

# OPTIMAL CONTROL OF CLOSED QUANTUM SYSTEMS VIA B-SPLINES WITH CARRIER WAVES\*

N. ANDERS PETERSSON<sup>†</sup> AND FORTINO GARCIA<sup>‡</sup>

**Abstract.** We consider the optimal control problem of determining electromagnetic pulses for implementing logical gates in a closed quantum system, where the Hamiltonian models the dynamics of coupled superconducting qubits. The quantum state is governed by Schrödinger’s equation, which we formulate in terms of the real and imaginary parts of the state vector and solve by the Störmer-Verlet scheme, which is a symplectic partitioned Runge-Kutta method. A novel parameterization of the control functions based on B-splines with carrier waves is introduced. The carrier waves are used to trigger the resonant frequencies in the system Hamiltonian, and the B-spline functions specify their amplitude and phase. This approach allows the number of control parameters to be independent of, and significantly smaller than, the number of time steps for integrating Schrödinger’s equation.

We present numerical examples of how the proposed technique can be combined with an interior point L-BFGS algorithm for realizing quantum gates, and generalize our approach to calculate risk-neutral controls that are resilient to noise in the Hamiltonian model. The proposed method is also shown to compare favorably with QuTiP/pulse\_optim and Grape-Tensorflow.

**Key words.** Quantum control, B-splines, Symplectic Runge-Kutta method, ODE-constrained optimization, Quantum computing.

**AMS subject classifications.** 49M25, 65D07, 65L06, 81Q93

**1. Introduction.** We consider the quantum optimal control problem of determining electromagnetic pulses for implementing unitary gates in a quantum computer. A truncated modal expansion of Schrödinger’s equation is used to model the quantum system, in which the state of the quantum system is described by a state vector in an  $N$ -dimensional Hilbert space<sup>1</sup>  $\psi \in \mathbb{C}^N$ . The elements of the state vector are complex probability amplitudes, where the magnitude squared of each element represents the probability that the quantum system occupies the corresponding state [27]. Because the probabilities must sum to one, the state vector is normalized such that  $\|\psi\|_2^2 = 1$ . The evolution of the state vector in the time interval  $t \in [0, T]$  is governed by Schrödinger’s equation:

$$(1.1) \quad \frac{d\psi}{dt} + iH(t; \alpha)\psi = 0, \quad 0 \leq t \leq T, \quad \psi(0) = \mathbf{g}.$$

Here,  $i = \sqrt{-1}$  is the imaginary unit and  $\mathbf{g}$  is the initial state. The Hamiltonian matrix  $H(t; \alpha) \in \mathbb{C}^{N \times N}$  (scaled such that Planck’s constant becomes  $\hbar = 1$ ) is Hermitian and is assumed to be of the form

$$(1.2) \quad H(t; \alpha) = H_s + H_c(t; \alpha),$$

where  $H_s$  and  $H_c$  are the system and control Hamiltonians, respectively. The control Hamiltonian models the action of external control fields on the quantum system. The

\*Submitted to the editors on August 23, 2022

**Funding:** This work was supported in part by LLNL Laboratory Directed Research and Development project 20-ERD-028 and in part by DOE Office of Advanced Scientific Computing Research under the Advanced Research in Quantum Computing program, award 2019-LLNL-SCW-1683.

<sup>†</sup>Center for Applied Scientific Computing, LLNL, Livermore, CA 94550, USA ([petersson1@llnl.gov](mailto:petersson1@llnl.gov))

<sup>‡</sup>Department of Applied Mathematics, University of Colorado, Boulder, CO 80309, USA ([fortino.garcia@colorado.edu](mailto:fortino.garcia@colorado.edu))

<sup>1</sup>This paper uses conventional matrix-vector notation. Its basic relations with the bra-ket notation are  $|\psi\rangle = \psi$ ,  $\langle\psi| = \psi^\dagger$ ,  $\langle\phi|A|\psi\rangle = \phi^\dagger A\psi$ , and  $|\phi\rangle\langle\psi| = \phi\psi^\dagger$ .

time dependence in the control Hamiltonian is parameterized by the control vector  $\boldsymbol{\alpha} \in \mathbb{R}^D$ . As a result, the state vector  $\boldsymbol{\psi}$  depends implicitly on  $\boldsymbol{\alpha}$  through Schrödinger's equation.

Following Palao et al. [30], we divide the Hilbert space into an essential and a guarded subspace. The essential subspace is of dimension  $E$  and spans the states with the lowest energy levels in the system. These are the states used for quantum information processing. The guarded subspace has dimension  $G$ , such that  $E+G = N$ . The purpose of the guarded subspace is to eliminate any couplings to even higher states, which are excluded from the model.

The goal of the quantum optimal control problem is to determine the parameter vector  $\boldsymbol{\alpha}$  such that the time-dependence in the Hamiltonian matrix leads to a solution of Schrödinger's equation such that  $\boldsymbol{\psi}(T) \approx V_{tg}\mathbf{g}$ , where  $V_{tg}$  is the target gate transformation. The gate transformation should be satisfied for all initial conditions in the essential subspace of the state vector. A basis of this subspace is provided by the matrix  $U_0 \in \mathbb{R}^{N \times E}$ . The definitions of  $U_0$  and  $V_{tg}$  are described in Appendix A.

To account for any initial condition in the essential subspace, we define the solution operator matrix  $U \in \mathbb{C}^{N \times E}$ . Each column of this matrix satisfies (1.1), leading to Schrödinger's equation in matrix form,

$$(1.3) \quad \frac{dU(t)}{dt} + iH(t; \boldsymbol{\alpha})U(t) = 0, \quad 0 \leq t \leq T, \quad U(0) = U_0.$$

The overlap between the target gate matrix and the solution operator matrix at the final time is defined by  $O_T := \langle U(T; \boldsymbol{\alpha}), V_{tg} \rangle_F$ , where  $\langle \cdot, \cdot \rangle_F$  denotes the Frobenius matrix scalar product. Because  $U_0$  spans an  $E$ -dimensional subspace of initial conditions, we have  $|O_T| \leq E$ . The difference between  $U(T; \boldsymbol{\alpha})$  and  $V_{tg}$  can be measured by the target gate infidelity [18, 21, 24, 25, 34],

$$(1.4) \quad \mathcal{J}_1(U_T(\boldsymbol{\alpha})) := 1 - \frac{1}{E^2} |\langle U_T(\boldsymbol{\alpha}), V_{tg} \rangle_F|^2, \quad U_T(\boldsymbol{\alpha}) := U(T; \boldsymbol{\alpha}).$$

Note that the target gate infidelity is invariant to global phase differences between  $U_T$  and  $V_{tg}$ . In quantum physics, the global phase of a state is considered irrelevant because it cannot be measured.

If the evolution generated by optimal control pulses should explore some of the higher states in the guarded subspace, the truncation of the Hilbert space may introduce a false nonlinearity that can misguide the optimization [21]. Including additional states can, in principle, mitigate this problem, but is computationally expensive. Also, higher states can be difficult to model accurately. To address these challenges, we introduce an intermediate-time objective function to penalize population of states in the guarded subspace,

$$(1.5) \quad \mathcal{J}_2(U(\cdot; \boldsymbol{\alpha})) = \frac{1}{T} \int_0^T \langle U(t; \boldsymbol{\alpha}), WU(t; \boldsymbol{\alpha}) \rangle_F dt.$$

This term is called the leakage. Here,  $W$  is an  $N \times N$  Hermitian positive semi-definite matrix. To derive an appropriate expression for  $W$ , we start by defining a projection matrix  $P_e$ , such that  $\boldsymbol{\psi}_e = P_e\boldsymbol{\psi}$  corresponds to the essential part of the state vector  $\boldsymbol{\psi}$ . The population in the essential subspace is bounded by  $0 \leq \langle \boldsymbol{\psi}, P_e\boldsymbol{\psi} \rangle_2 \leq 1$ . Maximizing the population in the essential subspace corresponds to minimizing it in the guarded subspace, defined by the orthogonal projection  $P_g = (I - P_e)$ . Let the guarded subspace be spanned a basis of orthonormal forbidden states [21],  $\{\boldsymbol{\psi}_{f,k}\}_{k=1}^G$ ,

and construct the projection as  $P_g = \sum_k \psi_{f,k} \psi_{f,k}^\dagger$ . If there is more than one forbidden state, it may be desirable to introduce a state-dependent weight coefficient, leading to

$$(1.6) \quad W = \sum_{k=1}^G \gamma_k \psi_{f,k} \psi_{f,k}^\dagger, \quad \gamma_k > 0.$$

From the above construction follows that if all columns of the matrix  $U(t; \boldsymbol{\alpha})$  are in the essential subspace,  $WU(t; \boldsymbol{\alpha}) = \mathbf{0}$ . If that equality holds for all  $t \in [0, T]$ , we have  $\mathcal{J}_2 = 0$ , corresponding to zero population of the guarded subspace and maximal population of the essential subspace.

For the quantum control problem with guard states, we formulate the optimization problem as

$$(1.7) \quad \min_{\boldsymbol{\alpha}} \mathcal{G}(\boldsymbol{\alpha}) := \mathcal{J}_1(U_T(\boldsymbol{\alpha})) + \mathcal{J}_2(U(\cdot; \boldsymbol{\alpha})),$$

$$(1.8) \quad \frac{dU}{dt} + iH(t; \boldsymbol{\alpha})U = 0, \quad 0 \leq t \leq T, \quad U(0; \boldsymbol{\alpha}) = U_0,$$

$$(1.9) \quad \alpha_{min} \leq \alpha_q \leq \alpha_{max}, \quad q = 1, 2, \dots, D.$$

For a discussion of the solvability of the quantum control problem, see for example Borzi et al. [6].

There are several different approaches to numerically determine controls that yield high fidelity quantum logic gates. Perhaps some of the simplest approaches to implement are gradient-free optimization methods, such as the CRAB algorithm [8]. The CRAB algorithm is constructed by choosing a functional approximation of the control functions (Lagrange polynomials, Fourier basis, etc.) to truncate the infinite dimensional control problem into a finite dimensional, multivariate optimization problem. This optimization problem is solved using a direct search method without the need for a gradient, which typically comes at the cost of slower convergence compared to gradient-based techniques unless the number of control parameters is small.

For gradient based approaches, there are broadly two classes of methods for computing the gradient of the objective function. The first accumulates the gradient by forward time-stepping, as is done in the GOAT algorithm [25]. This approach allows the gradient of the objective function to be calculated exactly, but requires  $(D + 1)$  Schrödinger systems to be solved when the control functions depend on  $D$  parameters. This makes the method computationally expensive when the number of parameters is large. The second approach for computing the gradient solves an adjoint equation backwards in time. This approach, together with the use of a second order accurate Magnus scheme [17], leads to the GRAPE algorithm [19, 21]. Of note is that a stair-step approximation of the control functions is imposed such that each control function is constant within each time step. As a consequence, the time step determines both the numerical accuracy of the state vector, *and* the number of control parameters. With  $Q$  control functions, and  $M$  time steps of size  $h$ , the control functions are thus described by  $D = M \cdot Q$  parameters. For problems in which the duration of the gate is long, or the quantum state is highly oscillatory, it is clear that the size of the parameter space becomes very large, which may hamper the convergence of the GRAPE algorithm.

As seen with the CRAB algorithm, the total number of parameters can be reduced by instead expanding the control functions in terms of basis functions. By using the chain rule, the gradient from the GRAPE algorithm can then be used to calculate

the gradient with respect to the coefficients in the basis function expansion. This approach is implemented in the GRAFS algorithm [24], where the control functions are expanded in terms of Slepian sequences. Many other parameterizations of quantum control functions have been proposed in the literature, for example cubic splines [12], Gaussian pulse cascades [11], and Fourier expansions [39].

This paper presents a different approach, based on parameterizing the control functions by B-spline basis functions with carrier waves. We note that the idea of using carrier waves has been used extensively for controlling molecular systems with lasers, see e.g. [33, 10, 29]. The carrier wave approach relies on the observation that transitions between the energy levels in a quantum system are triggered by resonance, at frequencies that can be determined by the system Hamiltonian. The carrier waves are used to specify the frequency spectra of the control functions, while the B-spline functions specify their envelope and phase, leading to narrow-band control functions where the frequency content is concentrated near the resonant frequencies of the system. We find that our approach allows the number of control parameters to be independent of, and significantly smaller than, the number of time steps for integrating Schrödinger’s equation.

The remainder of the paper is organized as follows. In Section 2, we introduce a Hamiltonian model and discuss the resonant frequencies needed to trigger transitions between the states in the system. These resonant frequencies naturally motivate us to parameterize the control functions using B-splines with carrier waves; details of this parameterization are presented in Section 3. In Section 4, we introduce a real-valued formulation of Schrödinger’s equation and present the symplectic Störmer-Verlet scheme that we use for its time-integration. To achieve an exact gradient of the discrete objective function, we apply the “first-discretize-then-optimize” approach. Based on the Störmer-Verlet scheme, in Section 5 we outline the construction of a discrete adjoint time integration method. Section 6 presents numerical examples. We illustrate how the proposed technique, combined with the interior point L-BFGS algorithm [28] from the IPOPT package [37], is used to optimize control functions for multi-level qudit gates. We additionally consider a simple noise model and risk-neutral optimization to illustrate the construction of controls that are robust to uncertainty in the Hamiltonian. The proposed scheme is implemented in the Julia [3] programming language, in an open source package called Juqbox.jl [13]. In Section 7, we compare the performance of Juqbox.jl and two implementations of the GRAPE algorithm. Concluding remarks are given in Section 8.

**2. Hamiltonian model.** The Hamiltonian for describing the quantum physics of super-conducting circuits is of the general form

$$H(t) = H_s + H_c(t),$$

where  $H_s$  is the system Hamiltonian and  $H_c(t)$  is the control Hamiltonian. Both terms are Hermitian matrices. The time-independent system Hamiltonian is in general not diagonal but, because it is Hermitian, always admits an eigenvalue decomposition  $H_s = V\tilde{H}_sV^\dagger$ , where  $V$  is unitary and  $\tilde{H}_s$  is diagonal. A change of basis via the eigenvectors of  $H_s$  yields the transformed Hamiltonian

$$(2.1) \quad \tilde{H}(t) = \tilde{H}_s + \tilde{H}_c(t), \quad \tilde{H}_c(t) = V^\dagger H_c(t)V,$$

with a diagonal time-independent Hamiltonian  $\tilde{H}_s$ .

The eigen-decomposition of a general  $N \times N$  system Hamiltonian can be calculated numerically as long as the dimension of the Hilbert space is not too large (for

a spin-chain Hamiltonian, the size limit has been estimated to  $N \leq 5 \cdot 10^4$  on present day computers [35]). Analytical formulae for diagonalizing  $H_s$  are in general difficult to derive. However, approximate techniques based on Bogoliubov [5] transformations and Schrieffer-Wolff [7] expansions can in some cases be used to diagonalize the system Hamiltonian. For example, a quantum system with  $Q$  components consisting of transmon qubits coupled to resonators can, in the dispersive limit, be modeled by (see [4] and Appendix D)

$$(2.2) \quad H_s = \sum_{q=1}^Q \left( \omega_q a_q^\dagger a_q - \frac{\xi_q}{2} a_q^\dagger a_q^\dagger a_q a_q - \sum_{p>q} \xi_{pq} a_p^\dagger a_p a_q^\dagger a_q \right).$$

In the above formulae,  $\omega_q$  is the ground state transition frequency and  $\xi_q$  is the self-Kerr coefficient of subsystem  $q$ ; the cross-Kerr coefficient between subsystems  $p$  and  $q$  is denoted  $\xi_{pq}$ . In the above formula, subsystem  $q$  is assumed to have  $n_q \geq 2$  energy levels, with lowering operator  $a_q$ . The lowering operator is constructed using Kronecker products,

$$(2.3) \quad a_q := I_{n_Q} \otimes \cdots \otimes I_{n_{q+1}} \otimes A_q \otimes I_{n_{q-1}} \otimes \cdots \otimes I_{n_1} \in \mathbb{R}^{N \times N}, \quad N = \prod_{q=1}^Q n_q,$$

where  $I_n$  denotes the  $n \times n$  identity matrix and the single-system lowering matrix satisfies

$$(2.4) \quad A_q := \begin{pmatrix} 0 & \sqrt{1} & & & \\ & \ddots & \ddots & & \\ & & \ddots & \ddots & \\ & & & \ddots & \sqrt{n_q - 1} \\ & & & & 0 \end{pmatrix} \in \mathbb{R}^{n_q \times n_q}.$$

We consider a control Hamiltonian with real-valued control functions that are parameterized by the control vector  $\boldsymbol{\alpha}$ ,

$$(2.5) \quad H_c(t; \boldsymbol{\alpha}) = \sum_{q=1}^Q f_q(t; \boldsymbol{\alpha}) (a_q + a_q^\dagger), \quad f_q(t; \boldsymbol{\alpha}) = 2 \operatorname{Re} (d_q(t; \boldsymbol{\alpha}) e^{i\omega_{r,q}t}).$$

where  $\omega_{r,q}$  is the drive frequency in subsystem  $q$ .

**2.1. Rotating wave approximation.** To slow down the time scales in the state vector, we apply a rotating frame transformation in Schrödinger's equation through the unitary change of variables  $\tilde{\psi}(t) = R(t)\psi(t)$ , where

$$(2.6) \quad R(t) = \bigotimes_{q=Q}^1 \exp(i\omega_{r,q}t A_q^\dagger A_q),$$

and  $\bigotimes_{q=Q}^1 C_q = C_Q \otimes C_{Q-1} \otimes \cdots \otimes C_1$ . Note that we use  $\omega_{r,q}$  as the frequency of rotation in subsystem  $q$ . The system Hamiltonian transforms into  $H_s^{rw} = H_s - \sum \omega_{r,q} a_q^\dagger a_q$ . Then, the rotating wave approximation is applied to transform the control Hamiltonian. Here, we substitute the laboratory frame control function  $f_q(t; \boldsymbol{\alpha})$  from (2.5) and

neglect terms oscillating with frequencies  $\pm 2\omega_{r,q}$ . As a result, the Hamiltonians (2.2) and (2.5) transform into (see Appendix B for details)

$$(2.7) \quad H_s^{rw} = \sum_{q=1}^Q \left( \Delta_q a_q^\dagger a_q - \frac{\xi_q}{2} a_q^\dagger a_q^\dagger a_q a_q - \sum_{p>q} \xi_{qp} a_q^\dagger a_q a_p^\dagger a_p \right),$$

$$(2.8) \quad H_c^{rw}(t; \boldsymbol{\alpha}) = \sum_{q=1}^Q (d_q(t; \boldsymbol{\alpha}) a_q + \bar{d}_q(t; \boldsymbol{\alpha}) a_q^\dagger),$$

where  $\Delta_q = \omega_q - \omega_{r,q}$  is called the detuning frequency. The main advantages of the rotating wave approximation are the reduction of the spectral radius in the system Hamiltonian (2.7), and the absence of the highly oscillatory factor  $\exp(i\omega_{r,q}t)$  in the control Hamiltonian (2.8). In the following we assume that the rotating wave approximation has already been performed, and the tilde on the state vector will be suppressed. We additionally note that the target unitary  $V_{t_g}$  is similarly transformed into the rotating frame via  $V_{t_g}^{rw} = R(T)V_{t_g}$ .

**2.2. Resonant frequencies.** To simplify the presentation we restrict our analysis to a bipartite quantum system, i.e.,  $Q = 2$ . While our methodology applies to general Hamiltonians, the main elements of our approach can be described by a model where the system Hamiltonian is diagonal and the control Hamiltonian of the form (2.8). The general case can be handled in an analogous way by first transforming the system Hamiltonian to diagonal form, resulting in a transformed control Hamiltonian as in (2.1). An example of the diagonalization approach is presented in Section 6.1.

In the following, we assume the system Hamiltonian to be of the form (2.7) and denote its elements by

$$(2.9) \quad H_s^{rw}|_{\mathbf{j}, \mathbf{k}} = \begin{cases} \kappa_{\mathbf{j}}, & \mathbf{j} = \mathbf{k}, \\ 0, & \text{otherwise,} \end{cases} \quad \kappa_{\mathbf{j}} = \sum_{q=1}^2 \left( \Delta_q j_q - \frac{\xi_q}{2} j_q(j_q - 1) \right) - \xi_{12} j_1 j_2,$$

for  $j_q \in [0, n_q - 1]$  and where  $\mathbf{j} = (j_2, j_1)$  is a multi-index.

Let us consider the case when the control functions  $d_k(t)$  oscillate with carrier wave frequencies  $\{\Omega_1, \Omega_2\}$ , and a small amplitude  $\epsilon$ . These assumptions give

$$(2.10) \quad H_c^{rw}(t) = \epsilon H_1(t), \quad H_1(t) = \sum_{k=1}^2 \left( e^{i\Omega_k t} a_k + e^{-i\Omega_k t} a_k^\dagger \right), \quad 0 < \epsilon \ll 1.$$

We proceed by identifying the carrier wave frequencies that trigger resonance. We make an asymptotic expansion of the solution of Schrödinger's equation (1.1),  $\boldsymbol{\psi} = \boldsymbol{\psi}^{(0)} + \epsilon \boldsymbol{\psi}^{(1)} + \mathcal{O}(\epsilon^2)$ . In this setting, resonance corresponds to algebraic growth in time of the amplitude of the first order perturbation. The zero'th and first order terms satisfy

$$(2.11) \quad \frac{d\boldsymbol{\psi}^{(0)}}{dt} + iH_s^{rw} \boldsymbol{\psi}^{(0)} = 0, \quad \boldsymbol{\psi}^{(0)}(0) = \mathbf{g},$$

$$(2.12) \quad \frac{d\boldsymbol{\psi}^{(1)}}{dt} + iH_s^{rw} \boldsymbol{\psi}^{(1)} = \mathbf{f}(t), \quad \boldsymbol{\psi}^{(1)}(0) = \mathbf{0}.$$

Because the system Hamiltonian is diagonal, (2.11) is a decoupled system of ordinary differential equation that is solved by  $\psi_j^{(0)}(t) = g_j e^{-i\kappa_j t}$ . The right hand side of (2.12)

satisfies  $\mathbf{f}(t) := -iH_1(t)\psi^{(0)}(t)$ , which can be written

$$(2.13) \quad \mathbf{f}(t) = \sum_{k=1}^Q \mathbf{f}^{(k)}(t), \quad \mathbf{f}^{(k)}(t) = -i \left( e^{i\Omega_k t} a_k + e^{-i\Omega_k t} a_k^\dagger \right) \psi^{(0)}(t).$$

Because the matrix  $H_s^{rw}$  is diagonal, the system for the first order perturbation, (2.12), is also decoupled. We are interested in cases when  $\psi_{\mathbf{j}}^{(1)}(t)$  grows in time, corresponding to resonance. Let  $\mathbf{e}_k$  denote the  $k^{\text{th}}$  unit vector and denote a shifted multi-index by  $\mathbf{j} \pm \mathbf{e}_1 = (j_2, j_1 \pm 1)$  and  $\mathbf{j} \pm \mathbf{e}_2 = (j_2 \pm 1, j_1)$ .

LEMMA 2.1. *The perturbation of the state vector,  $\psi_{\mathbf{j}}^{(1)}(t)$ , grows linearly in time when the carrier wave frequencies and the initial condition satisfy:*

$$(2.14) \quad \Omega_k = \kappa_{\mathbf{j}+\mathbf{e}_k} - \kappa_{\mathbf{j}}, \quad g_{\mathbf{j}+\mathbf{e}_k} \neq 0, \quad j_k \in [0, n_k - 2],$$

$$(2.15) \quad \Omega_k = \kappa_{\mathbf{j}} - \kappa_{\mathbf{j}-\mathbf{e}_k}, \quad g_{\mathbf{j}-\mathbf{e}_k} \neq 0, \quad j_k \in [1, n_k - 1],$$

for  $k = \{1, 2\}$ .

*Proof.* See Appendix C. □

We can evaluate the conditions for resonance by inserting the Hamiltonian elements from (2.9) into (2.14) and (2.15). For  $k = 1$  and  $j_2 \in [0, n_2 - 1]$ , resonance occurs in  $\psi_{\mathbf{j}}^{(1)}(t)$  when

$$(2.16) \quad \Omega_1 = \begin{cases} \Delta_1 - \xi_{1j_1} - \xi_{12}j_2, & g_{\mathbf{j}+\mathbf{e}_1} \neq 0, \quad j_1 \in [0, n_1 - 2], \\ \Delta_1 - \xi_{1(j_1-1)} - \xi_{12}j_2, & g_{\mathbf{j}-\mathbf{e}_1} \neq 0, \quad j_1 \in [1, n_1 - 1]. \end{cases}$$

For  $k = 2$  and  $j_1 \in [0, n_1 - 1]$ , the resonant cases are

$$(2.17) \quad \Omega_2 = \begin{cases} \Delta_2 - \xi_{2j_2} - \xi_{12}j_1, & g_{\mathbf{j}+\mathbf{e}_2} \neq 0, \quad j_2 \in [0, n_2 - 2], \\ \Delta_2 - \xi_{2(j_2-1)} - \xi_{12}j_1, & g_{\mathbf{j}-\mathbf{e}_2} \neq 0, \quad j_2 \in [1, n_2 - 1]. \end{cases}$$

For example, when  $n_1 = 3, n_2 = 3$  and  $g_{\mathbf{j}} \neq 0 \forall \mathbf{j}$ , the carrier wave frequencies:

$$\begin{aligned} \Omega_1 &= \left[ \Delta_1, \quad \Delta_1 - \xi_{12}, \quad \Delta_1 - 2\xi_{12}, \quad \Delta_1 - \xi_1, \quad \Delta_1 - \xi_1 - \xi_{12}, \quad \Delta_1 - \xi_1 - 2\xi_{12} \right], \\ \Omega_2 &= \left[ \Delta_2, \quad \Delta_2 - \xi_{12}, \quad \Delta_2 - 2\xi_{12}, \quad \Delta_2 - \xi_2, \quad \Delta_2 - \xi_2 - \xi_{12}, \quad \Delta_2 - \xi_2 - 2\xi_{12} \right], \end{aligned}$$

lead to resonance.

REMARK 1. *The purpose of the above analysis is to identify carrier wave frequencies that trigger resonance, i.e., algebraic growth in time of the amplitude of the first order perturbation. While the asymptotic expansion is only valid in the limit of small amplitudes, it is sufficient for determining the onset of resonance. Since Schrödinger's equation conserves total probability, the linear growth only occurs for short times.*

**3. Quadratic B-splines with carrier waves.** Motivated by the results from the previous section, we parameterize the rotating frame control functions using basis functions that act as envelopes for carrier waves with fixed frequencies:

$$(3.1) \quad d_k(t; \boldsymbol{\alpha}) = \sum_{n=1}^{N_f} d_{k,n}(t; \boldsymbol{\alpha}), \quad d_{k,n}(t; \boldsymbol{\alpha}) = \sum_{b=1}^{N_b} \hat{S}_b(t) \alpha_{b,n}^k e^{it\Omega_{k,n}}, \quad k \in [1, Q].$$

Here,  $\Omega_{k,n} \in \mathbb{R}$  is the  $n^{\text{th}}$  carrier wave frequency for system  $k$ . These frequencies are chosen to match the resonant frequencies in the system Hamiltonian (2.7), as outlined above. The complex coefficients  $\alpha_{b,n}^k = \alpha_{b,n}^{k(r)} + i\alpha_{b,n}^{k(i)}$  are control parameters that are to be determined through optimization, corresponding to a total of  $D = 2QN_bN_f$  real-valued parameters. It is convenient to also define the real-valued functions

$$(3.2) \quad p_{k,n}(t; \boldsymbol{\alpha}) = \sum_{b=1}^{N_b} \hat{S}_b(t) \alpha_{b,n}^{k(r)}, \quad q_{k,n}(t; \boldsymbol{\alpha}) = \sum_{b=1}^{N_b} \hat{S}_b(t) \alpha_{b,n}^{k(i)},$$

such that  $d_{k,n}(t; \boldsymbol{\alpha}) = (p_{k,n}(t; \boldsymbol{\alpha}) + iq_{k,n}(t; \boldsymbol{\alpha})) \exp(it\Omega_{k,n})$ .

There are several options for choosing the basis functions  $\hat{S}_b(t)$  in (3.1). For example, a Fourier or Chebyshev expansion has orthogonal basis functions and the Fourier basis has compact support in frequency space. However, using either of these bases makes it difficult to construct control functions that begin and end with zero amplitude, which may be desirable in an experimental implementation. Furthermore, the global support in time implies that each coefficient in the Fourier or Chebyshev expansion modifies the control function for all times, which may hamper the convergence of the optimization algorithm. An interesting alternative to these classical expansions is provided by semi-orthogonal B-spline wavelets, which can be optimally localized in both time and frequency [36]. Here, each basis function is a smooth polynomial function of time, where the smoothness depends on the spline order. Polynomial functions are inexpensive to evaluate and lead to a computationally efficient implementation. The basis functions have local support in time, making it easy to construct control functions that begin and end at zero. In this work we use quadratic B-splines, where at most three B-spline coefficients influence the control function at any time  $t$ .

The B-spline basis functions  $\hat{S}_b(t)$  are centered on a uniform grid in time (see Figure 1),

$$(3.3) \quad \tau_b = \Delta\tau(b - 1.5), \quad b = 1, \dots, N_b, \quad \Delta\tau = \frac{T}{N_b - 2}.$$

The basis function  $\hat{S}_b(t)$  has local support for  $t \in [\tau_b - 1.5\Delta\tau, \tau_b + 1.5\Delta\tau]$ . Thus, for any fixed time  $t$  a control function will get contributions from at most three B-spline wavelets, allowing the control functions to be evaluated very efficiently. We also remark that the control function (3.1) can be evaluated at any time  $t \in [0, T]$ . Importantly, this allows the time-integration scheme to be chosen independently of the parameterization of the control function, and allows the number of control parameters to be chosen independently of the number of time steps for integrating Schrödinger's equation.

**4. Real-valued formulation.** There are many accurate numerical schemes for integrating the complex-valued Schrödinger equation (1.1) in time. However, by first posing it as a Hamiltonian system in terms of the real and imaginary parts of the state vector, it becomes possible to use a partitioned Runge-Kutta method [17]. As will be explained in more detail below, such methods can be twice as efficient for solving Schrödinger's equation compared to a regular (non-partitioned) Runge-Kutta scheme.

A real-valued formulation of Schrödinger's equation (1.1) is given by

$$(4.1) \quad \begin{bmatrix} \dot{\mathbf{u}} \\ \dot{\mathbf{v}} \end{bmatrix} = \begin{bmatrix} S(t) & -K(t) \\ K(t) & S(t) \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \mathbf{v} \end{bmatrix} =: \begin{bmatrix} f^u(\mathbf{u}, \mathbf{v}, t) \\ f^v(\mathbf{u}, \mathbf{v}, t) \end{bmatrix}, \quad \begin{bmatrix} \mathbf{u}(0) \\ \mathbf{v}(0) \end{bmatrix} = \begin{bmatrix} \mathbf{g}^u \\ \mathbf{g}^v \end{bmatrix},$$



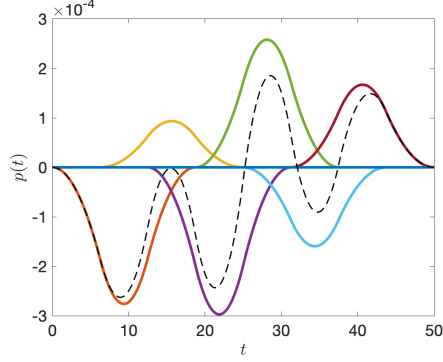


Fig. 1: The real part of a quadratic B-spline control function, with zero carrier frequency (dashed black). The solid colored lines are the individual B-spline wavelets.

where,

$$\mathbf{u} = \text{Re}(\boldsymbol{\psi}), \quad \mathbf{v} = -\text{Im}(\boldsymbol{\psi}), \quad K = \text{Re}(H), \quad S = \text{Im}(H),$$

Because the matrix  $H$  is Hermitian,  $K^T = K$  and  $S^T = -S$  (note that the matrix  $S$  is unrelated to the matrix overlap function  $S_V$ ). The real-valued formulation of Schrödinger's equation is a time-dependent Hamiltonian system corresponding to the Hamiltonian functional,

$$(4.2) \quad \mathcal{H}(\mathbf{u}, \mathbf{v}, t) = \mathbf{u}^T S(t) \mathbf{v} + \frac{1}{2} \mathbf{u}^T K(t) \mathbf{u} + \frac{1}{2} \mathbf{v}^T K(t) \mathbf{v}.$$

In general,  $S(t) \neq 0$ , which makes the Hamiltonian system non-separable.

In terms of the real-valued formulation, let the columns of the solution operator matrix in (1.3) satisfy  $U = [\mathbf{u}_1 - i\mathbf{v}_1, \mathbf{u}_2 - i\mathbf{v}_2, \dots, \mathbf{u}_E - i\mathbf{v}_E]$ . Here,  $(\mathbf{u}_j, \mathbf{v}_j)$  satisfy (4.1) subject to the initial conditions  $\mathbf{g}_j^v = \mathbf{0}$  and  $\mathbf{g}_j^u = \mathbf{e}_j$ , where  $\mathbf{j} = (j_Q, j_{Q-1}, \dots, j_1)$  is a multi-index such that  $j_q \in \{0, 1, \dots, m_q - 1\}$  and  $m_q$  is the number of essential levels of subsystem  $q$ . The columns in the target gate matrix  $V_{tg} = [\mathbf{d}_1, \dots, \mathbf{d}_E]$  correspond to

$$V_{tg} = [\mathbf{d}_1^u - i\mathbf{d}_1^v, \mathbf{d}_2^u - i\mathbf{d}_2^v, \dots, \mathbf{d}_E^u - i\mathbf{d}_E^v], \quad \mathbf{d}_j^u = \text{Re}(\mathbf{d}_j), \quad \mathbf{d}_j^v = -\text{Im}(\mathbf{d}_j).$$

Using the real-valued notation, the two parts of the objective function (1.7) can be written

$$(4.3) \quad \mathcal{J}_1(U_T(\boldsymbol{\alpha})) = \left( 1 - \frac{1}{E^2} |S_V(U_T(\boldsymbol{\alpha}))|^2 \right),$$

$$(4.4) \quad \mathcal{J}_2(U(\cdot, \boldsymbol{\alpha})) = \frac{1}{T} \sum_{j=1}^E \int_0^T \langle \mathbf{u}_j(t, \boldsymbol{\alpha}) - i\mathbf{v}_j(t, \boldsymbol{\alpha}), W(\mathbf{u}_j(t, \boldsymbol{\alpha}) - i\mathbf{v}_j(t, \boldsymbol{\alpha})) \rangle_2 dt,$$

where

$$(4.5) \quad S_V(U_T) = \sum_{j=1}^E \langle \mathbf{u}_j(T, \boldsymbol{\alpha}) - i\mathbf{v}_j(T, \boldsymbol{\alpha}), \mathbf{d}_j^u - i\mathbf{d}_j^v \rangle_2.$$

**4.1. Time integration.** Let  $t_n = nh$ , for  $n = 0, 1, \dots, M$ , be a uniform grid in time where  $h = T/M$  is the time step. Also let  $\mathbf{u}^n \approx \mathbf{u}(t_n)$  and  $\mathbf{v}^n \approx \mathbf{v}(t_n)$  denote the numerical solution on the grid. We use the Störmer-Verlet (S-V) scheme to discretize the real-valued formulation of Schrödinger's equation. It is a partitioned Runge-Kutta (PRK) scheme that is symplectic, reversible, and second order accurate [17]. Starting from the initial conditions  $\mathbf{u}^0 = \mathbf{g}^u$  and  $\mathbf{v}^0 = \mathbf{g}^v$ , the S-V scheme combines the trapezoidal and the implicit midpoint rules,

$$(4.6) \quad \mathbf{v}^{n+1/2} = \mathbf{v}^n + \frac{h}{2} \left( K_{n+1/2} \mathbf{u}^n + S_{n+1/2} \mathbf{v}^{n+1/2} \right),$$

$$(4.7) \quad \mathbf{u}^{n+1} = \mathbf{u}^n + \frac{h}{2} \left( S_n \mathbf{u}^n + S_{n+1} \mathbf{u}^{n+1} - (K_n + K_{n+1}) \mathbf{v}^{n+1/2} \right),$$

$$(4.8) \quad \mathbf{v}^{n+1} = \mathbf{v}^{n+1/2} + \frac{h}{2} \left( K_{n+1/2} \mathbf{u}^{n+1} + S_{n+1/2} \mathbf{v}^{n+1/2} \right),$$

for  $n = 1, 2, \dots, M - 1$ .

To evaluate the computational efficiency of the S-V method, we compare it to the Implicit Midpoint Rule (IMR), which is a non-partitioned Runge-Kutta method that also is symplectic, reversible, and second order accurate. Because the matrix  $S(t) \neq 0$  in (4.1), the S-V method requires two linear systems to be solved per time-step. The linear systems are diagonally dominant and can be solved using a Neumann iteration. The computational cost of each time step is dominated by solving these linear systems, which are of size  $N \times N$  with band width  $b$ . The number of real-valued operations for evaluating the matrix-vector products is  $\mathcal{O}(bN)$ . Because linear systems must be solved to calculate  $\mathbf{v}^{n+1/2}$  and  $\mathbf{u}^{n+1}$  in the S-V method, the total cost of taking one time step is  $\mathcal{O}((5 + 2N_{iter})bN)$ , where  $N_{iter}$  is the number of iterations needed to converge the solution (often  $N_{iter} \leq 5$ ). If, instead, we would have used the non-partitioned IMR time integrator, the corresponding real-valued linear system would consist of 4 blocks, each of size  $N \times N$ . Each block is banded with the same band width as before. This system is also diagonally dominant and can be solved by the same iterative scheme. The cost to evaluating these matrix vector products is  $\mathcal{O}((4 + 4N_{iter})bN)$ . Note that the cost of solving the corresponding complex-valued system with the IMR method is also of this order because multiplying two complex numbers requires 4 real-valued multiplications. The upshot is that the S-V method is about twice as efficient as the IMR method.

**4.2. Time step restrictions for accuracy and stability.** For simplicity, in the following we consider a single qubit system ( $Q = 1$ ) and note that the analysis can be extended to multiple systems in a straightforward manner. The accuracy in the numerical solution of Schrödinger's equation is essentially determined by resolving each of two fundamental time scales on the grid in time. The first time scale corresponds to the resonance frequencies in the control functions for triggering desired transitions between energy levels in the quantum system (as discussed in Section 2.2). In the Hamiltonian model (2.7) and (2.8), the angular transition frequencies between the essential energy levels (with detuning frequency  $\Delta_1$ ) are

$$\Omega_{1,n} = \Delta_1 - n\xi_1, \quad n = 0, \dots, N_f - 1.$$

The second time scale is due to the harmonic oscillation of the phase in the state vector. Recall that the Hamiltonian  $H \in \mathbb{C}^{N \times N}$  is Hermitian so that if  $\lambda$  is a real eigenvalue of  $H$  then the system matrix  $-iH$  has the eigenvalue  $-i\lambda$ . Thus, the harmonic oscillation corresponding to the eigenvalue of largest magnitude of  $H$  gives

the shortest period. A straightforward bound on the eigenvalue of largest magnitude of the Hamiltonian (2.7) and (2.8) can be obtained using Gershgorin’s circle theorem [16],

$$\rho(H) \leq \frac{|\xi_1|}{2}(N-1)(N-2) + 2d_\infty\sqrt{N-1} \equiv \rho_{\max}.$$

Here we have used that the control function is bounded by  $d_\infty = \max_t |d_1(t, \boldsymbol{\alpha})|$  for a given parameter vector  $\boldsymbol{\alpha}$ , in the interval  $0 \leq t \leq T$ . To resolve the shortest period in the solution of Schrödinger’s equation by at least  $C_P$  time steps requires

$$(4.9) \quad h \leq \frac{2\pi}{C_P \max\{\rho_{\max}, \max_n(|\Omega_{1,n}|)\}}.$$

The value of  $C_P$  that is needed to obtain a given accuracy in the numerical solution depends on the order of accuracy, the duration of the time integration, as well as the details of the time-stepping scheme. For second order accurate methods such as the Störmer-Verlet method, acceptable accuracy for engineering applications can often be achieved with  $C_P \approx 40$ . With the Störmer-Verlet method, we note that the time-stepping can become unstable if  $C_P \leq 2$ , corresponding to a sampling rate below the Nyquist limit.

**5. Discretizing the objective function and its gradient.** As discussed in the introduction, a powerful and efficient method to compute the gradient of the objective function is through the adjoint-state method. To that end, there are two approaches to compute the adjoint equation. The first is the “first-optimize-then-discretize” approach, in which the continuous adjoint equation is derived and then discretized independently of the forward (state) equation. The second is the “first-discretize-then-optimize” approach, in which a discretization is chosen for both the state equation and the objective function. With the chosen discretization, the Karush-Kuhn-Tucker (KKT) conditions [20] applied to a discrete Lagrangian yield an appropriate adjoint discretization scheme, which provides an *exact* discrete gradient. Here, we follow the “first-discretize-then-optimize” approach, as the former may yield inconsistent gradients. In the following we present a brief summary of the results of this approach. For the reader interested in the technical details of the derivation of the discrete adjoint scheme, we refer to [31].

Recall that the Störmer-Verlet scheme applied to the real-valued Schrödinger equation (4.1) is a partitioned Runge Kutta (PRK) scheme, essentially combining the trapezoidal and midpoint rules for the real and imaginary parts of the state vector, respectively. For the discretization of the guard level integral term in (4.4), we use the corresponding trapezoidal and midpoint rules as follows:

$$\mathcal{J}_2^h(\mathbf{u}, \mathbf{v}) = \frac{h}{2T} \sum_{j=1}^E \sum_{n=0}^{M-1} \left( \langle \mathbf{u}_j^n, W \mathbf{u}_j^n \rangle + \langle \mathbf{u}_j^{n+1}, W \mathbf{u}_j^{n+1} \rangle + 2 \langle \mathbf{v}_j^{n+1/2}, W \mathbf{v}_j^{n+1/2} \rangle \right).$$

The above expression assumes that  $W \in \mathbb{R}^{N \times N}$ . The general case of a complex-valued Hermitian  $W$  follows from straightforward algebra. The superscript  $h$  on  $\mathcal{J}_2^h$  denotes the discretized objective function. We note that the trace infidelity term  $\mathcal{J}_1$  in (4.3), is a terminal condition with the discrete analogue  $\mathcal{J}_1^h = \mathcal{J}_1$ . Thus we have the discrete version of the total objective function (1.7) as  $\mathcal{G}^h(\boldsymbol{\alpha}) = \mathcal{J}_1^h + \mathcal{J}_2^h$ . With these choices, a “first-discretize-then-optimize” approach leads to a partitioned Runge-Kutta (PRK) scheme that is a consistent approximation of the continuous

adjoint equation. The adjoint PRK scheme is closely related to the Störmer-Verlet scheme. However, the roles of the trapezoidal and midpoint rules are swapped. For example, the state variables  $\mathbf{u}$  are evolved with the trapezoidal rule in Störmer-Verlet, but the corresponding multiplier variables are evolved with the midpoint rule in the adjoint PRK scheme. In addition, the time-dependent matrices for the adjoint scheme are evaluated at slightly different time-levels, see [31] for further details.

**6. Numerical optimization.** Our numerical solution of the optimal control problem is based on the general purpose optimization package IPOPT [37]. This open-source library implements a primal-dual barrier approach for solving large-scale nonlinear programming problems, i.e., it minimizes an objective function subject to inequality (barrier) constraints on the parameter vector. Since the Hessian of the objective function is costly to calculate, we use the L-BFGS algorithm [28] in IPOPT, which only relies on the objective function and its gradient to be evaluated. Inequality constraints that limit the amplitude of the parameter vector  $\boldsymbol{\alpha}$  are enforced internally by IPOPT.

When solving an optimal control problem, the goal is to find a control vector that simultaneously gives a small gate infidelity and a small leakage. Because we minimize their sum, it is conceivable that the optimization algorithm would find an optima where only one of the terms is small. In practice this can occur when the Hilbert space is truncated too aggressively, leading to high leakage. This problem can be circumvented by inflating the guarded subspace. It can also happen that the optimizer converges to a solution where the infidelity is large. This behavior indicates that the bounds on the control vector are set too tight for the given gate duration. Increasing the bounds, or the gate duration, have been found to be effective ways of resolving that situation.

The routines for evaluating the objective function and its gradient are implemented in the Julia programming language [3], which provides a convenient interface to IPOPT. Given a parameter vector  $\boldsymbol{\alpha}$ , the routine for evaluating the objective function solves the Schrödinger equation with the Störmer-Verlet scheme and evaluates the objective function  $\mathcal{G}^h(\boldsymbol{\alpha})$  by accumulation. The routine for evaluating the gradient first applies the Störmer-Verlet scheme to calculate terminal conditions for the state variables. It then proceeds by accumulating the gradient  $\nabla_{\boldsymbol{\alpha}}\mathcal{G}^h$  by simultaneous reversed time-stepping of the discrete adjoint scheme and the Störmer-Verlet scheme. These two fundamental routines, together with many support functions are implemented in the software package Juqbox.jl [13]. This package was used to generate the numerical results below.

**6.1. A CNOT gate on two qudits with guard levels.** To test our methods on a realistic quantum optimal control problem, we consider a CNOT gate on two transmon qubits coupled by a resonator bus. Magesan and Gambetta [26] derived an effective Hamiltonian where the resonator modes are adiabatically eliminated. This model is used to describe the superconducting systems provided by IBM Quantum Experience [1]. In the rotating frame, the Hamiltonian  $H^{rw}(t) = H_s^{rw} + H_c^{rw}(t)$  satisfies

$$(6.1) \quad H_s^{rw} = \sum_{q=1}^2 \left( \Delta_q a_q^\dagger a_q - \frac{\xi_q}{2} a_q^\dagger a_q^\dagger a_q a_q \right) + J(a_1 a_2^\dagger + a_1^\dagger a_2),$$

$$(6.2) \quad H_c^{rw}(t) = \sum_{q=1}^2 (d_q(t) a_q + \bar{d}_q(t) a_q^\dagger).$$

Here, we have assumed that the frequency of rotation is the same in both subsystems, making the coupling term  $(a_1 a_2^\dagger + a_1^\dagger a_2)$  invariant under the rotating frame transformation. In the following, the frequency of rotation is chosen as the average of each qubit’s frequency,  $\omega_r = (\omega_1 + \omega_2)/2$ , resulting in the detuning frequencies  $\Delta_1 = -\Delta_2 = (\omega_2 - \omega_1)/2$ . We use representative parameters for qubits #3 and #5 from the `ibmq_jakarta` system, with qubit frequencies  $\omega_1/2\pi = 5.17839$  GHz and  $\omega_2/2\pi = 5.06323$  GHz. The self-Kerr coefficients are  $\xi_1/2\pi = 0.3411$  GHz and  $\xi_2/2\pi = 0.3413$  GHz, and the coupling coefficient is  $J/2\pi = 1.995 \cdot 10^{-3}$  GHz. Two guard levels are added to the two essential levels in each qubit, corresponding to a 16-dimensional Hilbert space for the state vector of the coupled system.

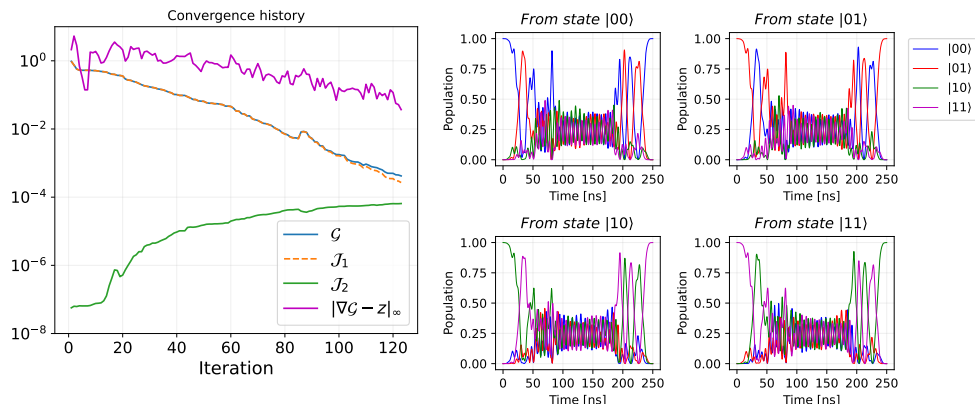
Because the system Hamiltonian is *not* diagonal, its resonant frequencies are determined after diagonalization,  $\tilde{H}_s = V^\dagger H_s^w V$ . Here we calculate the transformation numerically, but we could alternatively have utilized an approximate diagonalization, e.g., a Schrieffer-Wolff [7, 32] expansion. The transition frequencies in the system follow as differences between the eigenvalues of  $H_s^w$ , see Section 2.2. Because the coupling coefficient  $J$  is small compared to the detuning frequencies, the non-diagonal part of the Hamiltonian only imposes a small perturbation on the eigenvalues. As a result, the “dressed” detuning frequencies  $\tilde{\Delta}_1/2\pi \approx 5.7611 \cdot 10^{-2}$  GHz and  $\tilde{\Delta}_2 = -\tilde{\Delta}_1$  are very close to  $\Delta_1/2\pi = 5.7580 \cdot 10^{-2}$  GHz and  $\Delta_2 = -\Delta_1$ . A more significant outcome of the unitary transformation is that the control Hamiltonian in the first system becomes  $V^\dagger a_1 V \approx a_1 + \epsilon \tilde{a}_2$ , where  $\epsilon \approx 0.01$  and  $\tilde{a}_2$  has the same non-zero structure as  $a_2$ . Thus, in addition to triggering resonance in the first system, it can also trigger resonance in the second system (cross-resonance). We therefore apply two carrier wave frequencies in the first control function,  $\Omega_{1,1} = \tilde{\Delta}_1$  and  $\Omega_{1,2} = \tilde{\Delta}_2$ . The transformation of the second control Hamiltonian leads to a corresponding non-zero structure and we apply the same carrier wave frequencies in that control function. Note that the purpose of the diagonalization is to determine the carrier wave frequencies that trigger resonance between the essential states in the system. Once they are determined, we can solve Schrödinger’s equation using the original Hamiltonians.

To penalize occupation of the forbidden states that span the guarded subspace of the Hilbert space, we start by defining index ranges for the state vector. There are 4 levels in each subsystem; two essential and two guard levels. In this case, the Hilbert space is spanned by the basis states  $\mathcal{I} = \{|jk\rangle\}_{j,k=0}^3$  (here,  $|jk\rangle = e_j \otimes e_k$  where  $e_j \in \mathbb{R}^4$  is a canonical unit vector). The essential subspace in the coupled system is spanned by the basis states  $\mathcal{E} = \{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$  and we span the guarded subspace by the complimentary set of states  $\mathcal{F} = \mathcal{I} \setminus \mathcal{E}$ . The guarded subspace is further partitioned into a lower guarded subspace, spanned by the states in the set  $\mathcal{F}_1 = \{|02\rangle, |12\rangle, |20\rangle, |21\rangle\}$  and a higher guarded subspace,  $\mathcal{F}_2 = \mathcal{F} \setminus \mathcal{F}_1$ . We discourage population of forbidden states by constructing the weight matrix  $W$  as in (1.6), using the lower forbidden states  $\psi_{f,k} \in \mathcal{F}_1$  with weights  $\gamma_k = 1/7000$ , and the higher forbidden states  $\psi_{f,k} \in \mathcal{F}_2$  with weights  $\gamma_k = 1/7$ .

The control functions for each carrier wave are parameterized by B-splines with  $D_1 = 25$  basis functions resulting in a total of  $D = 200$  parameters. The amplitudes of the control functions are limited by the constraint

$$(6.3) \quad \|\alpha\|_\infty := \max_{1 \leq r \leq D} |\alpha_r| \leq \alpha_{max},$$

where we have set  $\alpha_{max} = 40$  MHz and enforce the control functions to begin and end at zero. We set the gate duration to  $T = 250$  ns and estimate the time step using the technique in Section 4.2. To guarantee at least  $C_P = 40$  time steps per period,



(a) IPOPT convergence history.

(b) Population of the essential states.

Fig. 2: The CNOT gate on two qudits.

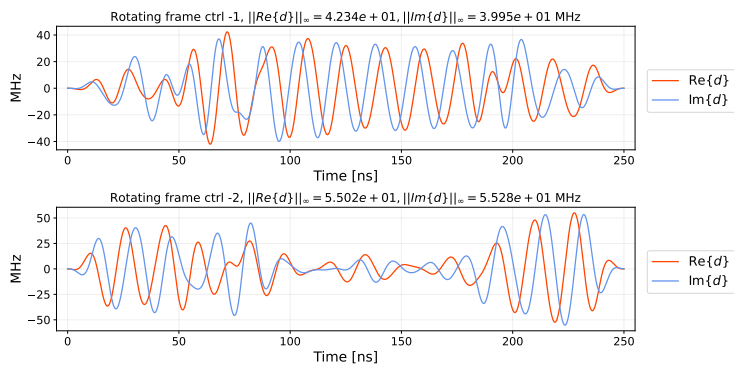


Fig. 3: The control functions (in the rotating frame) for qubit  $k = 1$  (top), and qubit  $k = 2$  (bottom), for realizing a CNOT gate with a fidelity exceeding 99.9%. The control function  $d(t)$  is defined in (3.1).

we use  $M = 20,750$  time steps, corresponding to  $h \approx 1.205 \cdot 10^{-2}$  ns. As initial guess for the elements of the parameter vector  $\alpha$ , we use a random number generator with a uniform distribution in  $[-\alpha_{max}/100, \alpha_{max}/100]$ . Additionally, we set the tolerance for the overall NLP error (see Equation (5) of [37]) to  $5 \cdot 10^{-2}$ .

In Figure 2a we present the convergence history of the optimization. We show the objective function  $\mathcal{G}$ , decomposed into  $\mathcal{J}_1^h$  and  $\mathcal{J}_2^h$ , together with the norm of the dual infeasibility,  $\|\nabla_{\alpha}\mathcal{G}-z\|_{\infty}$  (used by IPOPT to monitor convergence, see [37] for details). In this case, IPOPT converges to a solution with gate infidelity  $\mathcal{J}_1^h \approx 2.68 \cdot 10^{-4}$  in 122 iterations. Moreover,  $\mathcal{J}_2^h \approx 6.50 \cdot 10^{-5}$  and the population of the forbidden states remains small with a maximum population below  $5.60 \cdot 10^{-4}$  for all times and initial conditions. The evolution of the population of the essential states during the CNOT gate are presented in Figure 2b and the optimized control functions are shown in Figure 3.

**6.2. Risk-neutral controls.** In practice the entries of the Hamiltonian may have some uncertainty, especially for higher energy levels, and it is desirable to design control pulses that are more robust to noise. There are several ways to design noise resilient controls, including robust optimization methods in which a min-max problem is solved [14], sampling-based learning approaches for risk-neutral optimization [9], or risk-neutral/averse optimization to minimize the expectation of a utility function based on the original objective function subject to uncertain parameters [38, 15]. In the following we consider a risk-neutral optimization approach, which can be interpreted as designing a control to drive an ensemble of related quantum systems to the same final state [22, 23].

In this section we consider a risk-neutral strategy to design a  $|0\rangle \leftrightarrow |2\rangle$  SWAP gate on a single qubit ( $Q = 1$ ), with three essential levels and one guard level. Let  $\epsilon \sim \text{Unif}(-\epsilon_{\max}, \epsilon_{\max})$  be a uniform random variable for some  $\epsilon_{\max} > 0$ . As a simple example, we consider the uncertain system Hamiltonian  $H_s^u(\epsilon) = H_s^{rw} + H'(\epsilon)$  where  $H_s^{rw}$  is given by (2.7), and  $H'(\epsilon)$  is a diagonal perturbation:

$$\frac{H'(\epsilon)}{2\pi} = \begin{pmatrix} 0 & & & \\ & \epsilon/100 & & \\ & & \epsilon/10 & \\ & & & \epsilon \end{pmatrix}.$$

Here, no perturbation is imposed on the control Hamiltonian (2.8). From these assumptions follow that the uncertain system Hamiltonian has expectation  $\mathbb{E}[H_s^u(\epsilon)] = H_s^{rw}$ . We may correspondingly update the original objective function,  $\mathcal{G}(\boldsymbol{\alpha}, H_s^{rw})$ , to the risk-neutral utility function  $\tilde{\mathcal{G}}(\boldsymbol{\alpha}) = \mathbb{E}[\mathcal{G}(\boldsymbol{\alpha}, H_s^u(\epsilon))]$ . Given the simple form of the random variable  $\epsilon$ , we may compute  $\tilde{\mathcal{G}}$  by quadrature:

$$(6.4) \quad \mathbb{E}[\mathcal{G}(\boldsymbol{\alpha}, H_s^u(\epsilon))] = \int_{-\epsilon_{\max}}^{\epsilon_{\max}} \mathcal{G}(\boldsymbol{\alpha}, H_s^u(\epsilon)) d\epsilon \approx \sum_{k=1}^M w_k \mathcal{G}(\boldsymbol{\alpha}, H_s^u(\epsilon_k)),$$

where  $w_k$  and  $\epsilon_k$  are the weights and collocation points of a quadrature rule.

For the following example, we compare the optimal control obtained using the standard optimization procedure (no noise) and a risk-neutral control, in which the utility function (6.4) is computed using the Gauss-Legendre quadrature with  $N = 9$  collocation points and  $\epsilon_{\max} = 10$  MHz. We set the gate duration to  $T = 300$  ns, the maximum allowable amplitude to  $\alpha_{\max}/2\pi = 12$  MHz, the fundamental frequency to  $\omega_1/2\pi = 4.10336$  GHz, with detuning frequency  $\Delta_1 = 0$ , and the self-Kerr coefficient to  $\xi_1/2\pi = 0.2198$  GHz.

The control functions are constructed using two carrier waves with frequencies  $\Omega_{1,1} = 0$  and  $\Omega_{1,2} = -\xi_1$  for both the ‘‘noise-free’’ (NF) and ‘‘risk-neutral’’ (RN) cases. In each case we use  $D_1 = 12$  splines per control and carrier wave frequency for a total of  $D = 48$  splines. We additionally constrain the controls to start and end at zero. We set the tolerance for L-BFGS to  $10^{-5}$ , the maximum iteration count to 150, and use a maximum of five previous iterates to approximate the Hessian at each iteration. For the noise-free and risk-neutral optimized control functions, we use the perturbed Hamiltonian  $H_s^u(\epsilon)$  to evaluate the objective function  $\mathcal{G}$ , for evenly spaced  $\epsilon$  in the range  $[-30, 30]$  MHz. The results are shown in Figure 4. From Figure 4 we note that the optimal control corresponding to the noise-free approach obtains the smallest infidelity for  $\epsilon = 0$ , but it grows rapidly for  $|\epsilon| > 0$ . By comparison, the

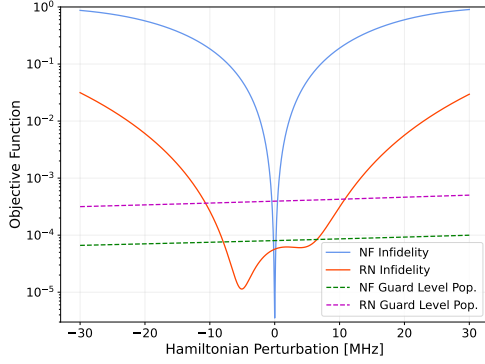


Fig. 4: Infidelity objective ( $\mathcal{J}_1$ ) and guard level objective ( $\mathcal{J}_2$ ) as function of  $\varepsilon$  in  $H_s^u(\varepsilon)$ . Here ‘NF’ and ‘RN’ correspond to the “Noise-Free” and “Risk-Neutral” cases.

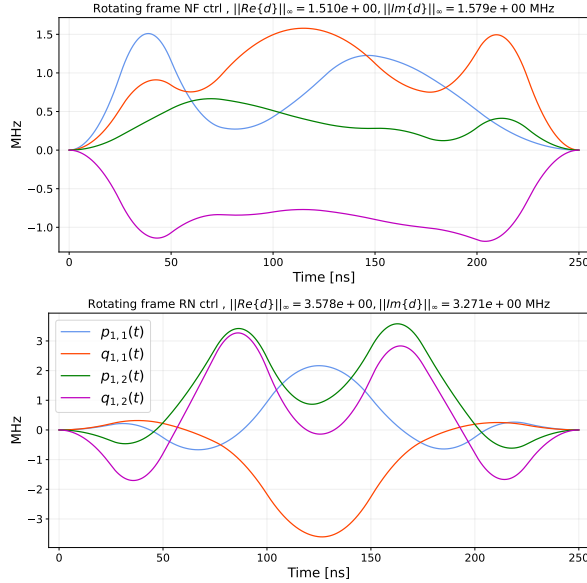


Fig. 5: Control functions (without carrier waves) for the cases: “noise-free” (top), and “risk-neutral” (bottom). Here,  $p_{k,n}(t)$  and  $q_{k,n}(t)$  are defined in (3.2).

optimal control found with the risk-neutral approach is much less sensitive to noise. We plot the control functions for both cases in Figure 5. Note that the risk-neutral controls (bottom panel) have larger amplitudes compared to the noise-free controls (top panel), indicating a potential drawback of the risk-neutral approach. However, a more systematic study of this issue is needed and left for future work.

**7. Code comparison.** Here we compare the proposed method, as implemented in the Jupyter package [13], with two Python implementations of the GRAPE algorithm: the pulse\_optim method from the QuTiP [18] framework and the Grape-TF code (TF is short for TensorFlow [2]). The latter code provides an enhanced imple-



mentation of the GRAPE algorithm, as described by Leung et al. [21]. It is callable from QuTiP and shares a similar problem setup with the pulse\_optim function.

As a test problem, we consider optimizing the control functions for a set of SWAP gates on a single qudit that transforms  $|0\rangle \leftrightarrow |d\rangle$ . The corresponding target gate transformation  $V_E$  equals the permutation matrix that swaps columns 0 and  $d$  in a  $(d + 1) \times (d + 1)$  matrix. To evaluate the amount of leakage to higher energy levels, we add one guard level ( $G = 1$ ) and evolve a total of  $N = d + 2$  states in Schrödinger’s equation. Note that the guard level is left unspecified in the target gate transformation. We implement the SWAP gates on a multi-level qudit with transition frequency  $\omega_1/2\pi = 4.8$  GHz and self-Kerr coefficient  $\xi_1/2\pi = 0.22$  GHz. We apply the rotating wave approximation, with detuning  $\Delta_1 = 0$ , and model the dynamics with the system Hamiltonian (2.7). As a realistic model for current superconducting quantum devices, we impose the control amplitude restrictions

$$(7.1) \quad \max_t |d(t; \boldsymbol{\alpha})| \leq c_\infty, \quad \frac{c_\infty}{2\pi} = 9 \text{ MHz},$$

in the rotating frame of reference.

**7.1. Setup of simulation codes.** QuTiP/pulse\_optim can minimize the target gate fidelity,  $\mathcal{J}_1$ , but does not suppress occupation of higher energy states. Thus, it does *not* minimize terms of the type  $\mathcal{J}_2$ . As a proxy for  $\mathcal{J}_2$ , we append one additional energy level to the simulation and measure its occupation as an estimate of leakage to higher energy states. In pulse\_optim, the control functions are discretized on the same grid in time as Schrödinger’s equation and no smoothness conditions are imposed on the control functions.

Grape-TF also discretizes the control functions on the same grid in time as Schrödinger’s equation. It minimizes an objective function that consists of a number of user-configurable parts. In our test, we minimize the gate infidelity ( $\mathcal{J}_1$ ) and the occupation of one guard (forbidden) energy level (similar to  $\mathcal{J}_2$ ). To smooth the control functions in time, the objective function also includes terms to minimize their first and second time derivatives. The gradient of the objective function is calculated using the automatic differentiation (AD) technique, as implemented in the TensorFlow package.

In Juqbox, we trigger the first  $d$  transition frequencies in the system Hamiltonian by using  $d$  carrier waves in the control functions, with frequencies

$$\Omega_{1,k} = (k - 1)(-\xi_1), \quad k = 1, 2, \dots, N_f, \quad N_f = d.$$

For all three codes, a pseudo-random number generator is used to construct the initial guesses for the parameter vector.

The pulse\_optim and Juqbox simulations were run on a Macbook Pro with a 2.6 GHz Intel iCore-7 processor. To utilize the GPU acceleration in TensorFlow, the Grape-TF simulations were run on one node of the Pascal machine at Livermore Computing, where each node has an Intel XEON E5-2695 v4 processor with two NVIDIA P-100 GPUs.

**7.2. Numerical results.** A SWAP gate where the control functions meet the control amplitude bounds (7.1) can only be realized if the gate duration is sufficiently long. Furthermore, the minimum gate duration increases with  $d$ . For each value of  $d$ , we used numerical experiments to determine a duration  $T_d$  such that at least two of the three simulation codes could find a solution with a small gate infidelity. For

Juqbox, we estimated the number of time steps using the technique in Section 4.2 with  $C_P = 80$ . The number of control parameters follow from  $D = 2N_f D_1$ , where  $N_f = d$  is the number of carrier wave frequencies used and  $D_1$  is the number of B-splines per control functions. We use  $D_1 = 10$  for  $d = 3, 4, 5$  and  $D_1 = 20$  for  $d = 6$ . For pulse\_optim and Grape-TF, we calculate the number of time steps based on the shortest transition period, corresponding to the highest transition frequency in the system. We then use 40 time steps per shortest transition period to resolve the control functions. For both GRAPE methods there are 2 control parameters per time step. The main simulation parameters are given in Table 1.

$d$	$T_d$ [ns]	# time steps		# parameters	
		Juqbox	GRAPE	Juqbox	GRAPE
3	140	14,787	4,480	60	8,960
4	215	37,843	7,568	80	15,136
5	265	69,962	11,661	100	23,322
6	425	157,082	22,441	240	44,882

Table 1: Gate duration, number of time steps ( $M$ ) and total number of control parameters ( $D$ ) in the  $|0\rangle \leftrightarrow |d\rangle$  SWAP gate simulations. The number of time steps and control parameters are the same for pulse\_optim and Grape-TensorFlow.

Optimization results for the pulse\_optim, Grape-TF and Juqbox codes are presented in Figure 6. The pulse\_optim code generates piecewise constant control functions that are very noisy and may therefore be hard to realize experimentally. To obtain a realistic estimate of the resulting dynamics, we interpolate the optimized control functions on a grid with 20 times smaller time step and use the `mesolve()` function in QuTiP to calculate the evolution of the system from each initial state. We then evaluate the gate infidelity using the states at the final time. Since the control functions from Grape-TF and Juqbox are significantly smoother, we report the gate fidelities as calculated by those codes.

For the  $|0\rangle \leftrightarrow |3\rangle$ ,  $|0\rangle \leftrightarrow |4\rangle$  and  $|0\rangle \leftrightarrow |5\rangle$  SWAP gates, all three codes produce control functions with very small gate infidelities. The most significant difference between the codes occurs for the  $d = 6$  SWAP gate. Here, the Grape-TF code fails to produce a small gate infidelity after running for almost 23 hours and the pulse\_optim code results in a gate fidelity that is about 2 orders of magnitude larger than Juqbox. While pulse\_optim and Juqbox require comparable amounts of CPU time to converge, the Grape-TF code is between 50-100 times slower despite the GPU acceleration.

To compare the smoothness of the optimized control functions, we study the Fourier spectra of the laboratory frame control functions, see Figure 7. Note that pulse\_optim produces a significantly noisier control function compared to the other two codes. The control function from Grape-TF is significantly smoother, even though its spectrum includes some noticeable peaks at frequencies that do not correspond to transition frequencies in the system. The Juqbox simulation results in a laboratory frame control function where each peak in the spectrum corresponds to a transition frequency in the Hamiltonian.

**8. Conclusions.** In this paper we developed numerical methods for optimizing control functions for realizing logical gates in closed quantum systems where the state

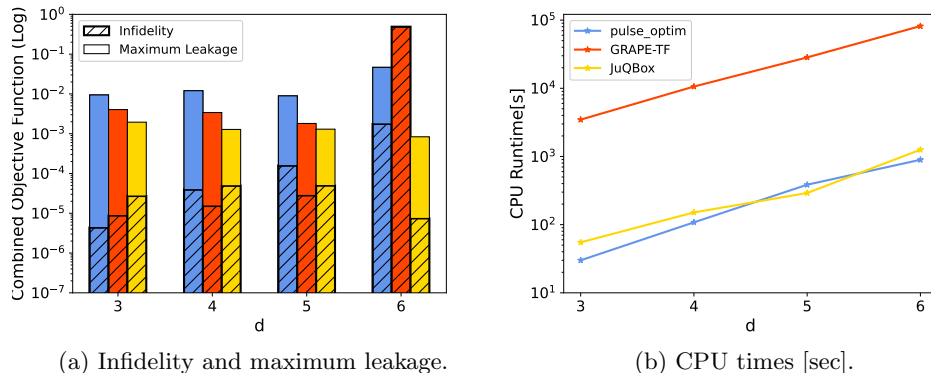


Fig. 6: Optimal control results for SWAP  $|0\rangle \leftrightarrow |d\rangle$  gates using pulse\_optim (blue), Grape-TensorFlow (red) and Juqbox (yellow). (a) Gate infidelity and maximum occupation of the forbidden state  $|d+1\rangle$ ; (b) CPU timings.

is governed by Schrödinger’s equation. By asymptotic expansion, we calculated the resonant frequencies in the system Hamiltonian, corresponding to transitions between energy levels in the state vector. We introduced a novel parameterization of the control functions using B-spline basis functions that act as envelopes for carrier waves, with frequencies that match the transition frequencies. This approach allows the number of control parameters to be independent of, and significantly smaller than, the number of time steps for integrating Schrödinger’s equation.

The objective function in the optimal control problem consists of two parts: the infidelity of the final gate transformation and a time-integral for evaluating leakage to higher energy levels. We apply a “first-discretize-then-optimize” approach and outline the derivation of the discrete adjoint equation that is solved to efficiently calculate the gradient of the objective function.

To demonstrate our approach, we optimized the control functions for a CNOT gate on two coupled qudits, resulting in a gate fidelity exceeding 99.9%. Based on a simple noise model, we also generalized the proposed method to calculate risk-neutral controls that are resilient to uncertainties in the Hamiltonian model. The results are promising and indicate that a more systematic study of optimization under uncertainty can yield controls that are robust to noise in quantum systems. We finally compared the performance of the proposed method, implemented in the Juqbox package [13], and two implementations of the GRAPE algorithm: the pulse\_optim method in QuTiP [18] and Grape-TensorFlow [21]. The codes were compared on a set of SWAP gates on a single qudit. Here, Juqbox was found to run 50-100 times faster than Grape-TensorFlow and produce control functions that are significantly smoother than pulse\_optim. The most significant difference between the codes occurred for the  $|0\rangle \leftrightarrow |6\rangle$  SWAP gate, where the Grape-TF code failed to solve the optimal control problem.

In future work, we intend to generalize our approach to solve optimal control problems for larger quantum systems.

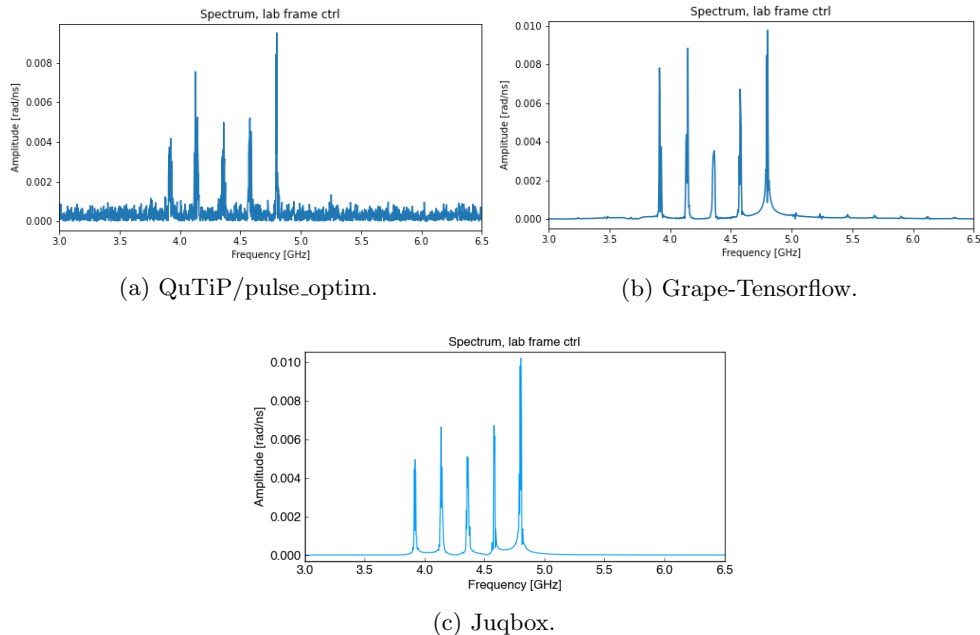


Fig. 7: Magnitude of the Fourier spectrum of the laboratory frame control function for the  $|0\rangle \leftrightarrow |5\rangle$  SWAP gate on a single qudit.

**Acknowledgment.** We would like to thank Prof. Daniel Appelö for bringing the Störmer-Verlet method to our attention. We also thank Dr. Björn Sjögren for fruitful discussions on resonant frequencies, as well as two anonymous referees who helped improve the presentation.

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344. This is contribution LLNL-JRNL-823853. The views and opinions of the authors do not necessarily reflect those of the U.S. government or Lawrence Livermore National Security, LLC neither of whom nor any of their employees make any endorsements, express or implied warranties or representations or assume any legal liability or responsibility for the accuracy, completeness, or usefulness of the information contained herein.

#### Appendix A. Composite quantum systems and essential states.

To simplify the notation we assume a bipartite quantum system ( $Q = 2$ ); the case  $Q = 1$  is trivial and  $Q > 2$  follows by straightforward generalizations. Let the number of energy levels in the subsystems be  $n_1$  and  $n_2$ , respectively, for a total of  $N = n_1 \cdot n_2$  states in the coupled system. We use the canonical unit vectors  $\mathbf{e}_j^{(n_q)} \in \mathbb{R}^{n_q}$ , for  $j = 0, \dots, n_q - 1$ , as a basis for subsystem  $q$ , where the superscript indicates its size. These basis vectors can be used to describe the total state of the coupled system,

$$(A.1) \quad \psi = \sum_{j_2=0}^{n_2-1} \sum_{j_1=0}^{n_1-1} \psi_{j_2, j_1} \left( \mathbf{e}_{j_2}^{(n_2)} \otimes \mathbf{e}_{j_1}^{(n_1)} \right) = \sum_{k=0}^{N-1} \vec{\psi}_k \mathbf{e}_k^{(N)}.$$

Here,  $\vec{\psi} \in \mathbb{C}^N$  denotes the one-dimensional representation of the two-dimensional state vector  $\boldsymbol{\psi}$ , using a natural ordering of the elements, i.e.,  $\vec{\psi}_k = \psi_{j_2, j_1}$  for  $k = j_1 + n_1 j_2 =: k_{ind}(j_2, j_1)$ . The mapping  $k = k_{ind}(j_2, j_1)$  is invertible for  $k \in [0, N - 1]$ .

We classify the energy levels in the total state vector as either essential or guarded levels. The unitary gate transformation is only specified for the essential levels. The guard levels are retained to justify the truncation of the modal expansion of Schrödinger's equation, and to avoid leakage of probability to even higher energy levels.

Let the number of essential energy levels in the subsystems be  $m_1$  and  $m_2$ , respectively, where  $0 < m_q \leq n_q$ . Similar to the total state vector, we use the canonical unit vectors as a basis for the essential subspace of each subsystem. The total number of essential levels equals  $E = m_1 \cdot m_2$ . Let the essential energy levels in the total state vector be represented by the essential state vector  $\boldsymbol{\phi}$ . Similar to the full state vector, we flatten its two-dimensional indexing using a natural ordering,

$$(A.2) \quad \boldsymbol{\phi} = \sum_{i_2=0}^{m_2-1} \sum_{i_1=0}^{m_1-1} \phi_{i_2, i_1} \left( \mathbf{e}_{i_2}^{(m_2)} \otimes \mathbf{e}_{i_1}^{(m_1)} \right) = \sum_{\ell=0}^{E-1} \vec{\phi}_\ell \mathbf{e}_\ell^{(E)} \in \mathbb{C}^E,$$

where  $\ell = i_1 + m_1 i_2 =: \ell_{ind}(i_2, i_1)$ . The elements in the essential state vector are defined from the total state vector by  $\phi_{i_2, i_1} = \psi_{i_2, i_1}$ , for  $i_1 \in [0, m_1 - 1]$  and  $i_2 \in [0, m_2 - 1]$ .

The initial condition for the solution operator matrix  $U(t)$  in Schrödinger's equation (1.3) needs to span a basis for the  $E$ -dimensional essential state space. Here we use the canonical basis consisting of the unit vectors  $\mathbf{e}_\ell^{(E)}$ . Let the columns of the initial condition matrix be  $U_0 = [\mathbf{g}_0, \mathbf{g}_1, \dots, \mathbf{g}_{E-1}] \in \mathbb{R}^{N \times E}$ . Because the total probabilities in each column vector  $\mathbf{g}_k$  must sum to one, the basis vectors in the total state space become

$$(A.3) \quad \mathbf{g}_\ell = U_0 \mathbf{e}_\ell^{(E)}, \quad g_{k, \ell} = \begin{cases} 1, & k = k_{ind}(i_2(\ell), i_1(\ell)), \\ 0, & \text{otherwise,} \end{cases} \quad \text{for } \ell = 0, 1, \dots, E - 1.$$

Here,  $i_2(\ell) = \lfloor \ell / m_1 \rfloor$  and  $i_1(\ell) = \ell - m_1 \cdot i_2(\ell)$ .

The target gate matrix  $V_E \in \mathbb{C}^{E \times E}$  defines the unitary transformation between the essential levels in the initial and final states,  $\boldsymbol{\phi}_T = V_E \boldsymbol{\phi}_0$ , for all  $\boldsymbol{\phi}_0 \in \mathbb{C}^E$ . Because  $V_E$  is unitary, each of its columns has norm one. To preserve total probabilities, we define the target gate transformation according to

$$(A.4) \quad V_{tg} = U_0 V_E \in \mathbb{C}^{N \times E}.$$

This implies that each column of  $V_{tg}$  also has norm one.

### Appendix B. The Hamiltonian in a rotating frame of reference.

The time-dependent and unitary change of variables  $\tilde{\boldsymbol{\psi}}(t) = R(t)\boldsymbol{\psi}(t)$  where  $R^\dagger R = I$ , results in the transformed Schrödinger equation

$$(B.1) \quad \frac{d\tilde{\boldsymbol{\psi}}}{dt} = -i\tilde{H}(t)\tilde{\boldsymbol{\psi}}, \quad \tilde{H}(t) = R(t)H(t)R^\dagger(t) + i\dot{R}(t)R^\dagger(t).$$

The rotating frame of reference is introduced by taking the unitary transformation to be the matrix (2.6). Because both  $R(t)$  and the system Hamiltonian (2.2) are

diagonal,  $RH_sR^\dagger = H_s$ . The time derivative of the transformation can be written

$$(B.2) \quad \dot{R}(t) = \left( \bigoplus_{q=Q}^1 i\omega_{r,q} A_q^\dagger A_q \right) \left( \bigotimes_{q=Q}^1 \exp(i\omega_{r,q} t A_q^\dagger A_q) \right),$$

where  $\oplus$  denotes the Kronecker sum,  $C \oplus D = C \otimes I_D + I_C \otimes D$ . Therefore,

$$(B.3) \quad \dot{R}(t)R^\dagger(t) = \bigoplus_{q=Q}^1 i\omega_{r,q} A_q^\dagger A_q = \sum_{q=1}^Q i\omega_{r,q} a_q^\dagger a_q.$$

As a result, the first term in the Hamiltonian (2.2) is modified by the term  $i\dot{R}(t)R^\dagger(t)$ . After noting that  $Ra_q = e^{-i\omega_{r,q}t}a_qR$ , the transformed Hamiltonian can be written as

$$(B.4) \quad H_s^{rw} = \sum_{q=1}^Q \left( \Delta_q a_q^\dagger a_q - \frac{\xi_q}{2} a_q^\dagger a_q^\dagger a_q a_q - \sum_{p>q} \xi_{qp} a_q^\dagger a_q a_p^\dagger a_p \right),$$

$$(B.5) \quad \tilde{H}_c(t) = \sum_{q=1}^Q f_q(t; \boldsymbol{\alpha}) (e^{-i\omega_{r,q}t} a_q + e^{i\omega_{r,q}t} a_q^\dagger),$$

where  $\Delta_q = \omega_q - \omega_{r,q}$  is the detuning frequency. The above system Hamiltonian corresponds to (2.7).

To slow down the time scales in the control Hamiltonian, we want to absorb the highly oscillatory factors  $\exp(\pm i\omega_{r,q}t)$  into  $f_q(t)$ . Because the control function  $f_q(t)$  is real-valued, this can only be done in an approximate fashion. We make the ansatz,

$$(B.6) \quad f_q(t) := 2 \operatorname{Re} (d_q(t) e^{i\omega_{r,q}t}) = d_q(t) e^{i\omega_{r,q}t} + \bar{d}_q(t) e^{-i\omega_{r,q}t},$$

where  $\bar{d}_q$  denotes the complex conjugate of  $d_q$ . By substituting this expression into the transformed control Hamiltonian (B.5), we get

$$\tilde{H}_c(t) = \sum_{q=1}^Q (d_q(t) a_q + \bar{d}_q(t) a_q^\dagger + \bar{d}_q(t) e^{-2i\omega_{r,q}t} a_q + d_q(t) e^{2i\omega_{r,q}t} a_q^\dagger).$$

The rotating wave approximation (RWA) follows by ignoring terms that oscillate with frequency,  $\pm 2i\omega_{r,q}$ , resulting in the approximate control Hamiltonian (2.8).

### Appendix C. Conditions for resonance.

Consider the scalar function  $y(t) := \psi_{\mathbf{j}}^{(1)}(t)$ . It satisfies an ordinary differential equation of the form

$$(C.1) \quad \frac{dy(t)}{dt} + \kappa_{\mathbf{j}} y(t) = \sum_{\ell} c_{\ell} e^{i\nu_{\ell} t}, \quad \nu_k \in \mathbb{R}.$$

We are interested in cases when  $y(t)$  grows in time, corresponding to resonance. Conditions for resonance are provided in the following lemma.

LEMMA C.1. *Let  $\kappa \in \mathbb{R}$  and  $\nu \in \mathbb{R}$  be constants. The solution of the scalar ordinary differential equation*

$$(C.2) \quad \frac{dy(t)}{dt} + i\kappa y(t) = c e^{i\nu t}, \quad y(0) = y_0,$$

is given by

$$(C.3) \quad y(t) = \begin{cases} y_0 e^{-i\kappa t} + c t e^{-i\kappa t}, & \nu + \kappa = 0, \\ y_0 e^{-i\kappa t} - \frac{ic}{\nu + \kappa} (e^{i\nu t} - e^{-i\kappa t}), & \text{otherwise.} \end{cases}$$

Corresponding to resonance, the function  $y(t)$  grows linearly in time when  $\nu + \kappa = 0$  and  $c \neq 0$ .

*Proof.* Follows by direct evaluation.  $\square$

We proceed by analyzing the right hand side of (2.12). It can be shown that the forcing function  $\mathbf{f}^{(k)}(t)$  is of the form

$$(C.4) \quad \mathbf{f}_j^{(k)}(t) = \begin{cases} -ig_{j+e_k} \sqrt{j_k + 1} e^{i(\Omega_k - \kappa_{j+e_k})t}, & j_k = 0, \\ \Theta_k(t), & j_k \in [1, n_k - 2], \\ -ig_{j-e_k} \sqrt{j_k} e^{-i(\Omega_k + \kappa_{j-e_k})t}, & j_k = n_k - 1, \end{cases}$$

where  $\Theta_k(t) = -ig_{j+e_k} \sqrt{j_k + 1} e^{i(\Omega_k - \kappa_{j+e_k})t} - ig_{j-e_k} \sqrt{j_k} e^{-i(\Omega_k + \kappa_{j-e_k})t}$ .

The right hand side satisfies  $\mathbf{f}(t) = \mathbf{f}^{(1)}(t) + \mathbf{f}^{(2)}(t)$ . The first set of frequencies and coefficients on the right hand side of (C.1) satisfy

$$\nu_1 = \Omega_k - \kappa_{j+e_k}, \quad c_1 = -ig_{j+e_k} \sqrt{j_k + 1},$$

for  $k = \{1, 2\}$  and  $j_k \in [0, n_k - 2]$ . The second set of frequencies and coefficients are

$$\nu_2 = -(\Omega_k + \kappa_{j-e_k}), \quad c_2 = -ig_{j-e_k} \sqrt{j_k}.$$

for  $k = \{1, 2\}$  and  $j_k \in [1, n_k - 1]$ . From Lemma C.1, component  $\psi_j^{(1)}(t)$  is in resonance if  $(\kappa_j + \nu_1 = 0, c_1 \neq 0)$  or  $(\kappa_j + \nu_2 = 0, c_2 \neq 0)$ . These conditions are equivalent to (2.14) and (2.15), which proves Lemma 2.1.

#### Appendix D. Non-diagonal system Hamiltonians.

Current superconducting quantum computing hardware consists of qubits coupled by high quality resonators. Here we consider a transmon (artificial atom) qubit in which a Josephson junction is coupled in parallel to a capacitor. This section provides a brief overview of the derivation of the Hamiltonian model in the dispersive limit. For further details, see for example the review paper by Blais et al. [4]. The interaction between the transmon qubit and the resonator can be modeled by the Hamiltonian [4],

$$(D.1) \quad H_{qr} = \omega_r a^\dagger a + \omega_q b^\dagger b - \frac{E_C}{2} b^\dagger b^\dagger b b + g(b^\dagger - b)(a^\dagger - a),$$

where  $\omega_r$  is the resonant frequency of the resonator,  $\omega_q$  is the transmon frequency,  $E_C$  is the charging energy, and  $g$  is the resonator-transmon coupling constant. Further,  $a$  and  $b$  are the respective lowering operators of the resonator and the transmon. The Hamiltonian can be simplified when the coupling constant is small compared to the resonant and transmon frequencies ( $|g| \ll \omega_r, |g| \ll \omega_q$ ), resulting in

$$(D.2) \quad H_{qr} \approx H_{lin} + H_{nl}, \quad H_{lin} = \omega_r a^\dagger a + \omega_q b^\dagger b - \frac{E_C}{2} b^\dagger b^\dagger b b, \quad H_{nl} = g(b^\dagger a + b a^\dagger).$$

Let  $\Delta = \omega_q - \omega_r$  be the qubit-resonator detuning frequency and define  $\lambda = g/\Delta$ . In the dispersive limit, when  $|\lambda| = |g/\Delta| \ll 1$ , the qubit and resonator are only

weakly entangled and the Hamiltonian can be diagonalized by a Schrieffer-Wolff [7, 32] expansion, or by using the Bogoliubov transformation [5]. In the latter case, we take the unitary transformation to be  $U_{disp} = \exp(\Lambda(a^\dagger b - ab^\dagger))$ . Under this transformation, the lowering operators transform as  $U_{disp}^\dagger a U_{disp} = \cos(\Lambda)a + \sin(\Lambda)b$  and  $U_{disp}^\dagger b U_{disp} = \cos(\Lambda)b - \sin(\Lambda)a$ . By choosing  $\tan(2\Lambda) = 2\lambda$ , the linear part of the Hamiltonian transforms to diagonal form,

$$(D.3) \quad U_{disp}^\dagger H_{lin} U_{disp} = \tilde{\omega}_r a^\dagger a + \tilde{\omega}_q b^\dagger b,$$

where the so-called ‘‘dressed’’ frequencies are  $\tilde{\omega}_r = 0.5(\omega_r + \omega_q - \sqrt{\Delta^2 + 4g^2})$  and  $\tilde{\omega}_q = 0.5(\omega_r + \omega_q + \sqrt{\Delta^2 + 4g^2})$ . After an additional Schrieffer-Wolff expansion in  $|\lambda| \ll 1$ , the non-linear part of the Hamiltonian can also be diagonalized, resulting in the dispersive Hamiltonian

$$(D.4) \quad U_{disp}^\dagger (H_{lin} + H_{nl}) U_{disp} \approx \tilde{\omega}_r a^\dagger a + \tilde{\omega}_q b^\dagger b + \frac{K_a}{2} a^\dagger a^\dagger a a + \frac{K_b}{2} b^\dagger b^\dagger b b + \chi_{ab} a^\dagger a b^\dagger b =: H_{disp}.$$

In the dispersive regime, the coefficients  $K_a$ ,  $K_b$  and  $\chi_{ab}$  are all negative. The dispersive Hamiltonian is diagonal. This approach can be generalized to several transmons and resonators [4], resulting in the model (2.2).

#### REFERENCES

- [1] *IBM Quantum*, 2021. <https://quantum-computing.ibm.com>, accessed 2022-03-15.
- [2] M. ABADI, P. BARHAM, J. CHEN, Z. CHEN, A. DAVIS, J. DEAN, M. DEVIN, S. GHEMAWAT, G. IRVING, M. ISARD, ET AL., *Tensorflow: A system for large-scale machine learning*, in 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16), 2016, pp. 265–283.
- [3] J. BEZANSON, A. EDELMAN, S. KARPINSKI, AND V. B. SHAH, *Julia: A fresh approach to numerical computing*, SIAM Review, 59 (2017), pp. 65–89.
- [4] A. BLAIS, A. L. GRIMSMO, S. M. GIRVIN, AND A. WALLRAFF, *Circuit quantum electrodynamics*, Rev. Mod. Phys., 93 (2021), p. 025005.
- [5] M. BOISSONNEAULT, J. M. GAMBETTA, AND A. BLAIS, *Dispersive regime of circuit QED: Photon-dependent qubit dephasing and relaxation rates*, Phys. Rev. A, 79 (2009), p. 013819.
- [6] A. BORZÌ, G. CIARMELLA, AND M. SPRENGEL, *Formulation and Numerical Solution of Quantum Control Problems*, Computational science and engineering, SIAM, 2017.
- [7] S. BRAVYI, D. P. DIVINCENZO, AND D. LOSS, *Schrieffer–Wolff transformation for quantum many-body systems*, Annals of Physics, 326 (2011), p. 2793–2826.
- [8] T. CANEVA, T. CALARCO, AND S. MONTANGERO, *Chopped random-basis quantum optimization*, Physical Review A, 84 (2011).
- [9] C. CHEN, D. DONG, R. LONG, I. R. PETERSEN, AND H. A. RABITZ, *Sampling-based learning control of inhomogeneous quantum ensembles*, Physical Review A, 89 (2014), p. 023402.
- [10] J. E. COMBARIZA, B. JUST, J. MANZ, AND G. K. PARAMONOV, *Isomerizations controlled by ultrashort infrared laser pulses: model simulations for the inversion of ligands (h) in the double-well potential of an organometallic compound, [(c5h5)(co)2feph2]*, The Journal of Physical Chemistry, 95 (1991), pp. 10351–10359.
- [11] L. EMSLEY AND G. BODENHAUSEN, *Gaussian pulse cascades: New analytical functions for rectangular selective inversion and in-phase excitation in NMR*, Chem. Phys., 165 (1989), pp. 469–476.
- [12] B. EWING, S. J. GLASER, AND G. P. DROBNY, *Development and optimization of shaped NMR pulses for the study of coupled spin systems*, Chem. Phys., 147 (1990), pp. 121–129.
- [13] F. GARCIA AND N. A. PETERSSON, *Jugbox.jl*. Github, 2021.
- [14] X. GE, H. DING, H. RABITZ, AND R.-B. WU, *Robust quantum control in games: An adversarial learning approach*, Physical Review A, 101 (2020), p. 052317.
- [15] X. GE AND R.-B. WU, *Risk-sensitive optimization for robust quantum controls*, Phys. Rev. A, 104 (2021), p. 012422.



- [16] G. H. GOLUB AND C. F. V. LOAN, *Matrix Computations*, Johns Hopkins University Press, 1996.
- [17] E. HAIRER, C. LUBICH, AND G. WANNER, *Geometric Numerical Integration*, no. 31 in Springer series in computational mathematics, Springer-Verlag, Heidelberg, 2nd ed., 2006.
- [18] J. JOHANSSON, P. NATION, AND F. NORI, *QuTiP 2: A Python framework for the dynamics of open quantum systems*, Computer Physics Communications, 184 (2013), pp. 1234 – 1240.
- [19] N. KHANEJA, T. REISS, C. KEHLET, T. SCHULTE-HERBRUGGEN, AND S. GLASER, *Optimal control of coupled spin dynamics: design of NMR pulse sequences by gradient ascent algorithms*, J. Magnetic Resonance, 172 (2005), pp. 296–305.
- [20] H. W. KUHN AND A. W. TUCKER, *Nonlinear programming*, in Traces and emergence of nonlinear programming, Springer, 2014, pp. 247–258.
- [21] N. LEUNG, M. ABDELHAFEZ, J. KOCH, AND D. SCHUSTER, *Speedup for quantum optimal control from automatic differentiation based on graphics processing units*, Phys. Rev. A, 95 (2017), p. 0432318.
- [22] J.-S. LI AND N. KHANEJA, *Control of inhomogeneous quantum ensembles*, Phys. Rev. A, 73 (2006), p. 030302.
- [23] ———, *Ensemble control of Bloch equations*, IEEE Transactions on Automatic Control, 54 (2009), pp. 528–536.
- [24] D. LUCARELLI, *Quantum optimal control via gradient ascent in function space and the time-bandwidth quantum speed limit*, Physical Review A, 97 (2018).
- [25] S. MACHNES, E. ASSÉMAT, D. TANNOR, AND F. K. WILHELM, *Tunable, flexible, and efficient optimization of control pulses for practical qubits*, Physical Review Letters, 120 (2018).
- [26] E. MAGESAN AND J. M. GAMBETTA, *Effective Hamiltonian models of the cross-resonance gate*, Phys. Rev. A, 101 (2020), p. 052308.
- [27] M. NIELSEN AND I. CHUANG, *Quantum computation and quantum information*, Cambridge University Press, 2000.
- [28] J. NOCEDAL AND S. J. WRIGHT, *Numerical Optimization*, Springer, 2nd ed., 2006.
- [29] M. S. P BRUMER, *Laser control of molecular processes*, Annual review of physical chemistry, 43 (1992), pp. 257–282.
- [30] J. P. PALAO, R. KOSLOFF, AND C. P. KOCH, *Protecting coherence in optimal control theory: State-dependent constraint approach*, Phys. Rev. A, 77 (2008), p. 063412.
- [31] N. A. PETERSSON, F. M. GARCIA, A. E. COPELAND, Y. L. RYDIN, AND J. L. DUBOIS, *Discrete adjoints for accurate numerical optimization with application to quantum control*, arXiv preprint arXiv:2001.01013, (2020).
- [32] J. R. SCHRIEFFER AND P. A. WOLFF, *Relation between the Anderson and Kondo Hamiltonians*, Phys. Rev., 149 (1966), pp. 491–492.
- [33] S. SHI AND H. RABITZ, *Optimal control of selectivity of unimolecular reactions via an excited electronic state with designed lasers*, The Journal of Chemical Physics, 97 (1992), pp. 276–287.
- [34] Y. SHI, N. LEUNG, P. GOKHALE, Z. ROSSI, D. I. SCHUSTER, H. HOFFMANN, AND F. T. CHONG, *Optimized compilation of aggregated instructions for realistic quantum computers*, Proceedings of the Twenty-Fourth International Conference on Architectural Support for Programming Languages and Operating Systems - ASPLOS '19, (2019).
- [35] P. SIERANT, M. LEWENSTEIN, AND J. ZAKRZEWSKI, *Polynomially filtered exact diagonalization approach to many-body localization*, Physical Review Letters, 125 (2020).
- [36] M. UNSER, *Ten good reasons for using spline wavelets*, Wavelet Applications in Signal and Image Processing V, 3169 (2002).
- [37] A. WÄCHTER AND L. T. BIEGLER, *On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming*, Mathematical Programming, 106 (2006), pp. 25–57.
- [38] R.-B. WU, H. DING, D. DONG, AND X. WANG, *Learning robust and high-precision quantum controls*, Phys. Rev. A, 99 (2019), p. 042327.
- [39] D. B. ZAX, G. GOELMAN, AND S. VEGA, *Amplitude-modulated composite pulses*, J. Magn. Reson., 80 (1988), pp. 375–382.