


Stochastic learning control of inhomogeneous quantum ensembles

Gabriel Turinici *

CEREMADE, Université Paris Dauphine, Paris Sciences et Lettres Research University,
Place du Maréchal De Lattre De Tassigny, 75775 Paris, France



(Received 21 May 2019; published 8 November 2019)

In quantum control, the robustness with respect to uncertainties in the system's parameters or driving-field characteristics is of paramount importance and has been studied theoretically, numerically, and experimentally. We test in this paper stochastic search procedures (Stochastic gradient descent and the Adam algorithm) that sample, at each iteration, from the distribution of the parameter uncertainty, as opposed to previous approaches that used a fixed grid. We show that both algorithms behave well with respect to benchmarks and discuss their relative merits. In addition the methodology allows to address high-dimensional parameter uncertainty; we implement numerically, with good results, a three-dimensional and a six-dimensional case.

DOI: [10.1103/PhysRevA.100.053403](https://doi.org/10.1103/PhysRevA.100.053403)

I. INTRODUCTION

Quantum control is a promising technology with many applications ranging from nuclear magnetic resonance (NMR) [1] to quantum computing [2] and laser control of quantum dynamics [3]. The controlling field encounters many molecules which, although identical in nature, may interact differently with the incoming field because of different Larmor frequencies or rf attenuation factors (in NMR spin control or quantum computing, see [4–9]), different spatial profile (see [10]), or other parameters (see [11–13]). For obvious practical reasons, it is of paramount importance to ensure that the control quality is robust with respect to this heterogeneity. Thus the quantum control problem involves a unique set of driving fields $u(t) \in \mathbb{R}^L$, the same for all molecules in the ensemble, however, each molecule is described by a set of parameters $\theta \in \Theta \subset \mathbb{R}^d$ and the control outcome depends on both u and θ ; the goal can be expressed as the maximization of the control quality averaged over θ . A different view is when the variability is not due to the presence of many different molecules but when uncertainties in the control implementation require to devise a field robust to fluctuations in those parameters.

A first natural question is whether this is at all possible, i.e., if a single field can drive several distinct molecules to a common target; the answer is given by the theory of ensemble control controllability, see [4,14–17] and is, in general, positive. However, the theory does not explain how to find the control (except under specific regimes, see [18]). To do so, different algorithms have been proposed: the pseudospectral approach of Li *et al.* [19–21] consider spectral and/or polynomial representations of the control problem in two dimensions ($d = 2$); Wang considered iterative procedures based on sampling [22]; the learning approach of Chen *et al.* [12] and Kuang *et al.* [23]

(the latter in the context of time-optimal control) considered a fixed uniform grid over the inhomogeneous parameter space and was tested for $d = 2$. Finally, Wu *et al.* [24] found robust controls using uniform grids in two and three dimensions ($d = 3$).

In all these works there is always a fixed grid (or fixed sampling) involved when the control is searched. The rationale behind this idea is that a fixed grid makes the search more stable and a good choice of the grid is enough to describe efficiently the mean performance of the control over the parameter space in the spirit of a quadrature formula for the average over θ . This is coherent with results from the approximation theory which inform that convergence is of order $e^{-\sqrt[4]{N}}$, with respect to the number N of grid points; however, the same formula indicates a bad scaling with respect to d . To address this *curse of dimensionality* and also explore the nature of the search landscape, we take here a different view: At each control iteration we use a new sampling in the spirit of Monte Carlo methods (see [25, Sec. 7.7]) for computing high-dimensional integrals. This will induce slight oscillations in the average but has the advantage to cover the space Θ of inhomogeneity even in high dimensions d . A similar approach was tested independently in a very recent work by Wu *et al.* [26] for a two-dimensional example and promising results were obtained; see Sec. II B for comments on the differences between the two approaches. The procedure we propose is detailed in the next section and the numerical results are the object of Sec. III.

II. ALGORITHMS FOR ENSEMBLE QUANTUM CONTROL

We consider a control $u(t) = (u_1(t), \dots, u_L(t)) \in \mathbb{R}^L$ acting on a molecule part of a larger ensemble. Each molecule is completely characterized by some inhomogeneity parameter $\theta \in \Theta \subset \mathbb{R}^d$ obeying a distribution law $P(\theta)$ on Θ (which can be the uniform distribution or any other). All molecules are subjected to the same control $u(t)$ during the time interval $[0, T]$ to reach some target.

*gabriel.turinici@dauphine.fr; www.ceremade.dauphine.fr/~turinici; IUF - Institut Universitaire de France.

A. Evolution equations

The dynamics of each molecule in the sample is governed by the Hamiltonian $H(\theta, u) = H_0(\theta) + \sum_{\ell=1}^L u_\ell(t)H_\ell(\theta)$ through the Schrödinger equation

$$i \frac{d}{dt} \psi(t; \theta) = H(\theta, u) \psi(t; \theta), \quad (1)$$

where ψ is the wave function of the molecule (here and below we set $\hbar = 1$). Of course, ψ depends on u but for notational convenience we omit to write explicitly this dependence from now on. Once a finite-dimensional basis $\{|j\rangle, j = 1, \dots, N\}$ is chosen, the state of the quantum system can be represented as

$$|\psi(t; \theta)\rangle = \sum_{j=1}^N c_j(t; \theta) |j\rangle. \quad (2)$$

Denoting $C(t; \theta) = [c_0(t; \theta), \dots, c_N(t; \theta)]^T$ the vector of coefficients C satisfies the equation

$$\frac{d}{dt} C(t; \theta) = X(\theta, u) C(t; \theta), \quad (3)$$

where X is the representation of the Hamiltonian H (including the $1/i$ factor) in the basis $|j\rangle, j = 1, \dots, N$.

Note that same setting also applies to nonlinear Hamiltonians, e.g., Bose-Einstein condensates (nonlinearity in ψ), or high-order control terms [27,28] (nonlinearity in u).

The quantum system can also be described in terms of a density matrix $\rho(t; \theta)$; this matrix is expressed in some basis for operators. The same happens when the molecule is coupled to a bath or when relaxation phenomena are at work, see [29]; in both cases the coefficients of this expansion follow an equation similar to Eq. (3).

B. Optimization by stochastic gradient descent and Adam algorithms

The control goal is encoded as the minimization, with respect to u , of an error, or “loss” functional $\mathcal{L}(u, \theta)$ depending on the control u and the Hamiltonian parameters θ . When all the ensemble is considered, the following loss functional is to be minimized:

$$\mathcal{J}(u) = \int_{\Theta} \mathcal{L}(u, \theta) P(d\theta). \quad (4)$$

The stochastic optimization algorithms described below construct an iterative process to find the u that minimizes Eq. (4).

Historically the first to be considered, the stochastic gradient descent algorithm [30] (henceforth called SGD) consists of the following procedure:

Algorithm 1 SGD

- 1: Choose a learning rate $\alpha > 0$, a minibatch size $M > 0$ and the initial control u^0 .
 - 2: Set iteration counter $k = 0$.
 - 3: **repeat**
 - 4: Draw M independent parameters $\theta_1^k, \dots, \theta_M^k$ from the distribution $P(\theta)$ and compute the approximation $g^k := \frac{1}{M} \sum_{m=1}^M \nabla_u \mathcal{L}(u^k; \theta_m^k)$ of the gradient $\nabla_u \mathcal{J}(u^k)$ of $\mathcal{J}(\cdot)$ at u^k .
 - 5: Set $u^{k+1} = u^k - \alpha g^k$ and $k = k + 1$.
 - 6: **until** stopping criterion is satisfied.
-

To accelerate the convergence of the SGD algorithm, several improvements have been proposed (see [31]) among with the Adam [32] variant which proved to be one of the most efficient and very scalable. The difference between Adam and SGD is that Adam uses a different learning rate for each parameter which is tuned as follows: When the uncertainty in the gradient is large the learning rate is taken to be small and contrary otherwise. To have a robust estimation for the gradient (in absolute value) an exponential moving average is computed on the fly (see below). It can be described as follows:

Algorithm 2 Adam

- 1: Choose the learning rate $\alpha > 0$, the EMA parameters β_1 and β_2 , the minibatch size $M > 0$, the epsilon $\varepsilon > 0$, and the initial control u^0 .
 - 2: Set iteration counter $k = 0$, first moment estimate $\mu = 0$, second moment estimate $v = 0$.
 - 3: Set $k = k + 1$.
 - 4: **repeat**
 - 5: Draw M independent parameters $\theta_1^k, \dots, \theta_M^k$ from the distribution $P(\theta)$ and compute the approximation $g^k := \frac{1}{M} \sum_{m=1}^M \nabla_u \mathcal{L}(u^{k-1}; \theta_m^k)$ of the gradient $\nabla_u \mathcal{J}(u^{k-1})$ of $\mathcal{J}(\cdot)$ at u^{k-1} .
 - 6: Compute the moving averages $\hat{\mu}^k := \beta_1 \mu^{k-1} + (1 - \beta_1) g^k$, $\hat{v}^k := \beta_2 v^{k-1} + (1 - \beta_2) |g^k|^2$.
 - 7: Compute bias-corrected moment estimates: $\hat{\mu}^k = \hat{\mu}^k / [1 - (\beta_1)^k]$, $\hat{v}^k = \hat{v}^k / [1 - (\beta_2)^k]$.
 - 8: set $u^k = u^{k-1} - \alpha \hat{\mu}^k / (\sqrt{\hat{v}^k} + \varepsilon)$.
 - 9: **until** some stopping criterion is satisfied.
-

The momentum algorithm used in [26] can be seen as being halfway between SGD and Adam; it is formally a special case of the Adam algorithm for $\beta_1 = \lambda$, $\beta_2 = 1$, $v^0 = 1$ and no bias correction step 7 (that is $\hat{\mu}^k = \mu^k$, $\hat{v}^k = v^k$). In practice, the numerical results are very similar and point in the same direction; in particular, we expect that the momentum algorithm is also relevant to high-dimensional robust control problems.

III. NUMERICAL RESULTS

We test the performance of the algorithms in Sec. II B for several benchmarks from the literature (or that generalize cases from the literature).

In Secs. III A and III A we compare the SGD algorithm with a fixed grid sampling method from the literature. Then in Secs. III C and III D we compare the SGD with the Adam algorithm and in Sec. III E we draw further conclusions concerning stochastic optimization.

In the situations considered below, the goal is to maximize the so-called *fidelity* denoted $\mathcal{F}(u; \theta)$. For Secs. III A and III B this has the formula $\mathcal{F}(u; \theta) = |\langle C(T; \theta), C_{\text{target}} \rangle|$ where C_{target} is a prescribed target state. But this expression is not differentiable everywhere and numerically it is easier to replace it with its square. Moreover, to express the problem as a minimization, a -1 multiplicative constant is introduced and 1 added to the result to have it positive. So the cost functional \mathcal{J} will be the mean, over $\theta \in \Theta$ of the error in the fidelity squared as in formula (8). On the contrary, when the fidelity is

more well behaved as in Sec. III C where $\mathcal{F}(u; \theta) = c_4(T, \theta)$ or in Sec. III D where $\mathcal{F}(u; \theta) = c_6(T, \theta)$ the square operation is useless and the cost functional has the form in Eqs. (11) or (13). However, in all sections, we will plot the error in the fidelity itself; the reason why not plotting the fidelity (instead of the error) is that the error can be very small (as in Sec. III A) and the results are more visible on a logarithmic scale. Note that in some cases the best control cannot attain the target with 100% quality (even for a single molecule). However, for any given value of the parameter θ , the best attainable performance is known (see [1,33,34]) and is denoted $F_{\max}(\theta)$. We will therefore consider the fidelity relative to $F_{\max}(\theta)$. In all cases the error is computed as the average over $M_{\text{test}} = 300$ random independent parameters $\theta_1^{\text{test}}, \theta_2^{\text{test}}, \dots, \theta_{M_{\text{test}}}^{\text{test}}$ drawn (once for all) from the distribution $P(\theta)$ and has the following expression:

$$\frac{1}{M_{\text{test}}} \sum_{k=1}^{M_{\text{test}}} \left(1 - \frac{\mathcal{F}(u; \theta_k^{\text{test}})}{F_{\max}(\theta_k^{\text{test}})} \right). \quad (5)$$

For Secs. III A and III B we will also plot the max relative error

$$\max_{k=1, \dots, M_{\text{test}}} \left(1 - \frac{\mathcal{F}(u; \theta_k^{\text{test}})}{F_{\max}(\theta_k^{\text{test}})} \right). \quad (6)$$

Finally, to compare our algorithm with those from the literature, we take as indicator of the numerical effort the number of gradient $\nabla_u \mathcal{L}(u; \theta)$ evaluations; for instance one iteration of SGD or Adam algorithms count as M gradient evaluations. In all situations we used for the Adam algorithm the standard values $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\varepsilon = 10^{-8}$.

A. Two-level inhomogeneous ensemble

Consider an ensemble of spins as in [12, Sec. III]. The spins have different Larmor frequencies ω in the range $[0.8, 1.2]$ and the controls ($L = 2$) have multiplicative inhomogeneity $\epsilon \in [0.8, 1.2]$; we set $\theta = (\omega, \epsilon)$ and with the previous notations the dynamics corresponds to the equation

$$\begin{pmatrix} \dot{c}_1(t; \theta) \\ \dot{c}_2(t; \theta) \end{pmatrix} = \begin{pmatrix} 0.5\omega i & 0.5\epsilon[u_2(t) - iu_1(t)] \\ -0.5\epsilon[u_2(t) - iu_1(t)] & -0.5\omega i \end{pmatrix} \times \begin{pmatrix} c_1(t; \theta) \\ c_2(t; \theta) \end{pmatrix}, \quad (7)$$

where c_1, c_2 are the coefficients of the wave function of the spin system in the canonical basis, as detailed in Eq. (2).

The initial state of each member of the quantum ensemble is set to $|\psi_0\rangle = |0\rangle$; i.e., $C_0 = (1, 0)^T$, and the goal is to reach the target state $|\psi_{\text{target}}\rangle = |1\rangle$; i.e., $C_{\text{target}} = (0, 1)^T$. The objective is encoded as the requirement to minimize

$$J(u) = \frac{1}{2} \left(1 - \int_{\Theta} |\langle C(T; \theta), C_{\text{target}} \rangle|^2 P(d\theta) \right). \quad (8)$$

Here $F_{\max}(\theta) = 1$. The total time is $T = 2$ is divided into $Q = 200$ time steps, of length $\Delta t = T/Q = 0.01$ each. The initial choice for the control u is $u^{k=0}(t) = \{u_1^0(t) = \sin t, u_2^0(t) = \sin t\}$.

Several minibatch sizes $M = 1, 4, 8, 16$, and 32 are tested and compared to the implementation in [12, Sec. III A] where

a two-dimensional (2D) uniform grid of 5×5 values for θ is chosen. In all cases very good convergence results are attained. We plot in Fig. 1 the results for $M = 1, M = 4$ relative to the convergence with the uniform 5×5 grid. In all cases ($M = 1, 4$, uniform grid) we set $\alpha = 500$; note that the learning rate α was optimized to obtain the best possible results for the fixed grid algorithm and indeed the results are better than those in [12, Sec. III A]. But similar conclusions are reached for any value of α . An acceleration by a factor of 5 is obtained for both $M = 1$ and $M = 4$, essentially due to the fact that each SGD iteration uses only M gradient evaluations. Note that the SGD algorithm oscillates but these oscillations can be cured by lowering α (or stopping the search) as soon as a good result is obtained. The question of which is the best choice among $M = 1$ and $M = 4$ is a matter of striking a balance between speed and uncertainty: for $M = 4$ the convergence is slightly slower, but oscillations are diminished. This behavior is observed, to a larger or lesser extent, in all test cases.

Note that to compare our learning rate α (for the fixed uniform grid) with that in [12, Sec. III A] a multiplicative factor of $\Delta t/2$ has to be introduced because our gradient (see the Appendix) contains an extra Δt factor and the coefficient $1/2$. Thus one should transform $\alpha = 500$ to $1/2 \times 0.01 \times 500 = 2.5$ to compare to 0.2 used in [12].

B. Three-level Λ atomic ensemble

In this section we test a Λ atomic ensemble from [12, Sec. IV] which can be written as a three-level system with the following dynamics:

$$\begin{pmatrix} \dot{c}_1(t; \theta) \\ \dot{c}_2(t; \theta) \\ \dot{c}_3(t; \theta) \end{pmatrix} = \begin{pmatrix} -1.5\omega i & 0 & -i\epsilon u_2(t) \\ 0 & -\omega i & -i\epsilon u_1(t) \\ -i\epsilon u_2(t) & -i\epsilon u_1(t) & 0 \end{pmatrix} \begin{pmatrix} c_1(t; \theta) \\ c_2(t; \theta) \\ c_3(t; \theta) \end{pmatrix}, \quad (9)$$

where ω and ϵ have uniform distributions in $[0.8, 1.2]$ and c_1, c_2, c_3 are the coefficients of the wave function of the spin system in the canonical basis, as detailed in Eq. (2).

The objective is to find a control $u(t) = [u_1(t), u_2(t)]$ which drives all the inhomogeneous members from $|\psi_0\rangle = \frac{1}{\sqrt{3}}(|1\rangle + |2\rangle + |3\rangle)$ [i.e., $C_0 = (\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}})$] to $|\psi_{\text{target}}\rangle = |3\rangle$ [i.e., $C_{\text{target}} = (0, 0, 1)$]; the objective is encoded as the minimization of Eq. (8). Here $F_{\max}(\theta) = 1$.

We plot in Fig. 2 the results for $M = 1$ and $M = 4$ relative to the convergence with an uniform grid as in [12, Sec. IV]. In all cases ($M = 1, 4$, uniform grid) we set $\alpha = 100$. The acceleration factor is around 7 for $M = 4$ and even larger for $M = 1$ (but at the price of larger oscillations as well).

C. Three-dimensional example: Two spin systems without cross-correlated relaxation

As argued before, methods from the literature may have difficulties to address high-dimensional parameters, and often limit to two-dimensional ($d = 2$) inhomogeneity (see, however, [24, Sec. V.B] for a three-dimensional (3D) case). To test the full power of our method, we consider two situations that extend cases treated in the literature but have never been treated before. The first test is a 3D ($d = 3$) example which

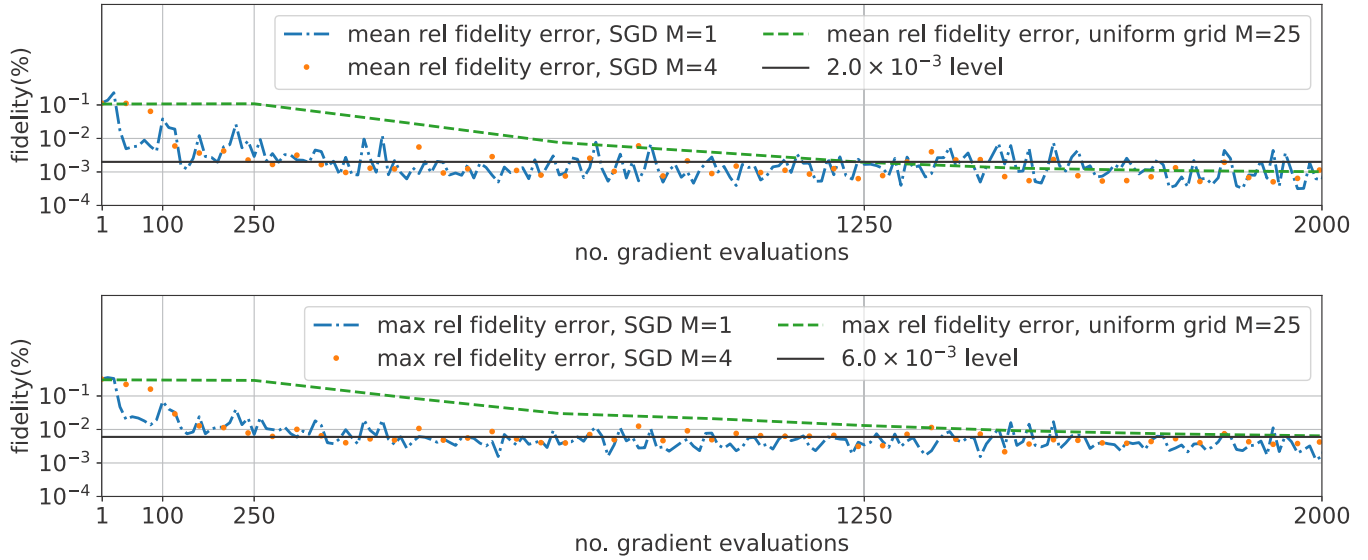


FIG. 1. Convergence for the numerical case in Sec. III A. Top image: Mean fidelity error [as defined in Eq. (5)]. Bottom image: Maximum (over the sample) fidelity error [as defined in Eq. (6)]. We consider three simulations: a fixed uniform 2D grid ($M = 25$) as in [12, Sec. III A] and the SGD algorithm with $M = 1$ and $M = 4$. This SGD converges about five times faster: the mean fidelity error of 2.0×10^{-3} is obtained after 1250 gradient evaluations of the fixed grid algorithm and after 250 evaluations of the SGD algorithm with $M = 1, 4$. Same for other levels of errors.

addresses the coherence transfer between two spins without cross-correlated relaxation, taken from [20, Sec. III.B.1. Eq. (15)] (but with an additional inhomogeneity dimension). An example of such a system is an isolated heteronuclear spin system composed of two coupled spins 1/2 corresponding to atoms 1H and ^{15}N . For a general treatment of the relaxation terms and the formulation of this equation see [29]. The spins display control inhomogeneity described by the parameter ϵ as above, but there is also variation in the relaxation rate and coupling constant, which, denoting $\theta = (\epsilon, J, \xi)$ results in the

dynamical system

$$\begin{pmatrix} \dot{c}_1(t; \theta) \\ \dot{c}_2(t; \theta) \\ \dot{c}_3(t; \theta) \\ \dot{c}_4(t; \theta) \end{pmatrix} = \begin{pmatrix} 0 & -\epsilon u_1(t) & 0 & 0 \\ \epsilon u_1(t) & -\xi & -J & 0 \\ 0 & J & -\xi & -\epsilon u_2 \\ 0 & 0 & \epsilon u_2 & 0 \end{pmatrix} \begin{pmatrix} c_1(t; \theta) \\ c_2(t; \theta) \\ c_3(t; \theta) \\ c_4(t; \theta) \end{pmatrix}. \tag{10}$$

Let us denote by $I_{1x} = \sigma_x/2, I_{1y} = \sigma_y/2, I_{1z} = \sigma_z/2$ (here $\sigma_x, \sigma_y, \sigma_z$ are the Pauli matrices) the spin operators corresponding

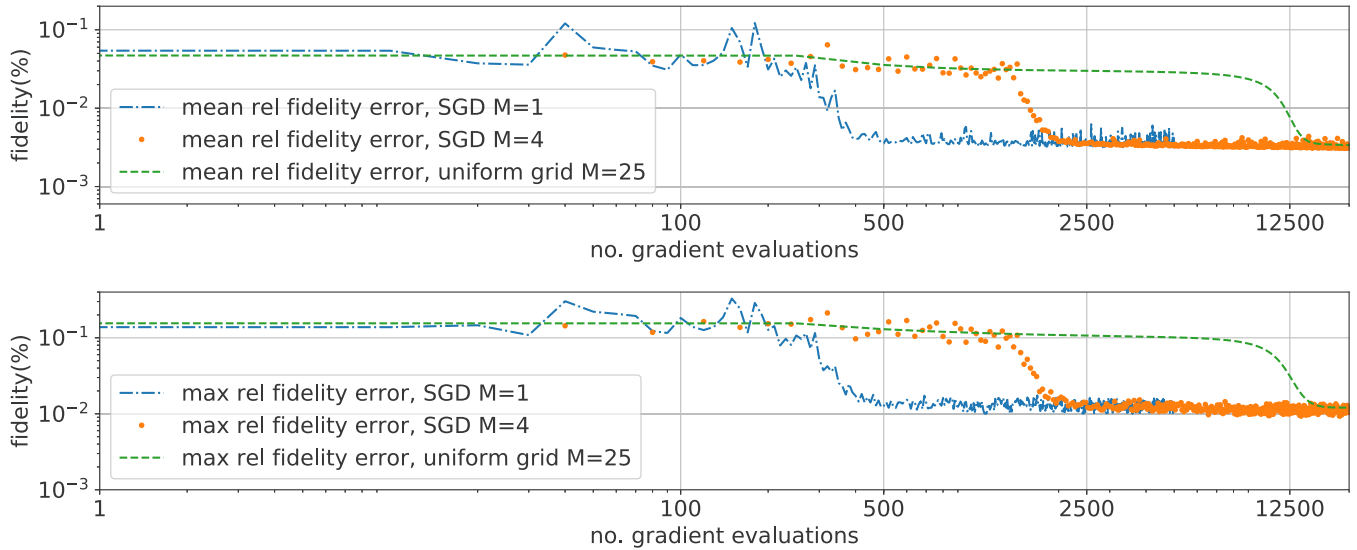


FIG. 2. Convergence for the numerical case in Sec. III B. Top image: Mean fidelity error as defined in Eq. (5). Bottom image: Maximum (over the sample) fidelity error [as defined in Eq. (6)]. We consider two algorithms: a fixed uniform 2D grid ($M = 25$) as in [12, Sec. IV] and the SGD algorithm with $M = 1$ and $M = 4$. This later approach converges about seven times faster: The convergence settles in after 17 500 gradient evaluations of the fixed grid algorithm compared to approximately 2500 evaluations of the SGD algorithm. This acceleration factor is even more important for $M = 1$, but at the price of larger oscillations.

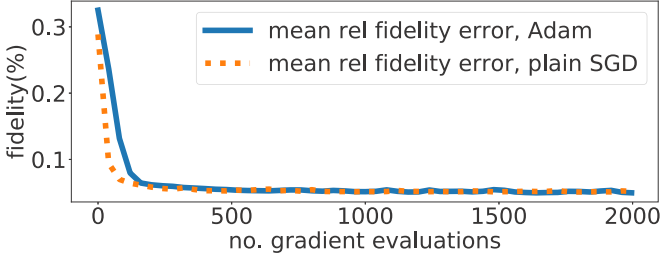


FIG. 3. Convergence for the numerical case in Sec. III C. The quantity plotted is given in Eq. (5). We set $M = 4$; for the SGD algorithm we choose $\alpha = 10.0$ and for the Adam algorithm we set $\alpha = 0.01$. The continuous (—) and dotted (· ·) curves stand for the mean fidelity errors of the plain SGD and Adam algorithm, respectively; the convergence is similar and a 95% mean target relative fidelity (or equivalently 5% mean target relative fidelity error) is obtained. For the controls see Fig. 4.

to the first spin and I_{2x}, I_{2y}, I_{2z} the corresponding objects for the second spin. With the usual notations for the Kronecker products, $c_1 = \langle I_{1z} \rangle$, $c_2 = \langle I_{1x} \rangle$, $c_3 = \langle 2I_{1y}I_{2z} \rangle$, $c_4 = \langle 2I_{1z}I_{2z} \rangle$; the exact derivation of this equation is beyond the scope of this work, see [1,29,34] for details. On the other hand also note that the dynamics is not reversible (relaxation is present) and the equations do not correspond to a unitary evolution.

The inhomogeneity $\theta = (\epsilon, J, \xi)$ is uniformly distributed in $\Theta = [0.9, 1.1] \times [0.5, 1.5] \times [0, 2]$. The final time $T = 7\pi/6$ is discretized with $Q = 200$ uniform time steps. The control is initialized as before. The initial state is encoded as $c_0 = (1, 0, 0, 0)$ and the target is to minimize the 3D integral

$$\mathcal{J}(u) = 1 - \int_{\Theta} c_4(T; \theta) P(d\theta). \quad (11)$$

Recall that here the fidelity is $\mathcal{F}(u; \theta) = c_4(T, \theta)$; in this case (see [1,34]) $F_{\max}(\theta) = \sqrt{1 + (\xi/J)^2} - \xi/J$ [the worse performance being $-F_{\max}(\theta)$]. The results are in Figs. 3 and 4. Note that although for each θ taken individually the figure $F_{\max}(\theta)$ can be attained with a pair (recall $L = 2$) of suitable control fields, it is unknown whether a unique control pair exists ensuring 100% [relative to $F_{\max}(\theta)$] target yield

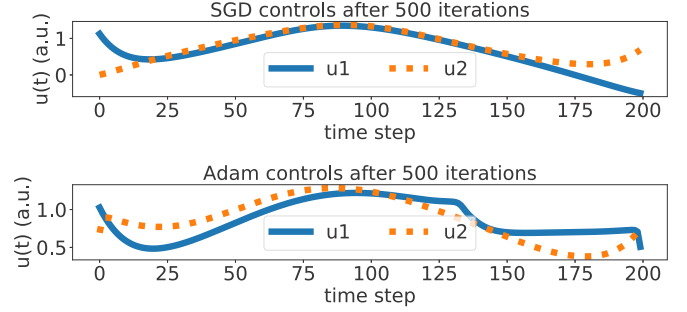


FIG. 4. Converged controls for the SGD (up) and Adam (bottom) for the situation in Sec. III C (for the convergence see Fig. 3). Controls obtained with the SGD algorithm are smoother than those from the Adam algorithm.

simultaneously for all $\theta \in \Theta$. In practice, we did not find any, irrespective of the algorithm hyperparameters such as α , the maximum number of iterations and so on; we conclude, on one hand, that this ensemble is not 100% simultaneously controllable and, on the other hand, that our procedure improves significantly the robustness of the control with respect to $\theta \in \Theta$ from an initial value of 67% up to 95%. Note that the results from the literature (which for this case only consider 2D inhomogeneity) do not obtain 100% control either (exact figure is not reported).

D. Six-dimensional example: Two spin systems with cross-correlated relaxation

We continue here to address alternate systems that previous methods could not treat. We consider an ensemble of two spin systems with cross-correlated relaxation as in [19, Sec. III.A.2], [20, Sec. III.B.2 Eq. (16)], and also [22, Example 3], [29].

The spins display control inhomogeneity described by the parameters ϵ_1 and ϵ_2 and there is also variation in the autocorrelated relaxation rate ξ_a , the quotient ξ_c/ξ_a of the cross-correlation relaxation rate ξ_c with respect to the autocorrelated relaxation rate ξ_a and finally, a dispersion in the Larmor frequencies of each spin. Denoting $\theta = (\epsilon_1, \epsilon_2, \omega_1, \omega_2, \xi_a, \xi_c/\xi_a) \in \Theta = [0.9, 1.1]^2 \times [0, 1]^2 \times [0.75, 1.25] \times [0.7, 0.9]$, the dynamical system can be written

$$\begin{pmatrix} \dot{c}_1(t; \theta) \\ \dot{c}_2(t; \theta) \\ \dot{c}_3(t; \theta) \\ \dot{c}_4(t; \theta) \\ \dot{c}_5(t; \theta) \\ \dot{c}_6(t; \theta) \end{pmatrix} = \begin{pmatrix} 0 & -\epsilon_1 u_1(t) & \epsilon_2 u_2(t) & 0 & 0 & 0 \\ \epsilon_1 u_1(t) & -\xi_a & \omega_1 & -J & -\xi_c & 0 \\ -\epsilon_2 u_2(t) & -\omega_1 & -\xi_a & -\xi_c & J & 0 \\ 0 & J & -\xi_c & -\xi_a & \omega_2 & -\epsilon_2 u_2(t) \\ 0 & -\xi_c & -J & -\omega_2 & -\xi_a & \epsilon_1 u_1(t) \\ 0 & 0 & 0 & \epsilon_2 u_2(t) & -\epsilon_1 u_1(t) & 0 \end{pmatrix} \begin{pmatrix} c_1(t; \theta) \\ c_2(t; \theta) \\ c_3(t; \theta) \\ c_4(t; \theta) \\ c_5(t; \theta) \\ c_6(t; \theta) \end{pmatrix}. \quad (12)$$

The vector $C = (c_1, \dots, c_6)$ has real entries and, with the same notations as in equation (10), $c_1 = \langle I_{1z} \rangle$, $c_2 = \langle I_{1x} \rangle$, $c_3 = \langle I_{1y} \rangle$, $c_4 = \langle 2I_{1y}I_{2z} \rangle$, $c_5 = \langle 2I_{1x}I_{2z} \rangle$, $c_6 = \langle 2I_{1z}I_{2z} \rangle$. The relations are similar to that in Sec. III C, with the exception that there are two new entries c_3 and c_5 due to the presence

of cross-correlation, see [1,29,34] for details of the derivation of the model; the dynamics is not reversible (relaxation is present) nor unitary.

We set $J = 1$; the total time $T = 5$ is discretized with $Q = 200$ uniform time steps. The control is initialized as before.

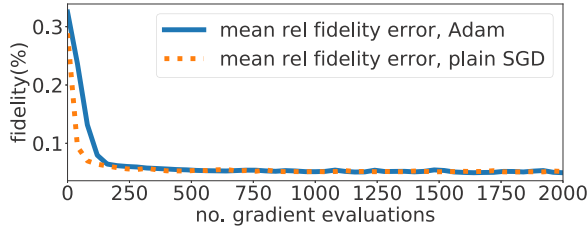


FIG. 5. Convergence for the numerical case in Sec. III D. The quantity plotted is defined as in the Fig. 3. We set $M = 4$; for the SGD algorithm we choose $\alpha = 10.0$ and for the Adam algorithm we set $\alpha = 0.01$. The continuous (—) and dotted (· ·) curves stand for the mean fidelity errors of the plain SGD and Adam algorithm, respectively; the convergence is similar and 91% mean relative fidelity is obtained. For the controls see Fig. 6.

The initial state is encoded as $c_0 = (1, 0, 0, 0, 0, 0)$ and the target is to minimize the six-dimensional integral

$$\mathcal{J}(u) = 1 - \int_{\Theta} c_6(T; \theta) P(d\theta). \quad (13)$$

Recall that here the fidelity is $\mathcal{F}(u; \theta) = c_6(T, \theta)$. In this case also the best attainable performance for a single molecule is known (see [1,34]) and defined by $F_{\max}(\theta) = \sqrt{1 + \eta^2} - \eta$ where $\eta = \sqrt{\frac{\xi_a^2 - \xi_c^2}{J^2 + \xi_c^2}}$.

The simulation results are in Figs. 5 and 6. The same conventions are kept as in the previous section (fidelity is relative to maximum attainable figure) and the same considerations still apply: 100% simultaneous controllability does not seem attainable but significant improvement in the robustness is obtained (91% up from -8%).

E. Stochastic convergence behaviors

The convergence of the stochastic algorithms can have two important regimes.

(1) First, when all members of the ensemble can be simultaneously optimized to 100%; in our situation this is equivalent to simultaneous controllability. In this case convergence is “easier” because it is “enough” to follow the gradient for each parameter value to converge; at convergence all gradients (as distribution with respect to ω), will collapse to (in practice will be close to) a Dirac mass.

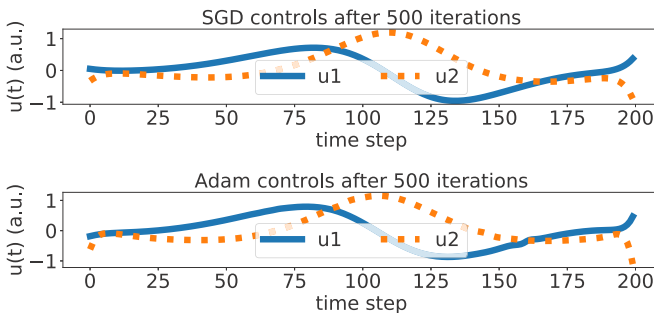


FIG. 6. Converged controls for the SGD (up) and Adam (bottom) for the situation in Sec. III D (for the convergence see Fig. 5). Controls obtained with the SGD algorithm are smoother than those from the Adam algorithm.

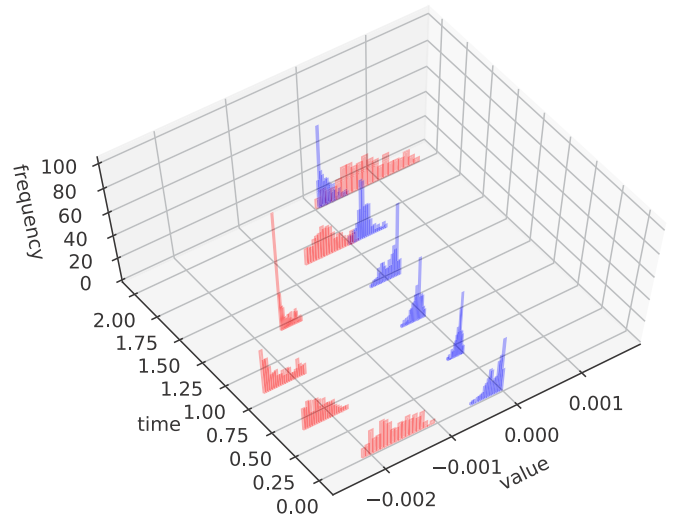


FIG. 7. Histogram of the gradients $\nabla_{u_1(t)} \mathcal{J}[u(t_n), \theta]$ computed over the test sample $\theta_1^{\text{test}}, \theta_2^{\text{test}}, \dots, \theta_{M_{\text{test}}}^{\text{test}}$ (recall $M_{\text{test}} = 300$). Six time instants t are chosen uniformly in $[0, T]; t = 0, T/5, 2T/5, \dots, T$. In red are the gradients at $u = u^1$ (iteration $k = 1$) and in blue the gradients at $u = u^{500}$ (iteration $k = 500$). Here we consider the case in Sec. III A, see Fig. 8 for the test case in Sec. III D.

(2) Second, when members of the ensemble cannot be simultaneously optimized; in this case, reaching full control for some θ value will harm the quality of some other parameter values $\theta' \neq \theta$. At convergence gradients will not be distributed as a Dirac mass any more, but the average with respect to theta will be zero (in practice small).

We illustrate this behavior in Figs. 7 and 8 where we plot the histograms of the gradient (with respect to the first field) $\nabla_{u_1(t)} \mathcal{J}[u(t_n), \theta]$ as random variables of θ at some time snapshots t . It is noticed that while in the first example it is possible to reduce significantly the gradient absolute value for all members of the sample (because simultaneous controllability holds true), in the second test case this reduction reaches

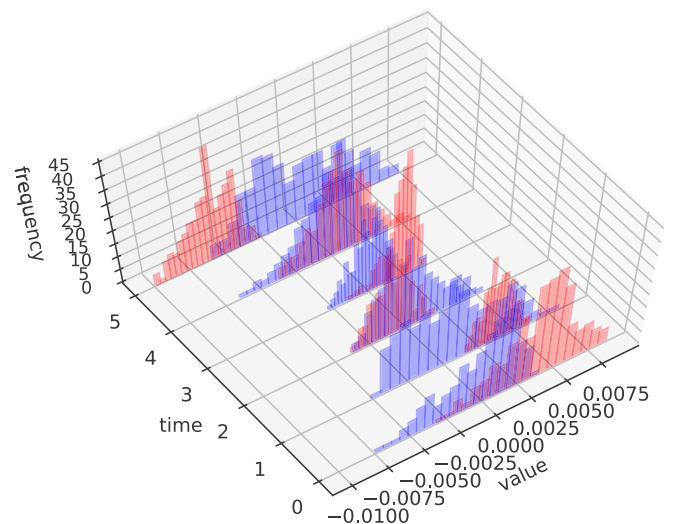


FIG. 8. Histogram of the gradients as in Fig. 7 except that here the results correspond to the test case in Sec. III D.

a limit and the algorithm tries instead to center the gradients on zero so that the average will be as low as possible.

IV. DISCUSSION AND CONCLUSION

We proposed and tested in this work a stochastic approach to compute the optimal controls of inhomogeneous quantum ensembles. The algorithms were employed before in other areas of stochastic optimization but not tested in this context (see [26] for similar algorithms). Their specificity is to draw at each iteration a new set of parameters from the inhomogeneous distribution. Although at first the intuition may not recommend such an approach, the numerical results indicate not only convergence but also faster convergence than methods based on fixed samples. In addition the method can address situations when the space of parameters is large and was tested successfully on a six-dimensional example.

For lower-dimensional examples (as in Secs. III A and III B) the acceleration of the stochastic algorithms (SGD, Adam) is due essentially to the lower effort per iteration compared to a fixed grid sampling (both being proportional to the number of samples used). In higher dimensions the fixed grid approach is inherently less efficient due to the *curse of dimensionality* and may even be prohibitively large.

On the other hand, compared to SGD, the Adam algorithm has the advantage to be more robust with respect to the choice of the learning rate α , but the controls are less regular.

Finally, one of the limitations of this work is to use constant learning rates. Variable learning rates are potentially interesting as it could speed up convergence in the initial

phases by using large values of α and avoid oscillations in the end by lowering α . Several schedules are proposed in the stochastic optimization literature (inverse linear, piecewise constant, etc.), but their analysis remains for future work.

APPENDIX: GRADIENT COMPUTATION

We detail below the computation of the gradient for a single parameter θ , the general case being just a mean over θ . Consider the so-called adjoint state $\lambda(t; \theta)$; it is defined at the final time as the derivative of the outcome with respect to $C(T; \theta)$. For instance, for Secs. III A to III B: $\lambda(T; \theta) = -(C_{\text{target}}, C(T, \theta))_{C_{\text{target}}}$ while for Secs. III C to III D we set $\lambda(T; \theta) = -1$. Then for $t < T$, $\lambda(t; \theta)$ is the solution of the (backward) equation $\frac{d}{dt}\lambda(t; \theta) = X(t, \theta)^\dagger \lambda(t; \theta)$, where $X(t, \theta)^\dagger$ is the transpose conjugate of X when X has complex entries (examples in Secs. III A and III B) and reduces to the transpose when X is a real matrix (examples in Secs. III C and III D). Then $\nabla_{u(t)} \mathcal{J} = \langle \lambda(t; \theta), \frac{\partial X(t; \theta)}{\partial u(t)} C(t; \theta) \rangle$. In practice, given that u is discretized, the state C and the adjoint state λ are also discretized at time instants $t_n = n\Delta t$: $C_n(\theta) \simeq C(t_n; \theta)$, $\lambda_n(\theta) \simeq \lambda(t_n; \theta)$ which satisfy $C_{n+1}(\theta) = e^{\Delta t X[u(t_n); \theta]} C_n(\theta)$ and $\lambda_n(\theta) = e^{\Delta t X[u(t_n); \theta]^\dagger} \lambda_{n+1}(\theta)$ and the exact discrete gradient is $\nabla_{u(t_n)} \mathcal{J} = \langle \lambda_{n+1}(\theta), \frac{\partial e^{\Delta t X[u(t_n); \theta]}}{\partial u(t_n)} C_n(\theta) \rangle$.

Finally, to compute $\frac{\partial e^{\Delta t X[u(t_n); \theta]}}{\partial u(t_n)}$ we use a “divide and conquer” approach coupled with an eighth-order expansion as in [35, formula (11)] to obtain at the same time the exponential and the gradient ([36, Chap. VI]) from the knowledge of the inputs $X[u(t_n); \theta]$ and $\frac{\partial X[u(t_n); \theta]}{\partial u_i(t_n)}$.

-
- [1] S. J. Glaser, T. Schulte-Herbrüggen, M. Sieveking, O. Schedletzky, N. C. Nielsen, O. W. Sørensen, and C. Griesinger, Unitary control in quantum ensembles: Maximizing signal intensity in coherent spectroscopy, *Science* **280**, 421 (1998).
 - [2] K. Khodjasteh and L. Viola, Dynamically Error-Corrected Gates for Universal Quantum Computation, *Phys. Rev. Lett.* **102**, 080501 (2009).
 - [3] C. Brif, R. Chakrabarti, and H. Rabitz, Control of quantum phenomena: past, present and future, *New J. Phys.* **12**, 075008 (2010).
 - [4] J.-S. Li and N. Khaneja, Control of inhomogeneous quantum ensembles, *Phys. Rev. A* **73**, 030302(R) (2006).
 - [5] T. E. Skinner, T. O. Reiss, B. Luy, N. Khaneja, and S. J. Glaser, Application of optimal control theory to the design of broadband excitation pulses for high-resolution nmr, *J. Magn. Reson.* **163**, 8 (2003).
 - [6] B. Yurke and J. S. Denker, Quantum network theory, *Phys. Rev. A* **29**, 1419 (1984).
 - [7] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information: 10th Anniversary Edition*, 10th ed. (Cambridge University Press, New York, 2011).
 - [8] I. N. Hincks, C. E. Granade, T. W. Borneman, and D. G. Cory, Controlling Quantum Devices with Nonlinear Hardware, *Phys. Rev. Applied* **4**, 024012 (2015).
 - [9] R. L. Kosut, M. D. Grace, and C. Brif, Robust control of quantum gates via sequential convex programming, *Phys. Rev. A* **88**, 052326 (2013).
 - [10] H. Rabitz and G. Turinici, Controlling quantum dynamics regardless of laser beam spatial profile and molecular orientation, *Phys. Rev. A* **75**, 043409 (2007).
 - [11] Y. Zhang, M. Lapert, D. Sugny, M. Braun, and S. J. Glaser, Time-optimal control of spin 1/2 particles in the presence of radiation damping and relaxation, *J. Chem. Phys.* **134**, 054103 (2011).
 - [12] C. Chen, D. Dong, R. Long, I. R. Petersen, and H. A. Rabitz, Sampling-based learning control of inhomogeneous quantum ensembles, *Phys. Rev. A* **89**, 023402 (2014).
 - [13] H.-J. Ding and R.-B. Wu, Robust quantum control against clock noises in multiqubit systems, *Phys. Rev. A* **100**, 022302 (2019).
 - [14] G. Turinici, V. Ramakrishna, B. Li, and H. Rabitz, Optimal discrimination of multiple quantum systems: controllability analysis, *J. Phys. A: Math. Gen.* **37**, 273 (2004).
 - [15] M. Belhadji, J. Salomon, and G. Turinici, Ensemble controllability and discrimination of perturbed bilinear control systems on connected, simple, compact Lie groups, *Eur. J. Control* **22**, 23 (2015).
 - [16] K. Beauchard, J.-M. Coron, and P. Rouchon, Controllability issues for continuous-spectrum systems and ensemble controllability of Bloch equations, *Commun. Math. Phys.* **296**, 525 (2010).
 - [17] A. Borzì, G. Ciaramella, and M. Sprengel, *Formulation and Numerical Solution of Quantum Control Problems* (SIAM, Philadelphia, 2017), Vol. 16.

- [18] N. Augier, U. Boscain, and M. Sigalotti, Adiabatic ensemble control of a continuum of quantum systems., *SIAM J. Control Optim.* **56**, 4045 (2018).
- [19] J.-S. Li, J. Ruths, and D. Stefanatos, A pseudospectral method for optimal control of open quantum systems, *J. Chem. Phys.* **131**, 164110 (2009).
- [20] J. Ruths and J.-S. Li, A multidimensional pseudospectral method for optimal control of quantum ensembles, *J. Chem. Phys.* **134**, 044128 (2011).
- [21] J.-S. Li, J. Ruths, T.-Y. Yu, H. Arthanari, and G. Wagner, Optimal pulse design in quantum control: A unified computational method, *Proc. Natl. Acad. Sci. USA* **108**, 1879 (2011).
- [22] S. Wang and J.-S. Li, Free-endpoint optimal control of inhomogeneous bilinear ensemble systems, *Automatica* **95**, 306 (2018).
- [23] S. Kuang, P. Qi, and S. Cong, Approximate time-optimal control of quantum ensembles based on sampling and learning, *Phys. Lett. A* **382**, 1858 (2018).
- [24] C. Wu, B. Qi, C. Chen, and D. Dong, Robust learning control design for quantum unitary transformations, *IEEE Trans. Cybern.* **47**, 4405 (2017).
- [25] W. Press, S. Teukolsky, W. Vetterling, and B. Flannery, *Numerical Recipes 3rd Edition: The Art of Scientific Computing* (Cambridge University Press, 2007).
- [26] R.-B. Wu, H. Ding, D. Dong, and X. Wang, Learning robust and high-precision quantum controls, *Phys. Rev. A* **99**, 042327 (2019).
- [27] C. M. Dion, A. Keller, O. Atabek, and A. D. Bandrauk, Laser-induced alignment dynamics of HCN: Roles of the permanent dipole moment and the polarizability, *Phys. Rev. A* **59**, 1382 (1999).
- [28] J.-M. Coron, A. Grigoriu, C. Lefter, and G. Turinici, Quantum control design by Lyapunov trajectory tracking for dipole and polarizability coupling, *New J. Phys.* **11**, 105034 (2009).
- [29] P. Allard, M. Helgstrand, and T. Hard, The complete homogeneous master equation for a heteronuclear two-spin system in the basis of cartesian product operators, *J. Magn. Reson.* **134**, 7 (1998).
- [30] H. Robbins and S. Monro, A stochastic approximation method, *Ann. Math. Stat.* **22**, 400 (1951).
- [31] S. Ruder, An overview of gradient descent optimization algorithms, [arXiv:1609.04747](https://arxiv.org/abs/1609.04747).
- [32] D. P. Kingma and J. Ba, Adam: A method for stochastic optimization, [arXiv:1412.6980](https://arxiv.org/abs/1412.6980).
- [33] G. Turinici and H. Rabitz, Quantum wavefunction controllability, *Chem. Phys.* **267**, 1 (2001).
- [34] N. Khaneja, B. Luy, and S. J. Glaser, Boundary of quantum evolution under decoherence, *Proc. Natl. Acad. Sci. USA* **100**, 13162 (2003).
- [35] P. Bader, S. Blanes, and F. Casas, An improved algorithm to compute the exponential of a matrix, [arXiv:1710.10989](https://arxiv.org/abs/1710.10989).
- [36] L. B. Rall, *Automatic Differentiation: Techniques and Applications*, Lecture Notes in Computer Science Vol. 120 (Springer, Berlin, 1981).