm allows for an efficient calculation of this gradient.

B. Open GRAPE

We consider an open quantum system whose dynamics is described by the Markovian master equation

$$\dot{\rho} = -i[H,\rho] + \hat{\Gamma}\rho \equiv \hat{\mathcal{L}}\rho. \tag{3}$$

In this expression, $\hat{\Gamma}\cdot$ is the superoperator for the different possible dissipation channels acting on the system and which can be expressed in standard Lindblad form as [32]

$$\hat{\Gamma}\rho = \sum_j \gamma_j \hat{\mathcal{D}}[a_j]\rho, \tag{4}$$

with $\hat{\mathcal{D}}[a_j]\rho = a_j\rho a_j^\dagger - \{a_j^\dagger a_j,\rho\}/2$ and γ_j the damping rate for channel j associated to the system operator a_j.

The formal solution to this equation can be expressed as the time-ordered exponential

$$\rho(t) = \mathcal{T} \exp \left\{ \int_0^t dt' \, \hat{\mathcal{L}}(t') \right\} \rho(0). \tag{5}$$

Taking advantage of the piecewise constant nature of the controls, this can be written more simply as

$$\rho(T) = \hat{L}_N \ldots \hat{L}_j \ldots \hat{L}_1 \rho(0), \tag{6}$$

with the evolution superoperator defined from time $(j-1)\Delta t$ to time $j \Delta t$ as

$$\hat{L}_j \cdot = \exp\{-i\Delta t \, ([H_j, (\cdot)] + i\hat{\Gamma}\cdot)\}, \tag{7}$$

where $H_j = H_0 + \sum_k u_k(j) H_k$ is the time-independent Hamiltonian associated to the jth time step.

For many control problems, the performance index can be expressed as a function of operator averages or, alternatively, as the overlap between a final state $\rho(T)$ and a target state. In both cases, the resulting performance index takes the form

$$\Phi = \mathrm{Tr}(\sigma \hat{L}_N \ldots \hat{L}_1 \rho(0)), \tag{8}$$

where σ is either the target state or an operator whose expectation value is evaluated. In the former case, if the target state is pure this figure of merit is bounded between 0 and 1, with $\Phi = 1$ for $\rho(T) = \sigma$.

Taking advantage of the piecewise constant character of the evolution, the derivative of the performance index takes the form [26]

$$\frac{\partial \Phi}{\partial u_k(j)} = \mathrm{Tr} \left\{ \lambda_j(\sigma) \frac{\partial \hat{L}_j}{\partial u_k(j)} \rho_{j-1} \right\}, \tag{9}$$

where

$$\rho_j = \hat{L}_j \ldots \hat{L}_1 \rho(0) \tag{10}$$

is a forward-in-time evolved density matrix, while

$$\lambda_j(\sigma) = \hat{L}_{j+1}^\dagger \ldots \hat{L}_N^\dagger \sigma \tag{11}$$

is the backward-in-time evolution from the final target state. To first order in Δt the derivative of the jth time-evolution operator is [22]

$$\frac{\partial \hat{L}_j}{\partial u_k(j)} \approx -i\Delta t \, [H_k, (\hat{L}_j \cdot)]. \tag{12}$$

Approximation of the gradient to higher order in Δt can improve convergence of the optimization [49]. Moreover, for simplicity, we have considered the controls to be parameters of the Hamiltonian only. This approach can, however, be adapted to allow for control over the dissipation rates γ_j [7].

Finally, the derivative of the performance index is

$$\frac{\partial \Phi}{\partial u_k(j)} = -i\Delta t \, \mathrm{Tr}\{\lambda_j(\sigma)[H_k, \rho_j]\}. \tag{13}$$

Thus, evaluating the gradient of the performance index requires the calculation of the forward-in-time evolved states ρ_j and of the backward-in-time evolved targets $\lambda_j(\sigma)$. The analytical result of Eq. (13) is the core of the GRAPE algorithm [26]. The standard approach to obtain these states, ρ_j and $\lambda_j(\sigma)$, is to express the density matrices and the master equation in Liouville space [22]. For a system with Hilbert

space dimension d, the superoperators then take the form of $d^2 \times d^2$ matrices and the N evolution operators $\hat{L}_j$ are obtained by computing matrix exponentials of these matrices. While simple to implement, this procedure is numerically intensive for moderate to large system sizes.

III. OPEN GRAPE WITH RUNGE-KUTTA INTEGRATION

Rather than relying on direct matrix exponentiation, we present here an approach based on numerical integration of the master equation using a standard Runge-Kutta routine. This approach is not a unique method to avoid the matrix exponentiation for optimal control [33,35,36], but below we argue that, even for moderate Hilbert space dimension d, this simple Runge-Kutta routine leads to useful computational speedups compared to performing matrix exponentials as used in standard implementations of open GRAPE [22].

With this method, the forward-in-time propagation is performed by numerical integration of the differential equation

$$d\rho = \hat{\mathcal{L}}\rho \, dt, \tag{14}$$

starting from the initial state $\rho(0)$ using standard Runge-Kutta routines. In practice, the integration is done in a stepwise manner to obtain ρ_j for all values of j. In other words, Eq. (14) is integrated for a time Δt from the initial state ρ_0 to obtain ρ_1, which is saved for later use. Then ρ_1 is used as the initial state and integrated for a time Δt to obtain ρ_2, and so on. Similarly, the backward-in-time propagation is performed by numerical integration of the master equation

$$-d\lambda = \hat{\mathcal{L}}^\dagger \lambda \, (-dt), \tag{15}$$

which is also solved stepwise but backward in time, such that $\lambda(t - \delta t) = \lambda(t) + \hat{\mathcal{L}}^\dagger \lambda(t)(-\delta t)$, with δt as a small numerical step, from the initial (target) state $\lambda_N = \lambda(T) = \sigma$. Backward-in-time integration for a time Δt leads to λ_{N-1}, which is then used as the next initial state and, continuing this way, all λ_j are obtained. With ρ_j and λ_j calculated, the derivative given in Eq. (13) is readily evaluated using the saved ρ_j and λ_j.

A. Complexity analysis

We now turn to an evaluation of the scaling with system size d of the standard approach versus the present Runge-Kutta integration method. For simplicity, we neglect the efficiency gain that can be obtained in both cases from taking advantage of the sparse character of matrices. We also take the complexity of the multiplication and exponentiation of $n \times n$ matrices to be $O(n^3)$. Better scaling can be obtained from state-of-the-art algorithms, resulting in improvements for both the standard approach and the present Runge-Kutta integration method.

In the standard approach were the density operator is represented as a vector, the matrix exponentiation involved in computing the superoperators $\hat{L}_j$ of dimensions $d^2 \times d^2$ has a complexity $O(d^6)$. For the N piecewise constant steps of the controls, the total complexity is therefore

$$\mathcal{C}_{\exp} = O(N \times d^6). \tag{16}$$

In contrast, the Runge-Kutta integration approach described here requires the products of operators represented by $d \times d$ matrices. One caveat of this method is that the calculation is

specific to the given input state $\rho(0)$. The complexity of this approach can then be estimated as

$$\mathcal{C}_{\mathrm{RK}} = O(n_s\, n_{\mathrm{RK}} \times d^3), \tag{17}$$

where n_s is the number of input states to be considered and n_{RK} the number of Runge-Kutta steps during the full time evolution. Improvement over the standard Liouville space approach is thus expected for system size $d \gg (n_s n_{\mathrm{RK}}/N)^{1/3}$. Importantly, the numbers n_s, n_{RK}, and N are often independent of system size, suggesting a computational speedup for large Hilbert spaces. When considering bandwidth filtered controls, where the N controls are replaced by $M \gg N$ subpixels in order to approximate a smooth function [9] (see Appendix A), computational speedup is expected for even smaller Hilbert space sizes. Note that Eq. (17) assumes that the Runge-Kutta integration is performed using matrix multiplications with complexity $O(d^3)$. As mentioned previously, there is an alternate representation of the master equation Eq. (3) where ρ is a vector of dimension $d^2 \times 1$ and the Lindbladian is a $d^2 \times d^2$ matrix. In that case, the Runge-Kutta integration requires matrix-vector multiplication of complexity $O(d^4)$ reducing the speedup.

A second advantage of the present approach, not captured by this simple analysis, is the reduced memory usage, since superoperators in Liouville space of matrix size $d^2 \times d^2$ are never created nor stored in memory. The reduced memory requirements by the Runge-Kutta approach is independent of the representation of the density matrix used for the integration.

The optimization of an arbitrary process requires averaging the performance index over $n_s = d^2$ input states spanning the full Liouville space [24]. However, this is not the case for many processes, where we can expect $n_s \ll d^2$. In particular, average over only three appropriately chosen input states is required to optimize a unitary process in the presence of dissipation [24]. Another related issue is that for a general process there may be up to d^2 dissipators in Eq. (4), which would result in the scaling of $O(n_s\, n_{\mathrm{RK}} \times d^5)$. However, for many problems of practical interest, such as the one presented in the next section, only a few dissipators are needed.

Estimating n_{RK} is a more difficult task since, with adaptive integration step size, the number of integration steps is parameter and problem dependent [50]. As an example, for the reset process described in Sec. IV A, we observe that $n_{\mathrm{RK}}/M \approx 10$–$100$, depending on the chosen value of M. Given that $n_s = 2$ for the reset problem, we therefore expect significant speedup even for a moderate Hilbert space size of $d \approx 10$.

Finally, we note that the Runge-Kutta approach presented here is only efficient if we perform a GRAPE-type concurrent update of the controls. In the case of a Krotov-type update where only one control is updated at each step of the optimization algorithm [21], the complexity of the present method and of the approach where the density matrix is represented as a vector are expected to be similar. Indeed, the latter approach allows one to reuse most of the calculated exponentials between updates. Consequently, here we consider a GRAPE-type update where all controls are updated concurrently.
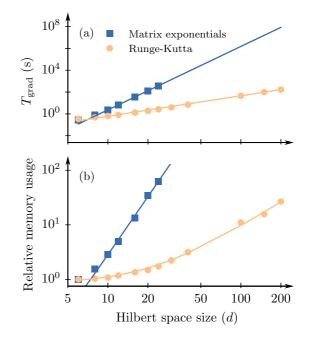

FIG. 2. (a) Average runtime per evaluation of the performance index and its gradient (T_{grad}). Points are numerical data obtained from the averaging over 100 optimization iterations. In the matrix exponentiation case, a first-order GRAPE implementation based on the control module of QuTiP is used [51]. (b) Average memory usage (RAM) during a GRAPE iteration normalized by the $d = 6$ value. Points are numerical data obtained using the "memory profiler" package (version 0.47). For both panels, solid lines are power-law fits. A constant background due to program overhead is included in the fits of the second panel (see text). The matrix-exponential calculations were done for a Hilbert space size of only $d \leqslant 24$ due to limitations of RAM. From an initial state $|\alpha\rangle \otimes |e\rangle$, with the cavity in a coherent state with $\alpha = \sqrt{d/8}$ and the qubit in the excited state, we use the two quadratures of the cavity drive to perform an active reset (target state $|0\rangle \otimes |e\rangle$) in a time $T = \pi/g$ separated in $N = 200$ steps. The qubit and cavity are coupled through a standard Jaynes-Cummings interaction with coupling strength $g = 100\kappa$. Additional parameters are $\omega_d = \omega_r$ and $\Delta = 10\kappa$ (see Sec. IV A for definitions).

B. Performance of implementation

As a verification of the above algorithmic complexity arguments, we consider the performance of our Runge-Kutta–based open GRAPE implementation for a simplified version of the resonator reset control problem considered in the following section. More precisely, we consider a cavity-qubit system coupled through a standard Jaynes-Cummings interaction. Starting from an initial state where the cavity is in a coherent state and the qubit in the excited state, we optimize a drive in order to empty the cavity in a time T, without perturbing the qubit state, such that the target final state is $|0\rangle \otimes |e\rangle$. This problem corresponds to a conditional reset where the qubit state is fixed, leading to a single initial and final state, i.e., $n_s = 1$ in Eq. (17). In the following section, the case of unconditional reset, where the reset protocol is independent of the qubit state, will be considered.

Figure 2(a) presents T_{grad}, the average runtime for the evaluation of Φ and its gradient as a function of Hilbert space size d for both our Runge-Kutta–based implementation of open

GRAPE (orange circles) and a standard matrix exponentiation-based implementation (blue squares) [51]. In both cases, the gradient is calculated using a first-order approximation as in Eq. (12). As the approaches considered are implemented using different programming languages leading to different runtime overheads, the relevant quantity in this figure is the scaling of runtime with respect to the Hilbert space size d rather than the absolute times. The solid lines are power-law fits to the numerical data, $T_{\text{grad}} \propto d^{\xi}$ with exponent ξ. Exponents obtained are in close agreement to the previous analysis, with $\xi = 5.8$ for the matrix exponentiation case and $\xi = 1.9$ for the Runge-Kutta–based approach. Note that this significant polynomial speedup is better than expected from the analysis in Sec. III A due to the use of sparse matrix properties in our implementation.

Figure 2(b) presents the average memory (RAM) used during a GRAPE iteration relative to the memory usage of the $d = 6$ case for each implementation. Performing again power-law fits but allowing for a constant background to take into account possible memory overheads, one finds the exponents $\xi = 3.5$ for matrix exponentiation and $\xi = 1.5$ for our Runge-Kutta implementation. Hence, as expected from the above complexity analysis, the memory requirement of the matrix exponentiation is much greater than the Runge-Kutta approach, as it requires the storage of propagators as $d^2 \times d^2$ matrices, limiting considerably the Hilbert space sizes on a standard computer.

IV. APPLICATION TO RESONATOR RESET

As an application of this open GRAPE implementation, we consider the problem of active reset following qubit readout in circuit QED [52,53]. Before presenting numerical results, we first briefly review qubit readout in this system and present the active reset problem.

A. Readout and reset in circuit QED

Circuit QED is characterized by the strong electric-dipole coupling g between a superconducting qubit of frequency ω_a and a microwave resonator of frequency ω_r. In the dispersive regime, where the qubit-resonator detuning $|\Delta| = |\omega_a - \omega_r| \gg g$, the system is described by the effective Hamiltonian ($\hbar = 1$) [53]

$$H_0 = (\omega_r + \chi \sigma_z) a^{\dagger} a + \frac{\omega_a}{2} \sigma_z + \varepsilon(t)[a^{\dagger} e^{-i\omega_d t} + \text{H.c.}], \quad (18)$$

where $\chi = g^2/\Delta$ is the dispersive shift and H.c. stands for hermitian conjugate. The last term represents a drive on the cavity of amplitude $\varepsilon(t)$ and frequency ω_d. Because of the dispersive coupling, the cavity frequency is shifted by $\pm\chi$ depending on the state of the qubit. Under drive, the time evolution leads to a qubit-state-dependent population and/or phase of the cavity state. This dependency can be resolved by homodyne detection of the photons leaking out of the cavity at a rate κ, leading to a qubit measurement.

In order to include cavity damping in our calculations, we use the master equation

$$\dot{\rho} = -i[H, \rho] + \kappa \hat{\mathcal{D}}[a]\rho, \quad (19)$$

where κ is the cavity decay rate associated to the dissipator $\hat{\mathcal{D}}[a]\rho = a\rho a^{\dagger} - \{a^{\dagger} a, \rho\}/2$. Under a constant drive of amplitude ε, the steady-state solution in the dispersive regime (i.e., $H = H_0$) of this master equation leads to the qubit-state-dependent intracavity average photon number

$$\bar{n}_{g/e} = \frac{\varepsilon^2}{(\omega_r \pm \chi - \omega_d)^2 + (\kappa/2)^2}. \quad (20)$$

Here, we are concerned with the return to vacuum state once the measurement is completed. The common approach of passive reset is to wait for a time $T \gg 1/\kappa$ for the photons to naturally escape from the resonator. We use our implementation of open GRAPE to find an optimal $\varepsilon(t)$ to speed-up this process to times smaller than $1/\kappa$ through an active process.

When driving at a frequency $\omega_r \approx \omega_d$, the average number of photons is independent of the qubit state and an active reset is easily obtained by changing the phase of the drive. However, active reset is not as simple when considering the nonlinear corrections to the dispersive Hamiltonian. The first of these corrections is a qubit-induced nonlinearity of the cavity described by the Hamiltonian[1] [54–56]

$$H_K = K(a^{\dagger} a)^2, \quad (21)$$

with K the Kerr nonlinearity, which is negative in superconducting quantum circuits. This correction makes exact analytical solutions of the active reset problem difficult as it leads to nonlinear equations of motions for the resonator state. This nonlinearity can, moreover, lead to vastly different qubit-state-dependent resonator states, something that has been exploited for qubit readout, e.g., in the Josephson bifurcation amplifier [57]. Here, because of this nonlinearity, a reset pulse more complicated than in the purely dispersive case is found to be necessary [43].

In the next section, we present numerical results for active cavity reset based on the experimental parameters reported in Ref. [43]. For these calculations, we use the master equation of Eq. (19) with Hamiltonian $H = H_0 + H_K$ and the experimentally relevant parameters $\chi = 2\pi \times 1.3$ MHz, $K = -2\pi \times 2.1$ kHz, and $\kappa = 2\pi \times 1.1$ MHz, corresponding to a photon decay time of $T_{\kappa} = 1/\kappa = 145$ ns. Moreover, to help in making comparisons, we will express the drive strength in similar terms as in Ref. [43] by introducing the normalized drive power $P_{\text{norm}} = P/P_{\text{1ph}}$, where P is the applied drive power and P_{1ph} is the drive power leading to an average steady-state resonator population of one photon. With the above parameters, we numerically identify the corresponding driving amplitude $\sqrt{P_{\text{1ph}}} = 2\pi \times 1.595$ MHz such that $\varepsilon = \sqrt{P_{\text{norm}} P_{\text{1ph}}}$.

---

[1] In the two-level approximation of circuit QED, the sign of this nonlinear correction is qubit-state dependent, with $H_K \propto \sigma_z$. However, in the more complete multilevel treatment, the Kerr nonlinearities K_g (K_e) of the resonator for a qubit in the ground (excited) state can have the same sign [54]. In particular, for the parameters considered here the Kerr nonlinearities have the same sign and are of similar amplitudes for both qubit states [43]. For simplicity, we consider $K_e \approx K_g$.

B. Active reset using open GRAPE

We now turn to a numerical study of active resonator reset using the open GRAPE implementation introduced in Sec. III. For simplicity, we assume the measurement preceding the resonator reset to be quantum nondemolition and, thus, consider the qubit's state to be fixed throughout the process. As a result, we can replace the operator σ_z by the number ± 1 in Eq. (18).

As we seek an active reset protocol independent of measurement outcomes, the performance index used for the open GRAPE optimization is averaged over the two qubit states. Following Eq. (8), the simplest performance index is

$$\Phi = \sum_{i=g,e} \text{Tr}\{\rho_T \rho_i(T)\}, \qquad (22)$$

with $\rho_{i=g,e}(t)$ the qubit-dependent resonator state and $\rho_T = |0\rangle\langle 0|$ the target (vacuum) state. Here, $\rho_{i=g,e}(t=0)$ are the qubit-dependent resonator states following a measurement pulse $\varepsilon(t)$ of duration T_m similar to that used in Ref. [43]. In our simulations, the resonator is initialized to the vacuum state at time $t = -T_m$, the state is then time evolved using the master equation, Eq. (19) with $H = H_0 + H_K$, leading to the qubit-dependent states, $\rho_{i=g,e}(t=0)$. Starting from these states and using the same master equation, the open GRAPE algorithm is then used to optimize the unconditional reset pulse shape $\varepsilon(t)$ for $t \in (0,T)$, with $\varepsilon(t=0)$ fixed by the measurement pulse and $\varepsilon(t=T) = 0$.

Using the parameters of the previous section, Fig. 3(a) compares the average intracavity photon number as a function of time under various resonator reset schemes. In particular, the passive reset (thin light orange curves) is compared to the GRAPE optimized active reset (thick dark purple curves) for duration $T = 300$ ns $\approx 2T_\kappa$. While there is still significant resonator population after a wait time $T \gtrsim 2T_\kappa$ in the passive case, the GRAPE optimized pulse empties the cavity independently of the qubit state. More precisely, the log-scale inset shows that the optimized pulse shape brings the photon number below 10^{-4} while at the same time passive reset leads to a residual average photon population close to 1. The numerically found pulse shape corresponding to these results is the thick dark purple line in Fig. 3(b). It shows a fast oscillating behavior on top of a slowly evolving envelope. Importantly, the quality of the reset is only marginally affected by these rapid oscillations. Indeed, as shown by the thick light purple lines in both panels, a polynomial fit to the optimized pulse shape essentially leads to changes in the average photon number that are only visible on the log-scale inset of Fig. 3(a). This indicates that a complex pulse shape is not essential to obtain good performance, and that the solution may be amenable to regularization, whereby penalties are added to the objective function (for instance, to penalize rapid changes in time) in order to make the result simpler and/or more robust. As pointed out in Ref. [58], the effect of noise in the control pulse, similar to fast oscillations observed here, is negligible if the noise level is below the error in reaching the target state.

As a comparison, the thin dark orange lines in Fig. 3 correspond to the average photon number and pulse shape used in an optimized two-step active reset similar to the so-called CLEAR (cavity-level excitation-and-reset) pulse introduced in Ref. [43]. Compared to CLEAR, the GRAPE pulse shape
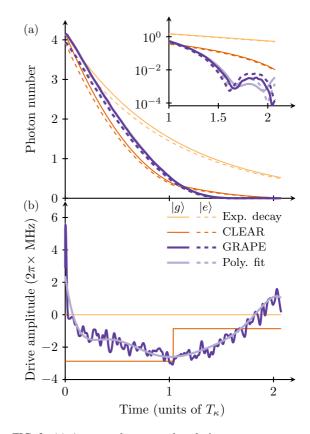

FIG. 3. (a) Average photon number during resonator reset procedures following a readout process with drive power $P_{\text{norm}} = 4$. The solid (dashed) lines indicate results for the qubit in the ground (excited) state. See panel (b) for legend. The inset shows the average photon number for $T_\kappa > 1$ on a logarithmic scale to allow for better comparison of the reset schemes. (b) Pulse shapes for the resonator reset procedures used in panel (a). System parameters and the Hamiltonian are described in Sec. IV A. Additional parameters for the GRAPE algorithm include a control duration $\Delta t = 1$ ns and Gaussian filtering with bandwidth $\omega_B/2\pi = 100$ MHz and subpixel duration $\delta t = 0.1$ ns. (See Appendix A for parameter definitions.)

leads to a smaller residual photon population of the cavity in $T = 300$ ns $\approx 2T_\kappa$. Importantly, because photon decay under GRAPE optimized pulse shapes is far from exponential, in the example of Fig. 3 the cavity is already close to having reached its final state at a time ~220 ns. This suggests that faster resets are possible.

To further speed-up the process, we follow the insight from DRAG (derivative removal by adiabatic gate) and optimize over two quadratures of the drive [5]. In a frame rotating at the drive frequency, the last term of the dispersive Hamiltonian of Eq. (18) is then replaced by

$$H_d = \varepsilon_X(t)(a^\dagger + a) + i\varepsilon_Y(t)(a^\dagger - a). \qquad (23)$$

Results from optimization of these two quadratures are presented in Fig. 4, which shows the average photon number at the final pulse time T for increasing measurement power P_{norm}. As an initial guess, the X quadrature is set to the CLEAR pulse shape and the Y quadrature is randomly set. These results are shown for four different values of T, ranging from 150 ns $\approx 1.04T_\kappa$ (green circles) to times as short as 40 ns $\approx 0.3T_\kappa$ (red ×). Following the convention of Fig. 3(a),
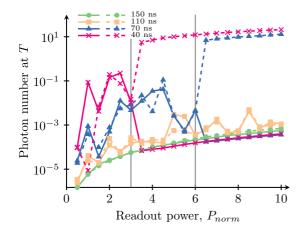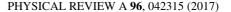
FIG. 4. Average photon number at the end of the active reset pulse as a function of the readout power P_{norm}. The results are shown for pulses of duration $T = 150$ ns, 110 ns, 70 ns, and 40 ns, corresponding to $T \approx 1.04T_\kappa, 0.76T_\kappa, 0.48T_\kappa$, and $0.28T_\kappa$. The solid (dashed) line is the average final photon number for the qubit in the ground (excited) state. The vertical gray lines indicate the failure points for the 70-ns and the 40-ns optimizations.

the full lines correspond to the qubit ground state and the dashed lines the qubit excited state. Unsurprisingly, the general trend is an increase of the residual photon number with P_{norm}. However, for 150 ns $\approx 1.04T_\kappa$, the optimization results in residual population as small as 10^{-3} at high power $P_{norm} = 10$.

The difficulty of the open GRAPE algorithm to converge with decreasing T is made apparent with the large fluctuations of the residual photon number with P_{norm}. Despite this, and quite remarkably, final populations of less than 10^{-3} photons are obtained for reset times under T_κ and all P_{norm} values considered. The complexity in converging becomes more apparent at very short times where we observe large fluctuations and large separations between the results obtained for the two qubit states. These branchings, corresponding to a change in the optimization landscape as a function of T and P_{norm} [59], are illustrated by vertical gray lines for the two shortest values of T. Beyond the branching time, the optimization finds a good solution only for the qubit in the ground state. This is a result of the sign of the Kerr nonlinearity K. Indeed, the effective detuning from the drive at high drive power is smaller when the qubit is in the ground state; thus, changing the sign of K leads to finding low-photon-number solutions when the qubit is rather in the excited state.

Figure 5 presents this branching time as a function of P_{norm}. As illustrated by the dashed line, this failure time follows a simple power-law behavior. This is reminiscent of a quantum speed limit, which here corresponds to the minimal time T in which the optimization can be successful [60,61]. For pure state evolution, the quantum speed limit can be expressed analytically in terms of the mean value and the variance of the energy [62,63]. Expressions have also been obtained for open processes [64,65]. The observed simple behavior with P_{norm} suggests that analytical expressions could also be obtained for the reset problem. We note, however, that variations in the initial guess for the controls, cost function, or optimization algorithm could lead to faster reset times [59], and that the
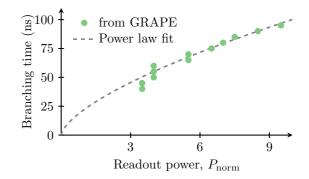

FIG. 5. The green dots are the numerical speed limit extracted from the open GRAPE optimizations. We define here the speed limit as the time where the optimization fails, corresponding to the branching points indicated by gray lines in Fig. 4 for the 40-ns and 70-ns curves. The dashed gray line is a power-law fit $\propto (P_{norm})^\xi$ to the data with $\xi = 0.65$.

results of Fig. 5 therefore do not represent an absolute speed limit.

To gain more insights on the optimization, Fig. 6 presents the average photon number as a function of time and the corresponding pulse shapes obtained from GRAPE (inset). These results are shown for $T = 70$ ns (light orange lines) and $T = 110$ ns (dark blue lines) with a readout power of $P_{norm} = 6$. Both pulse shapes are similar and are reminiscent of a smoothed CLEAR pulse [43]. The Y quadrature also appears to have minimal impact and is always close to zero. For both of the final times T, the average photon number first increases from its initial value of ~ 6 before decreasing to the value shown in Fig. 4. This increase is particularly notable for the short pulse time $T = 70$ ns and points to the difficulty in converging as the reset time T is decreased. In practice, this large photon population can lead to a breakdown of the dispersive approximation used here and to a departure from the quantum nondemolition character of the dispersive readout [66]. With the parameters used here, this breakdown
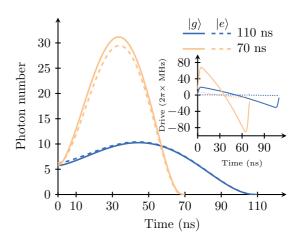

FIG. 6. Photon number as a function of time during the active reset pulse for a pulse duration of 110 ns and 70 ns. The solid (dashed) line is for the qubit in the ground (excited) state. The inset shows the corresponding Gaussian filtered drives. The solid lines of the inset are the X drive while the dotted lines are the Y drive.
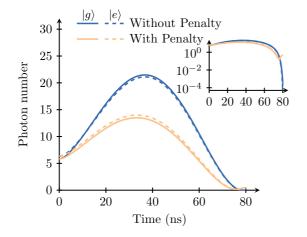
FIG. 7. Photon number as a function of time for optimized drives with (light orange curves) and without (dark blue curves) the photon number penalty Φ_p included. The solid (dashed) curves are for the qubit in the ground (excited) state. The inset displays the same data with a logarithm photon number axis. The parameters are the same as Fig. 3. We use the penalty weight $\beta = 0.2/T$.

is expected to occur for $T = 70$ ns, where the average photon number exceeds the critical photon number $n_{\mathrm{crit}} = (\Delta/2g)^2 \approx 29$ for a short period of time.

To prevent this large photon number increase, a penalty Φ_p related to the intracavity photon number can be added to the performance index such that

$$\Phi = \Phi_0 - \beta\Phi_p, \tag{24}$$

with Φ_0 defined by Eq. (22) and β, a constant weighting the penalty, which is determined by trial and error. To penalize large photon populations we take

$$\Phi_p = \sum_{i=g,e} \int_0^T \mathrm{Tr}\{a^\dagger a\, \rho_i(t)\}dt. \tag{25}$$

Details about the numerical implementation of Φ_p and its derivative with respect to the controls can be found in Appendix B. Results for optimization with this modification of the performance index are presented in Fig. 7 for $T = 80$ ns and $P_{\mathrm{norm}} = 6$. For these values, the optimization without penalty reaches a final photon population of 10^{-4} but reaches close to 25 photons in the transient dynamics. On the other hand, by using Eq. (24) with the initial value of the pulse given by the results obtained without penalty, the transient photon number can be kept well below n_{crit}. This is, however, achieved at the cost of an increase of the final photon number to $\sim 10^{-1}$. These results for the photon penalty may be improved by considering more diverse initial pulse shapes probing a larger region of the optimization space. In addition, a more systematic study of the role and optimal value of the weight β could improve the results.

V. CONCLUSION AND OUTLOOK

We have shown an implementation of the GRAPE algorithm for open quantum systems that circumvents the usual explicit calculation of matrix exponentials. This implementation is advantageous when optimizing quantum processes in large open quantum systems, leading to reduced computation times and memory requirements compared to standard implementations of open GRAPE based on matrix exponentials.

As an example of this approach, we have demonstrated an optimized reset protocol for a readout resonator in circuit QED. As the reset time limits the repetition time of current experiments, rapid qubit reset after readout is of high practical importance. Moreover, rapid qubit recycling can be advantageous in the implementation of quantum algorithms [67]. Furthermore, the results of our optimization may be directly applied to protocols that rely on repetitive qubit readout in circuit QED, e.g., in quantum feedback schemes [68–70] or in quantum error correction protocols [45,71]. The numerical optimization presented in this work presents a reset scheme that significantly reduces reset time compared to passive reset. Moreover, this study pinpoints the issues occurring when extremely short reset times are sought and yields a branching point beyond which the optimization algorithm fails to find a qubit-state-independent solution. We find that this branching follows a power law as a function of the readout power, indicating a relation between the system's energy and the shortest time required to achieve the target states. Finally, we identify that our scheme can be readily extended to include additional constraints such as a penalty on large average photon numbers in order to keep the cavity population below the critical photon number set by the dispersive approximation.

While resonator reset in the dispersive regime of circuit QED serves as an instructive study, we emphasize that this implementation of GRAPE may have much broader use. As a second practical example, our approach has also been recently applied by some of us to fast cat states generation in nonlinear resonators [47]. Following recent experimental results, our work could be expanded to study resonator reset in the strongly nonlinear regime of circuit QED [44]. Our approach appears ideally suited to simulate the large Hilbert space that is needed to simulate these experiments. Another application is the optimization of qubit measurement in circuit QED [7,72]. Finally, our implementation may prove useful in optimizing unitary gates that not only work in the qubit subspace but rely on the full Hilbert space of a resonator and multiple qubits [73].

APPENDIX A: GAUSSIAN FILTER

In this Appendix, we present the Gaussian filtering procedure developed by Motzoi *et. al.* in Ref. [9], and mentioned in Sec. II. In circuit QED, while typical electronics limits the controls $\{u_k(j)\}$ to a minimal duration Δt of a few nanoseconds, the limited bandwidths of control lines and pulse generators leads to a smoothed drive which can significantly modify the dynamics from the one expected from piecewise constant drives. The filtering approach summarized here allows us to incorporate these experimental constraints in the GRAPE algorithm. The main idea of Ref. [9] is to calculate the dynamics using a new smoothed pulse $s_k(t) \equiv s_k[\{u_k(j)\},t]$, which is a functional of the set of controls, while still performing the optimization on the N controls $\{u_k(j)\}$.

As the GRAPE algorithm requires a piecewise constant field, this new smoothed drive $s_k(t)$ is approximated as a piecewise constant drive, with each step a subpixel of amplitude $s_{k,n}$ and duration $\delta t \ll \Delta t$. The set of controls $\{u_k(j)\}$ now translates into a set of drive amplitudes $s_k(n)$ for a time $t \in [(n-1)\delta t; n\delta t]$, with $n \in \{1,2,\ldots M\}$ and $M = T/\delta t \gg N$ the number of subpixels. The controls and the smoothed drive are related by

$$s_k(n) = \sum_{j=1}^{N} T_{k,n,j}\, u_k(j), \tag{A1}$$

with $T_{k,n,j}$ a transfer function matrix which acts as a filter on the controls. The derivatives of the performance index can be calculated using the chain rule,

$$\frac{\partial \Phi}{\partial u_k(j)} = \sum_{n=1}^{M} \frac{\partial \Phi}{\partial s_k(n)} \frac{\partial s_k(n)}{\partial u_k(j)}, \tag{A2}$$

where the derivative with respect to $s_k(n)$ can be found using Eq. (12), while $\partial s_k(n)/\partial u_k(j)$ comes directly from the transfer matrix.

In this paper, all numerics use transfer functions based on Gaussian filters, since most experimental hardware constraints can be approximated well by such a filter [9]. Hardware components are typically characterized by their 3-dB attenuation bandwidth, ω_B. Using a filter function

$$F(\omega) = \exp(-\omega^2/\omega_0^2), \tag{A3}$$

with the reference bandwidth for a given control field given by $\omega_0 = \omega_B/[-\ln(1/\sqrt{2})]^{1/2} \approx \omega_B/0.5887$, the transfer matrix can now be calculated as [9]

$$T_{k,n,j} = \int_{-\infty}^{\infty} \frac{F(\omega)\cos\left(\omega \frac{2(n-1)\delta t - (2j-1)\Delta t}{2}\right)\sin\left(\frac{\omega \Delta t}{2}\right)}{\pi \omega}\, d\omega$$

$$= \frac{\mathrm{erf}\left[\omega_0 \frac{(n-1)\delta t - (j-1)\Delta t}{2}\right] - \mathrm{erf}\left[\omega_0 \frac{(n-1)\delta t - j\Delta t}{2}\right]}{2}, \tag{A4}$$

with erf being the error function.

APPENDIX B: PHOTON NUMBER PENALTY

In this Appendix, we detail the numerical calculation of the gradient $\partial \Phi_p/\partial s_k(j)$ of the photon number penalty to the performance index Φ_p defined in Eq. (25) of Sec. IV B. Using Appendix A, this can be translated into $\partial \Phi_p/\partial u_k(j)$ needed

for the update rule, Eq. (2). We show that even though Φ_p is the result of a time integration over the full duration of the reset process, the gradient can still be calculated using a single forward and a single modified backward evolution.

In order to calculate Φ_p numerically, we approximate the continuous integral of Eq. (25) by a discrete sum over the subpixels defined in Appendix A:

$$\Phi_p \approx \sum_{i=e,g} \sum_{n=0}^{M} \delta t\, \mathrm{Tr}(a^\dagger a \hat{L}_n \ldots \hat{L}_1 \rho_i(0)). \tag{B1}$$

Now, we need to find $\partial \Phi_p/\partial u_k(j)$.

The gradient of the integration over time of the mean value of an operator A is in general given by

$$\sum_{n=0}^{M} \delta t \frac{\partial \langle A \rangle_n}{\partial s_k(j)} = \sum_{n=0}^{M} \delta t\, \mathrm{Tr}\left(A \frac{\partial (\hat{L}_n \ldots \hat{L}_1)}{\partial s_k(j)} \rho \right) \tag{B2}$$

$$= \sum_{n=0}^{M} \delta t\, \mathrm{Tr}\left[A\, \hat{L}_n \ldots \hat{L}_{j+1} \left(\frac{\partial \hat{L}_j}{\partial s_k(j)} \right) \right.$$

$$\left. \times \hat{L}_{j-1} \ldots L_1 \rho(0) \right] \Theta(n-j), \tag{B3}$$

where we have used the Heaviside step function

$$\Theta(n) = \begin{cases} 0 & \text{if } n < 0 \\ 1 & \text{if } n \geqslant 0 \end{cases}. \tag{B4}$$

Using the linearity of the trace, we see that

$$\sum_{n=0}^{M} \delta t \frac{\partial \langle A \rangle_n}{\partial s_k(j)} = \mathrm{Tr}\left[\left(\sum_{n=0}^{M} \delta t\, \Theta(n-j) A\, \hat{L}_n \ldots \hat{L}_{j+1} \right) \right.$$

$$\left. \times \frac{\partial \hat{L}_j}{\partial s_k(j)} (\hat{L}_{j-1} \ldots L_1 \rho(0)) \right], \tag{B5}$$

such that the last parentheses inside the trace is the same as the forward evolution used for the calculation of Φ_0, while the first parenthesis is a stepwise backward evolution starting from the operator A. This backward evolution is equivalent to a sum over backward evolutions starting at all time steps. For example, for $j = M - 2$ the parenthesis reads $A\hat{L}_M \hat{L}_{M-1} + A\hat{L}_{M-1} + A = (A\hat{L}_M + A)\hat{L}_{M-1} + A$. Therefore we can rewrite the gradient of the photon number penalty as

$$\frac{\partial \Phi_p}{\partial s_k(j)} = \delta t \sum_{i=e,g} \mathrm{Tr}\left(\zeta_{M-j} \frac{\partial \hat{L}_j}{\partial s_k(j)} \rho_{j-1} \right), \tag{B6}$$

with the quantities $\zeta_{M-j} = a^\dagger a + \hat{L}_{j+1}^\dagger \zeta_{M-j+1}$ defined recursively starting from $\zeta_M = a^\dagger a$ and $\rho_j = \hat{L}_j \ldots \hat{L}_1 \rho(0)$ as defined in Eq. (10). The derivative $\partial \hat{L}_j/\partial s_k(j)$ is calculated as in Eq. (12). Thus, by adding $a^\dagger a$ to the result of the backward evolution at each time step, the scaling of the GRAPE algorithm is not affected by this more complicated performance index and the gradient of the penalty function is obtained by the calculation of only one forward and one modified backward evolution per qubit state considered.

[1] J. P. Palao and R. Kosloff, Phys. Rev. Lett. **89**, 188301 (2002).

[2] P. Treutlein, T. W. Hänsch, J. Reichel, A. Negretti, M. A. Cirone, and T. Calarco, Phys. Rev. A **74**, 022312 (2006).

[3] S. Schirmer, J. Mod. Opt. **56**, 831 (2009).

[4] M. M. Müller, D. M. Reich, M. Murphy, H. Yuan, J. Vala, K. B. Whaley, T. Calarco, and C. P. Koch, Phys. Rev. A **84**, 042315 (2011).

[5] F. Motzoi, J. M. Gambetta, P. Rebentrost, and F. K. Wilhelm, Phys. Rev. Lett. **103**, 110501 (2009).

[6] D. J. Egger and F. K. Wilhelm, Supercond. Sci. Technol. **27**, 014001 (2014).

[7] D. J. Egger and F. K. Wilhelm, Phys. Rev. A **90**, 052331 (2014).

[8] P. Doria, T. Calarco, and S. Montangero, Phys. Rev. Lett. **106**, 190501 (2011).

[9] F. Motzoi, J. M. Gambetta, S. T. Merkel, and F. K. Wilhelm, Phys. Rev. A **84**, 022307 (2011).

[10] S. J. Glaser, U. Boscain, T. Calarco, C. P. Koch, W. Köckenberger, R. Kosloff, I. Kuprov, B. Luy, S. Schirmer, T. Schulte-Herbrüggen, D. Sugny, and F. K. Wilhelm, Eur. Phys. J. D **69**, 279 (2015).

[11] M. Shapiro and P. Brumer, *Principles of the Quantum Control of Molecular Processes* (Wiley-VCH, Weinheim, 2003), Vol. 1, p. 250.

[12] C. Brif, R. Chakrabarti, and H. Rabitz, New J. Phys. **12**, 075008 (2010).

[13] N. C. Nielsen, C. Kehlet, S. J. Glaser, and N. Khaneja, *Optimal Control Methods in NM Spectroscopy. eMagRes* (John Wiley & Sons, New York, 2007).

[14] N. Timoney, V. Elman, S. Glaser, C. Weiss, M. Johanning, W. Neuhauser, and C. Wunderlich, Phys. Rev. A **77**, 052334 (2008).

[15] J. M. Chow, L. DiCarlo, J. M. Gambetta, F. Motzoi, L. Frunzio, S. M. Girvin, and R. J. Schoelkopf, Phys. Rev. A **82**, 040305 (2010).

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

[17] N. Didier, F. Qassemi, and A. Blais, Phys. Rev. A **89**, 013820 (2014).

[18] Y. Liu, S. Shankar, N. Ofek, M. Hatridge, A. Narla, K. M. Sliwa, L. Frunzio, R. J. Schoelkopf, and M. H. Devoret, Phys. Rev. X **6**, 011022 (2016).

[19] F. Verstraete, M. M. Wolf, and J. Ignacio Cirac, Nat. Phys. **5**, 633 (2009).

[20] M. Wenin and W. Pötz, Phys. Rev. A **78**, 012358 (2008).

[21] S. Machnes, U. Sander, S. J. Glaser, P. de Fouquières, A. Gruslys, S. Schirmer, and T. Schulte-Herbrüggen, Phys. Rev. A **84**, 022305 (2011).

[22] T. Schulte-Herbrüggen, A. Spörl, N. Khaneja, and S. J. Glaser, J. Phys. B: At., Mol. Opt. Phys. **44**, 154013 (2011).

[23] F. F. Floether, P. de Fouquieres, and S. G. Schirmer, New J. Phys. **14**, 073023 (2012).

[24] M. H. Goerz, D. M. Reich, and C. P. Koch, New J. Phys. **16**, 055012 (2014).

[25] C. P. Koch, J. Phys.: Condens. Matter **28**, 213001 (2016).

[26] N. Khaneja, T. Reiss, C. Kehlet, T. Schulte-Herbrüggen, and S. J. Glaser, J. Magn. Reson. **172**, 296 (2005).

[27] V. Krotov, *Global Methods in Optimal Control Theory* (CRC Press, Boca Raton, FL, 1995), Vol. 195.

[28] Y. Maday and G. Turinici, J. Chem. Phys. **118**, 8191 (2003).

[29] T. Caneva, T. Calarco, and S. Montangero, Phys. Rev. A **84**, 022326 (2011).

[30] N. Rach, M. M. Müller, T. Calarco, and S. Montangero, Phys. Rev. A **92**, 062343 (2015).

[31] V. Engel, C. Meier, and D. J. Tannor, Adv. Chem. Phys. **141**, 29 (2009).

[32] C. Gardiner and P. Zoller, *Quantum Noise: A Handbook of Markovian and Non-Markovian Quantum Stochastic Methods with Applications to Quantum Optics* (Springer, New York, 2004).

[33] M. H. Goerz, *Optimizing Robust Quantum Gates in Open Quantum Systems*, Ph.D. thesis, Kassel University, 2015.

[34] G. Ashkenazi, R. Kosloff, S. Ruhman, and H. Tal-Ezer, J. Chem. Phys. **103**, 10005 (1995).

[35] M. H. Gutknecht, in *Frontiers of Computational Science* (Springer, New York, 2007), pp. 53–62.

[36] H. Tal-Ezer, SIAM J. Sci. Comput. **29**, 2426 (2007).

[37] L. C. G. Govia and F. K. Wilhelm, Phys. Rev. A **93**, 012316 (2016).

[38] L. C. G. Govia and F. K. Wilhelm, Phys. Rev. Appl. **4**, 054001 (2015).

[39] M. Khezri, E. Mlinar, J. Dressel, and A. N. Korotkov, Phys. Rev. A **94**, 012347 (2016).

[40] M. Boissonneault, J. M. Gambetta, and A. Blais, Phys. Rev. A **77**, 060305 (2008).

[41] A. P. Sears, A. Petrenko, G. Catelani, L. Sun, H. Paik, G. Kirchmair, L. Frunzio, L. I. Glazman, S. M. Girvin, and R. J. Schoelkopf, Phys. Rev. B **86**, 180504 (2012).

[42] C. Rigetti, J. M. Gambetta, S. Poletto, B. L. T. Plourde, J. M. Chow, A. D. Córcoles, J. A. Smolin, S. T. Merkel, J. R. Rozen, G. A. Keefe, M. B. Rothwell, M. B. Ketchen, and M. Steffen, Phys. Rev. B **86**, 100506 (2012).

[43] D. T. McClure, H. Paik, L. S. Bishop, M. Steffen, J. M. Chow, and J. M. Gambetta, Phys. Rev. Appl. **5**, 011001 (2016).

[44] C. C. Bultink, M. A. Rol, T. E. O'Brien, X. Fu, B. C. S. Dikken, C. Dickel, R. F. L. Vermeulen, J. C. de Sterke, A. Bruno, R. N. Schouten, and L. DiCarlo, Phys. Rev. Appl. **6**, 034008 (2016).

[45] A. G. Fowler, M. Mariantoni, J. M. Martinis, and A. N. Cleland, Phys. Rev. A **86**, 032324 (2012).

[46] A. A. Houck, J. A. Schreier, B. R. Johnson, J. M. Chow, J. Koch, J. M. Gambetta, D. I. Schuster, L. Frunzio, M. H. Devoret, S. M. Girvin, and R. J. Schoelkopf, Phys. Rev. Lett. **101**, 080502 (2008).

[47] S. Puri, S. Boutin, and A. Blais, npj Quantum Info. **3**, 18 (2017).

[48] R. Fletcher, *Practical Methods of Optimization* (John Wiley & Sons, New York, 2000).

[49] P. de Fouquieres, S. Schirmer, S. Glaser, and I. Kuprov, J. Magn. Reson. **212**, 412 (2011).

[50] M. Galassi, J. Davies, J. Theiler, B. Gough, G. Jungman, P. Alken, M. Booth, and F. Rossi, *GNU Scientific Library: Reference Manual* (Network Theory, Ltd., Godalming, Enland, 2011).

[51] J. Johansson, P. Nation, and F. Nori, Comput. Phys. Commun. **184**, 1234 (2013).

[52] A. Wallraff, D. Schuster, A. Blais, L. Frunzio, R.-S. Huang, J. Majer, S. Kumar, S. Girvin, and R. Schoelkopf, Nature (London) **431**, 162 (2004).

[53] A. Blais, R.-S. Huang, A. Wallraff, S. M. Girvin, and R. J. Schoelkopf, Phys. Rev. A **69**, 062320 (2004).

[54] M. Boissonneault, J. M. Gambetta, and A. Blais, Phys. Rev. Lett. **105**, 100504 (2010).

[55] J. Bourassa, F. Beaudoin, J. M. Gambetta, and A. Blais, Phys. Rev. A **86**, 013814 (2012).

[56] S. E. Nigg, H. Paik, B. Vlastakis, G. Kirchmair, S. Shankar, L. Frunzio, M. H. Devoret, R. J. Schoelkopf, and S. M. Girvin, Phys. Rev. Lett. **108**, 240502 (2012).

[57] R. Vijay, M. H. Devoret, and I. Siddiqi, Rev. Sci. Instrum. **80**, 111101 (2009).

[58] S. Lloyd and S. Montangero, Phys. Rev. Lett. **113**, 010502 (2014).

[59] J. J. W. H. Sørensen, M. K. Pedersen, M. Munch, P. Haikka, J. H. Jensen, T. Planke, M. G. Andreasen, M. Gajdacz, K. Mølmer, A. Lieberoth, and J. F. Sherson, Nature (London) **532**, 210 (2016).

[60] T. Caneva, M. Murphy, T. Calarco, R. Fazio, S. Montangero, V. Giovannetti, and G. E. Santoro, Phys. Rev. Lett. **103**, 240501 (2009).

[61] M. H. Goerz, T. Calarco, and C. P. Koch, J. Phys. B: At., Mol. Opt. Phys. **44**, 154011 (2011).

[62] V. Giovannetti, S. Lloyd, and L. Maccone, Phys. Rev. A **67**, 052109 (2003).

[63] L. B. Levitin and T. Toffoli, Phys. Rev. Lett. **103**, 160502 (2009).

[64] A. del Campo, I. L. Egusquiza, M. B. Plenio, and S. F. Huelga, Phys. Rev. Lett. **110**, 050403 (2013).

[65] M. M. Taddei, B. M. Escher, L. Davidovich, and R. L. de Matos Filho, Phys. Rev. Lett. **110**, 050402 (2013).

[66] M. Boissonneault, J. M. Gambetta, and A. Blais, Phys. Rev. A **79**, 013819 (2009).

[67] E. Martín-López, A. Laing, T. Lawson, R. Alvarez, X.-Q. Zhou, and J. L. O'Brien, Nat. Photon. **6**, 773 (2012).

[68] D. Riste, M. Dukalski, C. Watson, G. de Lange, M. Tiggelman, Y. M. Blanter, K. Lehnert, R. Schouten, and L. DiCarlo, Nature (London) **502**, 350 (2013).

[69] C. K. Andersen, J. Kerckhoff, K. W. Lehnert, B. J. Chapman, and K. Mølmer, Phys. Rev. A **93**, 012346 (2016).

[70] N. Roch, M. E. Schwartz, F. Motzoi, C. Macklin, R. Vijay, A. W. Eddins, A. N. Korotkov, K. B. Whaley, M. Sarovar, and I. Siddiqi, Phys. Rev. Lett. **112**, 170501 (2014).

[71] A. Córcoles, E. Magesan, S. J. Srinivasan, A. W. Cross, M. Steffen, J. M. Gambetta, and J. M. Chow, Nat. Commun. **6**, 6979 (2015).

[72] E. Magesan, J. M. Gambetta, A. D. Córcoles, and J. M. Chow, Phys. Rev. Lett. **114**, 200501 (2015).

[73] A. W. Cross and J. M. Gambetta, Phys. Rev. A **91**, 032325 (2015).