



OPEN ACCESS

RECEIVED
5 June 2023REVISED
15 September 2023ACCEPTED FOR PUBLICATION
6 October 2023PUBLISHED
23 October 2023

Original Content from
this work may be used
under the terms of the
[Creative Commons
Attribution 4.0 licence](#).

Any further distribution
of this work must
maintain attribution to
the author(s) and the title
of the work, journal
citation and DOI.



PAPER

High-dimensional multi-fidelity Bayesian optimization for quantum control

Marjuka F Lazi^{*} , Christian R Shelton , Simon N Sandhofer and Bryan M Wong

University of California, Riverside, CA, United States of America

^{*} Author to whom any correspondence should be addressed.E-mail: mlazi003@ucr.edu**Keywords:** Bayesian optimization, quantum control, multi-fidelity

Abstract

We present the first multi-fidelity Bayesian optimization (BO) approach for solving inverse problems in the quantum control of prototypical quantum systems. Our approach automatically constructs time-dependent control fields that enable transitions between initial and desired final quantum states. Most importantly, our BO approach gives impressive performance in constructing time-dependent control fields, even for cases that are difficult to converge with existing gradient-based approaches. We provide detailed descriptions of our machine learning methods as well as performance metrics for a variety of machine learning algorithms. Taken together, our results demonstrate that BO is a promising approach to efficiently and autonomously design control fields in general quantum dynamical systems.

1. Introduction

Inverse problems pose some of the most intriguing yet difficult questions in the physical sciences as they start with an observed effect and require us to calculate its cause. This is distinctly different from the more common forward (or initial value) problem, which starts with the cause and subsequently calculates the effects. In the context of chemical/material systems, quantum optimal control [1] is one of the best-known examples of an inverse problem; it seeks to drive the system from a known initial state to a desired target state using a tailored external electromagnetic pulse, $E(t)$. Constructing $E(t)$ is central to providing critical initial conditions for time-resolved experiments including light-harvesting complexes [2–6], quantum information processing [7–9], and laser cooling [10–13]. Controlling these phenomena could have ground-breaking implications in chemistry and materials science since it would allow manipulation of these systems at a time-resolved quantum level of detail.

Although several theoretical approaches and algorithms [14–21] have been used to construct optimal quantum control fields, most of them are iterative and require intricate numerical techniques to achieve convergence. The nonlinear nature of these dynamic optimization problems results in a large number of floating point operations [16, 22], rendering them computationally intractable for high-dimensional quantum systems. To address these computational bottlenecks, we previously carried out the first ‘conventional’ machine learning approaches, including convolutional neural networks [23] and reinforcement learning [24], to solve quantum control problems of continuous systems (i.e. systems characterized by a continuous Hilbert space as opposed to finite-dimensional spin-1/2 systems). While these previous machine learning approaches showed some success, the training and validation processes in these approaches were quite computationally intensive.

To address these computational issues, we present the first high-dimensional Bayesian optimization (BO) machine learning approach for efficiently solving time-dependent quantum control problems in reduced-dimensional chemical systems. While Bayesian statistics have been used before [25–27], they have not performed BO, a particular approach to optimization. One notable exception [28] optimizes time-varying control; however, it focuses on how to adapt the noise model and considers controls with five degrees of freedom (compared with the 22 in this work). These dynamical time-dependent systems pose a

unique challenge for machine learning techniques, and we investigate a variety of approaches for predicting optimal control fields, $E(t)$, in these systems. We extend the existing sparse axis-aligned subspace BO (SaasBO) method to allow for multiple computational fidelities¹ to speed up computation. This novel method, multi-fidelity SaasBO (MFSaasBO), performs better than other computational approaches.

We first briefly outline the basic concepts of quantum control and the data inputs/outputs used by the machine learning approaches in our work. Next, we describe the details of our BO approach and compare its performance against conventional gradient ascent methods typically used in quantum control algorithms. Finally, we conclude with a brief discussion and perspective look at potential future applications and extensions of our machine learning approach.

2. Background

2.1. Brief overview of quantum optimal control

Since the primary objective of this study is to harness BO techniques to control dynamic chemical systems, we only present a concise introduction to quantum optimal control. Interested readers are directed to relevant topical reviews [29–32] and our prior work [23, 33] for further details. For chemical systems, the quantum optimal control formalism commences with the time-dependent Schrödinger equation for describing the temporal dynamics of nuclei, which, in atomic units is given by

$$i \frac{\partial}{\partial t} \psi(x, t) = \left[-\frac{1}{2m} \frac{\partial^2}{\partial x^2} + V(x) - \mu(x) E(t) \right] \psi(x, t). \quad (1)$$

In the equation above, x signifies the reduced coordinate along a chosen reaction path [34–37], m is the effective mass associated with the molecular motion along this path [38, 39], $V(x)$ is the Born–Oppenheimer electronic energy of the molecule, $\mu(x)$ is the dipole moment function, $E(t)$ is the time-dependent external electric field, and $\psi(x, t)$ represents the probability amplitude for nuclear motion along the reduced coordinate path. Both $V(x)$ and $\mu(x)$ can be obtained from a standard quantum chemistry calculation of a relaxed potential energy scan [40, 41].

With x and $V(x)$ properly chosen or computed, equation (1) is an initial value problem that poses the question: ‘Given an electric field $E(t)$, how does an initial state, $\psi_0(x, t=0)$, evolve after some final time T has elapsed?’ However, as described in the introduction, quantum optimal control instead seeks the answer to the inverse question: ‘To achieve a desired final state $\psi_{N-1}(x, t=T)$ at time T (after $N-1$ propagation steps), what functional form should $E(t)$ take?’ More precisely, quantum control seeks the functional shape of an external electric field, $E(t)$, that maximizes the functional $P[\psi_{N-1}, E]$ given by:

$$P[\psi_{N-1}, E] = \left| \int_{-\infty}^{\infty} \psi_f^*(x) \psi_{N-1}(x) dx \right|^2, \quad (2)$$

where ψ_f is a desired final target wavefunction provided by the user, and ψ_{N-1} is obtained after applying $N-1$ consecutive propagation steps of the time-dependent Schrödinger equation (i.e. equation (1)). In short, equation (2) mathematically measures the similarity of the final target and the propagated wavefunction. Providing accurate and efficient answers to this inverse question is the ultimate goal of the BO approaches described in this work.

2.2. Bayesian optimization

BO [42, 43] is an optimization process that finds the location $x \in R^d$ corresponding to the global maxima (or minima) of a function $f: R^d \rightarrow R$. It is predicated on the assumption that f is a computationally expensive black-box function. BO expedites the optimization of f by using all prior evaluations of f to select the next point x at which to evaluate f . BO tracks a posterior distribution over the function f conditioned on the evaluated points thus far, and it uses this distribution over the space of functions to select the next point to query the function with the goal of constraining the position of the maximum of the function as quickly as possible.

The process of BO starts by sampling the function f on a small subset of the input domain $\{x_1, x_2, \dots, x_t\}$. These points are used to construct the conditional probability distribution $P(f | f(x_1), f(x_2), \dots, f(x_t))$, also known as the surrogate model. This model function is much cheaper to compute than f and can estimate $f(x)$ (and uncertainty about the value of $f(x)$) for unobserved values of x .

¹ In this paper, ‘fidelity’ is used in its Bayesian optimization context: the accuracy of the computation of the function to be optimized. We do not use it to refer to the quality of the found solution.

This surrogate model drives the choice of the next value of x to query; $f(x)$ is computed for this new x , the surrogate is updated to include the new information, and the optimization process continues.

To select the next sample point x given the surrogate function, BO relies on an acquisition function that maps x to a heuristic value of the benefit of selecting x as the next point to evaluate. The general form of BO procedure is as follows:

- Choose a surrogate model for modeling the true function f and define its prior.
- Given a set of data or observations $D_t = \{x_i, y_i\}_{i=1, \dots, t-1}$, where x_i is the input chosen to be queried on the i th iteration, and $y_i = f(x_i)$ is the function evaluated at x_i , obtain the surrogate posterior: $P(f | D_t)$.
- Use an acquisition function $\alpha(x)$, which is a function of the posterior, to find the next point $x_t = \arg \max_x \alpha(x)$.
- Evaluate $y_t = f(x_t)$ and add $\{x_t, y_t\}$ to D_t and repeat the procedure from step b until the given budget (number of iterations or total computation time) elapses.

2.3. Gaussian processes

BO usually uses a Gaussian process (GP) [44] as its surrogate model. A GP can be defined as an infinite collection of random variables, indexed by x , such that the joint distribution of any finite subset of the variables is a multivariate Gaussian. The surrogate model assumption of a GP is denoted $f \sim \text{GP}(\mu, k)$, where the parameters of the process are the mean function $\mu(x)$, and the kernel/covariance-function $k(x, x')$.

Unless we have some knowledge about the properties of our function f , the mean function μ is typically assumed to be zero. The kernel encodes information about the prior distribution's smoothness: how similar in value (correlated) two function values $f(x)$ and $f(x')$ are based on the distance between x and x' .

A myriad of kernel functions are possible. The squared-exponential kernel or radial-basis function (RBF) kernel is most commonly chosen because it is infinitely differentiable and decays as a function of distance [45]

$$k_{\text{RBF}}(x, x') = \sigma^2 e^{-\frac{1}{2} \sum_d \rho_d (x_d - x'_d)^2}, \quad (3)$$

where σ^2 is the variance of $f(x)$ at any point x and ρ_d is the inverse squared length scale associated with dimension d .

Given any set of points $\{x_1, x_2, \dots, x_t\}$, the corresponding function values $\{f(x_1), f(x_2), \dots, f(x_t)\}$ are jointly Gaussian:

$$\begin{bmatrix} f(x_1) \\ f(x_2) \\ \vdots \\ f(x_t) \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} \mu(x_1) \\ \mu(x_2) \\ \vdots \\ \mu(x_t) \end{bmatrix}, \begin{bmatrix} k(x_1, x_1) & k(x_1, x_2) & \cdots & k(x_1, x_t) \\ k(x_2, x_1) & k(x_2, x_2) & \cdots & k(x_2, x_t) \\ \vdots & \vdots & \ddots & \vdots \\ k(x_t, x_1) & k(x_t, x_2) & \cdots & k(x_t, x_t) \end{bmatrix} \right), \quad (4)$$

where $\mathcal{N}(m, \Sigma)$ denotes a multivariate normal distribution with mean vector m and covariance matrix Σ .

To model the predictive distribution $P(f(x_t) | f(x_1), f(x_2), \dots, f(x_{t-1}))$, we formulate the joint distribution as in equation (4) and condition it on all values except $f(x_t)$, resulting in a normal posterior distribution over $f(x_t)$ using standard multivariate normal distribution formulas. We denote the mean and standard deviation of $f(x)$ given $\{f(x_1), f(x_2), \dots, f(x_{t-1})\}$ as $\mu_t(x)$ and $\sigma_t(x)$. Thus $(x | D_t) \sim \mathcal{N}(\mu_t(x), \sigma_t^2(x))$.

2.4. Acquisition function

In BO, the acquisition function is used to estimate the benefit of next evaluating f at point x . Common acquisition functions include expected improvement (EI) [46], probability of improvement (PI) [47], and upper confidence bound [48]. PI chooses the next query point with the highest probability of being larger than the best point found so far, which often results in under-exploration and missing the global optima. EI was proposed as an alternative to PI that chooses the point with the highest Expected Improvement over the best point found so far. EI performs poorly when sampling from low fidelity evaluations as it does not consider correlation between different fidelities [49]. The knowledge-gradient (KG) [50] acquisition function is similar to EI, but allows for the possibility that the new maximum of the surrogate model may not be at the newly chosen point. If the evaluation of f is noiseless, this is not the case and EI and KG are the same.

As the function f in our case is deterministic, we will mostly focus on utilizing the EI method. However, we revisit KG in section 2.6. The goal of EI is to choose the next query point where the EI over the current best point found so far is the highest. Let $y^+ = \max_i y_i$ be the best value found so far. The analytical expression of EI for a GP surrogate model is [51]

$$\text{EI}(x) = E[\max(0, f(x) - y^+)], \quad (5)$$

where the expectation is with respect to the surrogate model's posterior distribution over the value of $f(x)$, conditioned on all of the observed data so far. If the surrogate is a GP, the posterior of $f(x)$ is a normal distribution and the analytic form for the EI can be written as a function of the mean and the standard deviation of the posterior distribution of $f(x)$:

$$\text{EI}(x) = g(\mu_t(x), \sigma_t(x), y^+), \quad (6)$$

where

$$g(\mu, \sigma, y^+) = \begin{cases} (\mu - y^+) \Phi\left(\frac{\mu - y^+}{\sigma}\right) + \sigma \phi\left(\frac{\mu - y^+}{\sigma}\right) & \text{if } \sigma > 0, \\ 0 & \text{if } \sigma = 0, \end{cases} \quad (7)$$

and $\Phi(\cdot)$ and $\phi(\cdot)$ are the CDF and PDF of the unit normal distribution. When $\mu_t(x) - f(x^+)$ (the difference between the expected value at x and the best value found so far) or the uncertainty of the value at the point x , $\sigma_t(x)$, is high, EI will be high.

2.5. High-dimensional Bayesian optimization

BO becomes challenging as the input dimension grows. Learning the probabilistic surrogate model (GP in our case) becomes exponentially complex as the number of parameters increases [52]. Many methods have been proposed to address the problems of BO in the high dimensional spaces. Most of them use the idea of optimizing the objective function in a low dimensional space by mapping the high dimensional space to a low dimensional space [52–56].

Others assume additive structures in the objective function [57–61]. Another approach uses a cylindrical transformation [62] for scaling to the higher dimension. Li *et al* [63] use a dropout strategy by randomly selecting some dimensions and filling in the dropped out dimensions using different methods such as random values etc. Other techniques also include a geometry-aware BO method based on Riemannian manifolds [64]. A trust region based method has also been proposed that confines the search region to promising regions [65].

We found that SaasBO [66] outperforms other methods for our quantum control problem. It places a prior distribution over the parameters of a GP surrogate. In particular, it places a half-Cauchy distribution over each ρ_d (see equation (3)). This encourages the inverse (squared) length scales to be near zero, thus encouraging the surrogate to model f as depending only on a few dimensions (those with inverse length scales that are non-zero). However, half-Cauchy distributions have heavy tails, and thus with adequate evidence, the posterior distributions over ρ_d will be directed to larger values, allowing BO to explore more complex functions. Because SaasBO places a distribution over the parameters of the GP, the surrogate is no longer a GP, but rather an infinite mixture of (or distribution over) GPs. The probability distribution of function f conditioning on D is

$$P(f(x) | D) = \int P(\mu, k | D) N(f(x) | \mu, k, D) d\mu dk, \quad (8)$$

where $N(f(x) | \mu, k, D)$ is the posterior GP distribution over $f(x)$ given mean function μ , kernel function k , and data D (as in section 2.3). This integral is approximated as an average of L samples drawn from $P(\mu, k | D)$ (the posterior distribution over the GP parameters): $\frac{1}{L} \sum_{l=1}^L N(f(x) | \mu_l, k_l, D)$, where μ_l and k_l are the mean and kernel functions from the l th sample of GP parameters. The samples are drawn from the posterior distribution over the parameters of the GP using a variant of Markov chain Monte Carlo, the no-U-turn sampler (NUTS) [67].

Thus the approximation of the posterior over f is a mixture of GPs. For an acquisition function like EI, its evaluation is the average over the samples of the same formula for a GP:

$$\text{EI}(x) = \frac{1}{L} \sum_{l=1}^L g(\mu_t^l(x), \sigma_t^l(x), y^+), \quad (9)$$

where $\mu_t^l(x)$ and $\sigma_t^l(x)$ are the posterior mean and covariance of $f(x)$ for the GP using the l th sample from NUTS.

The query point x_t is achieved by optimizing the EI function using L-BFGS-B [68].

2.6. Multi-fidelity Bayesian optimization

In some optimization processes, the function f can be evaluated at different fidelities. In our problem, the fidelity corresponds to the temporal resolution at which the system is simulated (not the value of the functional P), as is typical in the BO literature. Function evaluations at lower fidelities reduce computation time, but the results are less accurate. The goal of multi-fidelity BO is to use low-fidelity evaluations in combination with full-fidelity evaluations to find the maximum of the full-fidelity function f . Note that while low-fidelity evaluations may more quickly help to find the maximum, only full-fidelity evaluations are relevant for determining the success of the optimization. Therefore any multi-fidelity BO method must carefully consider whether to make a low- or high-fidelity query each iteration of the algorithm.

We let $z \in \{0, 1, \dots, N\}$ denote the fidelity, with $z = 0$ being the high ‘true’ fidelity. In this work we focus on the $N = 1$ low fidelity approximation, but the technique can be generalized. We let $f(x, z)$ be the result of evaluating the function at x with fidelity z . At iteration t in the algorithm, multi-fidelity BO selects x_t (the function input) and z_t (the function fidelity) and receives $y_t = f(x_t, z_t)$. To complete the multi-fidelity set up, a known cost function is required; $\lambda : Z \rightarrow R$, which denotes the (often computational) cost of evaluating the function f at fidelity z .

Different multi-fidelity procedures have been proposed, such as an auto-regressive approach to correlate between the lower fidelity function and the higher fidelity function by applying linear regression between different levels of fidelity [69]. Kandasamy *et al* [70] use separate GP for each fidelity level to explore the input space in lower fidelity. Song *et al* [71] suggest a two phased model with combined GPs across multiple fidelity levels and emphasize on mutual information acquired from different fidelities. To better understand the correlations between different fidelities, a deep neural network based model [72] has also been proposed that uses neural networks to represent each fidelity. Multi-fidelity methods are particularly attractive for high-dimensional problem spaces to reduce the increased computational cost associated with needing more function evaluations.

3. Computational approach

We have added a multi-fidelity approach to SaasBO [66] to gain both the benefits of SaasBO on high-dimensional search problems as well as the computational gains of multi-fidelity functions to save on evaluation time.

For our quantum control problem, we can simulate the evolution of the quantum system with different temporal resolutions. For example, we select a resolution of one-third the target resolution as our low-fidelity approximation for our experimental results.

To add multi-fidelity capability to SaasBO, we use the SaasBO surrogate on the function $f(x, z)$ —that is, we augment x with an extra dimension, the fidelity, and build a surrogate model on this $d + 1$ -dimensional function.

Our acquisition function is the EI in the high-fidelity function’s maximum. We call this improvement $EI_0(x, z)$ to indicate that it is the EI in the high-fidelity (0) from querying point x with fidelity z :

$$EI_0(x, z) = E \left[\max \left(0, \max_{x'} f(x', 0) - y^+ \right) \right], \quad (10)$$

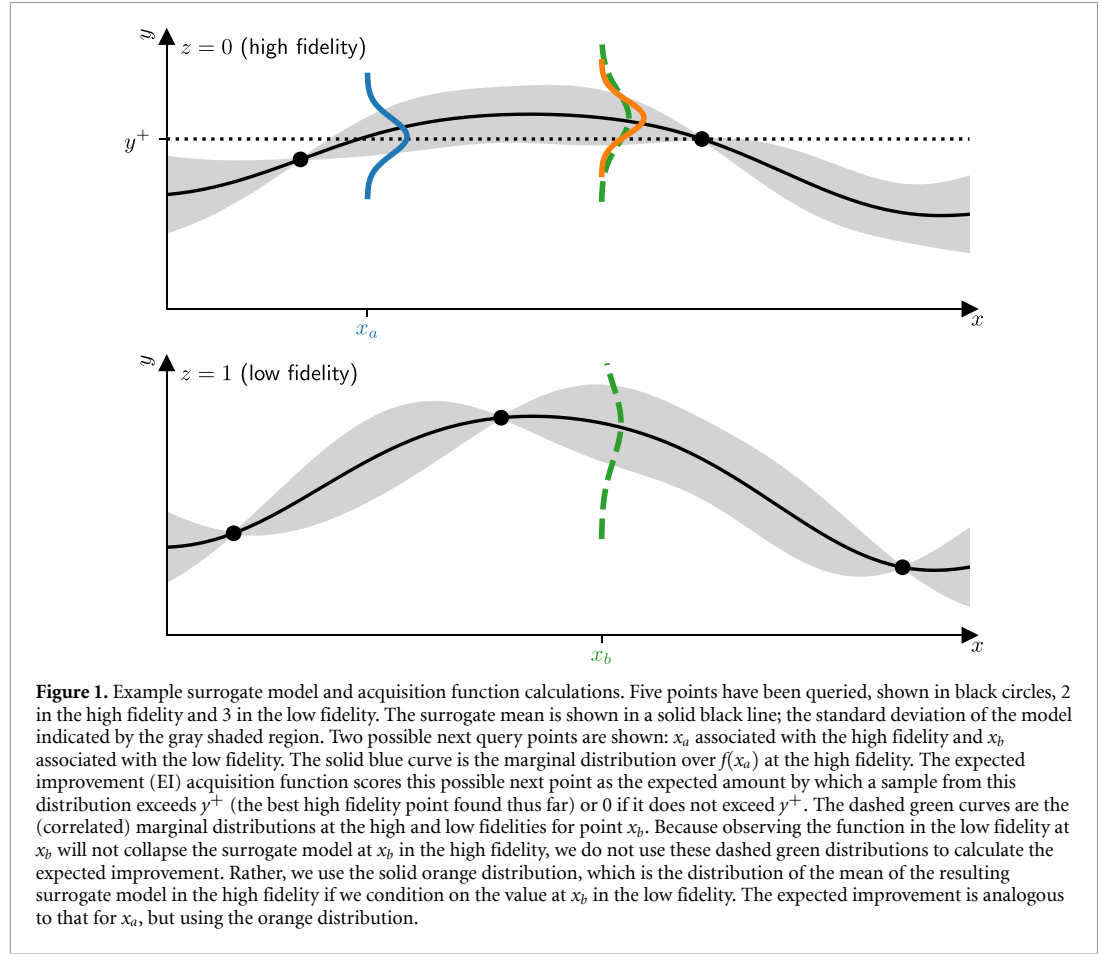
where

$$y^+ = \max_{i|z_i=0} y_i \quad (11)$$

is the maximum value found so far during high-fidelity evaluations. We divide by the cost of the associated fidelity to score a particular (x, z) choice based on the rate at which it improves:

$$\alpha(x, z) = \frac{EI_0(x, z)}{\lambda(z)}. \quad (12)$$

For the high-fidelity model, $EI_0(x, 0)$ is the same as EI as in equation (9) because x' (in equation (10)) is equal to x (at least for a GP or SaasBO surrogate model). However, for low-fidelity queries, $EI_0(x, 1)$ differs from EI. It should not be the improvement at the point $(x, z=1)$, because our goal is to find the maximum for $z=0$. Rather, $EI_0(x, 1)$ is more similar to the KG acquisition function which asks how does the expected maximum of f change, even if the new maximum might not be at the query’s point x .



A complete calculation of equation (10) would require a search over x' inside the expectation, which is too computationally expensive. Therefore, we make the assumption that if adding the queried low-fidelity function evaluation at point x causes the surrogate model to find an improved high-fidelity maximum, that maximum will also be at the point x . Thus, we assume that $x' = x$ in equation (10).

For a GP surrogate, the joint distribution over $f(x, 0)$ and $f(x, 1)$ conditioned on the previous observations is a normal distribution that can be found by formulating the joint distribution over all of the queried points, $(x, 0)$, and $(x, 1)$ (see equation (4)) and then conditioning on the calculated values at the queried points. We denote the result as

$$\begin{bmatrix} f(x, 0) \\ f(x, 1) \end{bmatrix} | D_t \sim \mathcal{N} \left(\begin{bmatrix} \mu_t(x, 0) \\ \mu_t(x, 1) \end{bmatrix}, \begin{bmatrix} \sigma_t^2(x, 0) & c_t(x, 0; x, 1) \\ c_t(x, 0; x, 1) & \sigma_t^2(x, 1) \end{bmatrix} \right), \quad (13)$$

where, conditioned on the data available at the t th iteration, $\mu_t(x, z)$ is the conditional mean of $f(x, z)$ under the surrogate model, $\sigma_t^2(x, z)$ is the conditional variance of $f(x, z)$, and $c_t(x, z; x', z')$ is the conditional covariance between $f(x, z)$ and $f(x', z')$.

Thus, by properties of the normal distribution, the variance of $f(x, 0)$ conditioned on the data and the value of $f(x, 1)$ would be $\frac{c^2(x, 0; x, 1)}{\sigma^2(x, 0)}$, regardless of the value of $f(x, 1)$. In expectation, $\mu_t(x, 0)$ will not change based on the new observation. Therefore, the EI at $(x, 0)$ from observing $(x, 1)$ is $g(y^+, \mu_t(x, 0), \frac{c_t(x, 0; x, 1)}{\sigma_t(x, 0)})$ and thus our entire formula for the EI at $z = 0$ from observing (x, z) is

$$\text{EI}_0(x, z) = \begin{cases} g(y^+, \mu_t(x, 0), \sigma_t(x, 0)) & \text{if } z = 0, \\ g(y^+, \mu_t(x, 0), \frac{c_t(x, 0; x, 1)}{\sigma_t(x, 0)}) & \text{if } z = 1. \end{cases} \quad (14)$$

Figure 1 graphically illustrates the calculation. When considering a high-fidelity point, EI_0 is the EI from the distribution, shown in blue. When considering a low-fidelity point, it is the expected improvement from the

distribution shown in orange. These EIs are divided by the query cost of the associated fidelity to determine the value of the acquisition function at this x - z pair.

Equation (14) is the EI in the high-fidelity model for a GP surrogate. To use the SaasBO surrogate, we must take the expectation of this formula over settings of the GP parameters. Just as in equation (9), using MCMC sampling to produce L posterior samples over possible GPs, the final EI for multi-fidelity SaasBO is

$$\text{EI}_0(x, z) = \begin{cases} \frac{1}{L} \sum_{l=1}^L g(x, y^+, \mu_t^l(x, 0), \sigma_t^l(x, 0)) & \text{if } z = 0, \\ \frac{1}{L} \sum_{l=1}^L g\left(x, y^+, \mu_t^l(x, 0), \frac{c_t^l(x, 0; x, 1)}{\sigma_t^l(x, 0)}\right) & \text{if } z = 1. \end{cases} \quad (15)$$

Therefore the final algorithm of the multi-fidelity version of SaasBO is as follows, given a budget of T iterations:

- (i) Select an initial set of m random points to query: $\{(x_1, z_1), (x_2, z_2), \dots, (x_m, z_m)\}$, with some in the high fidelity ($z = 0$) and some in the low fidelity ($z = 1$).
- (ii) Evaluate these initial m points: $y_i = f(x_i, z_i)$ for $i \in 1, 2, \dots, m$, and let $D_{m+1} = \{(x_1, z_1, y_1), (x_2, z_2, y_2), \dots, (x_m, z_m, y_m)\}$.
- (iii) For t from $m + 1$ to T
 - a. Sample L GP parameters from a SaasGP surrogate posterior distribution conditioned on D_t .
 - b. Find best point so far: $y^+ = \max_{i|z_i=0} y_i$.
 - c. Maximize the acquisition function: $x, z = \arg \max_{x, z} \frac{\text{EI}_0(x, z)}{\lambda(z)}$.
 - d. Evaluate the function: $y = f(x, z)$.
 - e. Add observation to data: $D_{t+1} = D_t \cup \{(x, z, y)\}$.
- (iv) Find best point: $i^+ = \arg \max_{i|z_i=0} y_i$.
- (v) Return (x_{i^+}, y_{i^+}) .

4. Computational experiments

We applied these methods to the optimization of a time-varying electric field to control a series of one-dimensional quantum systems [24, 33] discussed previously in section 2.1. The simulation propagates an initial wavefunction $\psi_0(x)$ to a propagated state $\psi_{N-1}(x)$ after $N - 1$ iterations of a split-operator propagator after discretizing both space and time on a uniform grid. Our goal is to maximize P (given by equation (2)), which is the inner product (overlap) of our propagated state $\psi_{N-1}(x)$ with a provided target final state, $\psi_f(x)$. To optimize the electric field, $E(t)$, we let $E(t)$ be a superposition of a fixed set of frequencies with variable amplitudes and phases:

$$E(t) = \sum_i a_i \sin(2\pi f_i t + \phi_i) \quad (16)$$

where $\{f_i\}_{i=1}^{11}$ are fixed and we optimize over 22 parameters: $\{a_i\}_{i=1}^{11}$ and $\{\phi_i\}_{i=1}^{11}$. The set of frequencies are centered on the theoretical ideal frequency associated with the energetic difference between the start and target states. The two parameters, amplitude and phase, are the input X of the objective function. Multi-fidelity is incorporated by changing the system's temporal resolution (10 000 time-steps for low fidelity and 30 000 for high fidelity) for our proposed optimization method. All other methods make use of the high-fidelity temporal resolution.

We tested our methods on a large set of optimization problems, each with a unique background potential. This set of problems was separated into three levels of difficulty, depending on the maximum transition probability P obtained by a gradient-based NIC-CAGE algorithm [33] (modified to use a split-operator propagator) after 60 iterations. An optimization problem was considered to be 'easy' if $P > 0.99$, 'medium' if $0.01 < P < 0.99$, and 'hard' if P did not exceed 0.01 in 60 iterations. The full problem set consisted of 7000 easy problems, 2497 medium problems and 1305 hard problems, consistent with those found in [24]. Note that NIC-CAGE optimizes a general electric field (in comparison with our experiment which considers a fixed set of frequencies). However, this classification is helpful to identify generally easier or harder problems and stratify the results accordingly.

For each of the three sets of problems, we carried out experiments with eight different algorithms: Nelder–Mead (NM) [73, 74], conjugate-gradient (CG) [75], Newton-CG [75], truncated-Newton [76], Broyden–Fletcher–Goldfarb–Shanno (BFGS) [75], standard BO, SaasBO [66], and MFSaasBO.

The BO methods (BO, SaasBO, and MFSaasBO) and NM do not require gradients and therefore require less computational time per evaluation. NM reflects, stretches, or shrinks a simplex of solutions by evaluating the function at the vertices of the simplex. This direct-search method works well, but convergence is not guaranteed.

The gradient-based methods all performed very similarly, so in the results below, we show only the best-performing method: CG. CG overcomes the drawback of steepest descent [77] by moving in the conjugate direction for faster convergence. This method is quadratically convergent. It considers the next search direction by linearly combining the gradient and the previous direction in such a way so that the new direction is conjugate to the previous search direction.

All eight methods were tested with a budget of 50 iterations. After every evaluation of the forward-propagated initial state, we recorded the overall compute time, compute time spent on simulation, and P , the maximum transition probability achieved over the duration of the control pulse. We let P^* be the best value of P found up to any particular iteration. We report the final P^* after all 50 iterations, P^* as a function of total computation time, and P^* as a function of the total simulation time.

5. Results and discussion

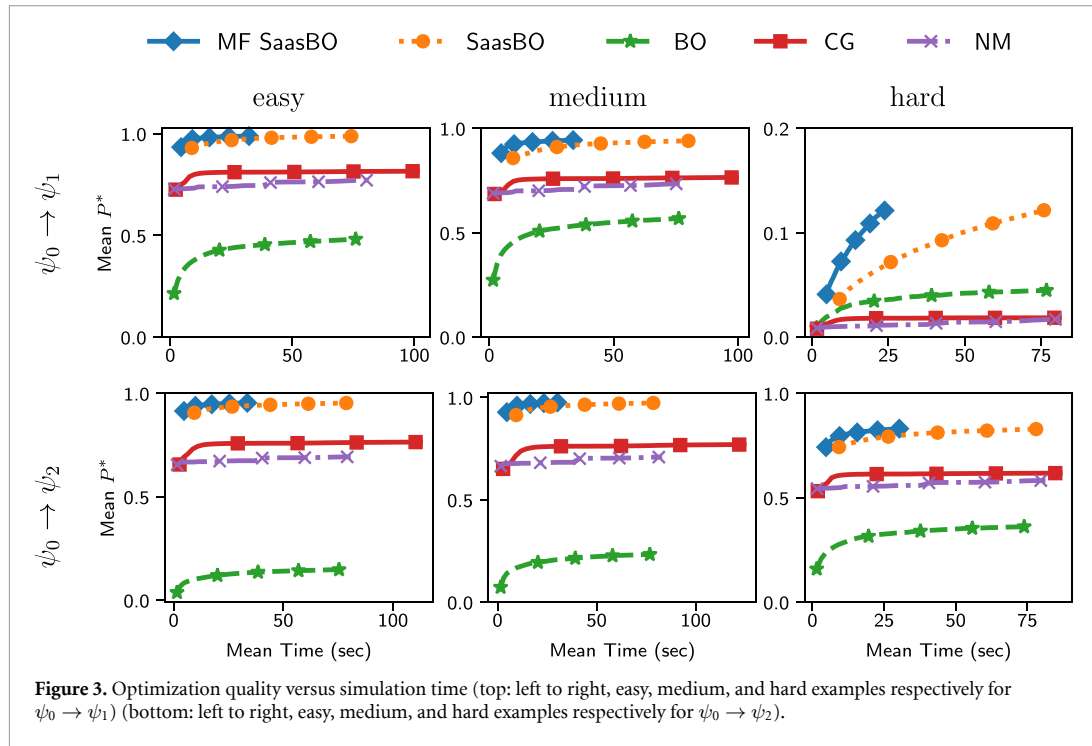
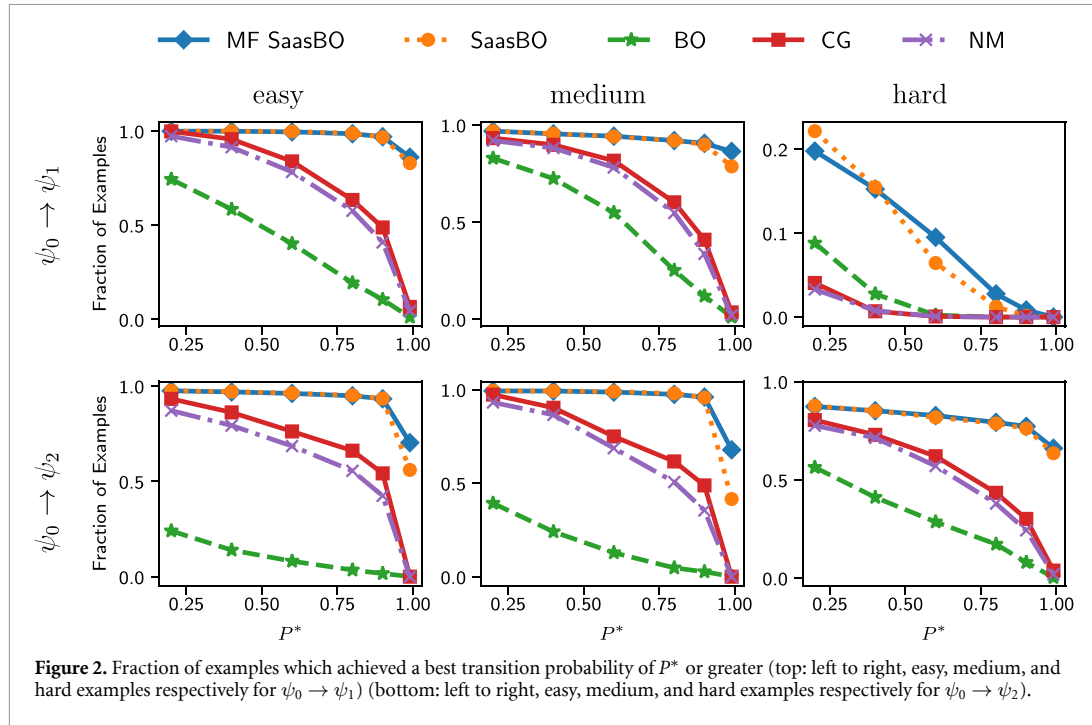
The main objective of our work was to reduce the compute time required to find some time-dependent control-field, $E(t)$, that maximizes the probability P that a transition from state ψ_0 to ψ_f occurs. To test the capability of our BO routine, we explored two different scenarios: the transition from the ground to the first excited state ($\psi_0 \rightarrow \psi_1$), and the transition from the ground to the second excited state ($\psi_0 \rightarrow \psi_2$).

Figure 2 shows what fraction of the tested problems were able to be optimized to a given quality as a function of that quality, P^* , after 50 iterations. For instance, for $\psi_0 \rightarrow \psi_1$ transitions that can be classified as ‘easy’ examples for the CG method, approximately 50% of the problems were optimized to a value of $P \geq 0.9$. From figure 2 it is clear that the proposed method achieves the highest accuracy in the majority of the examples, whereas the performance of standard BO is poor. The performance of CG and NM are quite similar to one another, but still perform worse than SaasBO and MFSaasBO. The number of examples for which the CG and NM methods reach $P^* \geq 0.99$ are fewer in comparison to SaasBO and MFSaasBO. The performance of all methods in the ($\psi_0 \rightarrow \psi_1$) case is significantly worse in the case of the hard problem. It is notable that for all cases, MFSaasBO outperforms SaasBO in the fraction of examples that reached $P^* > 0.99$.

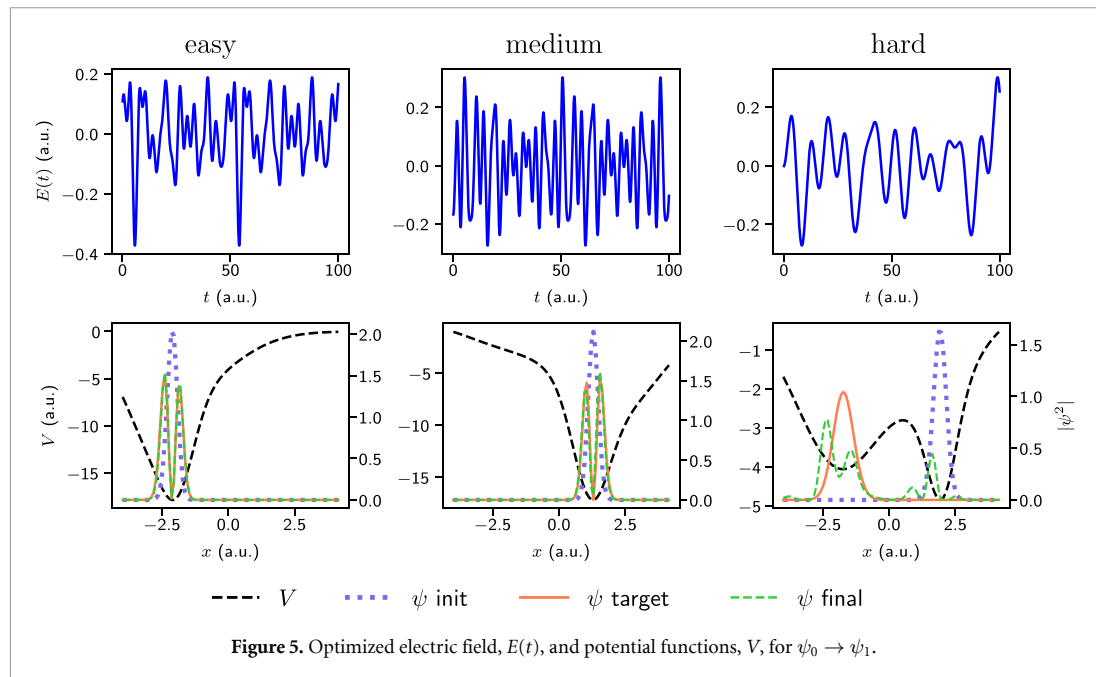
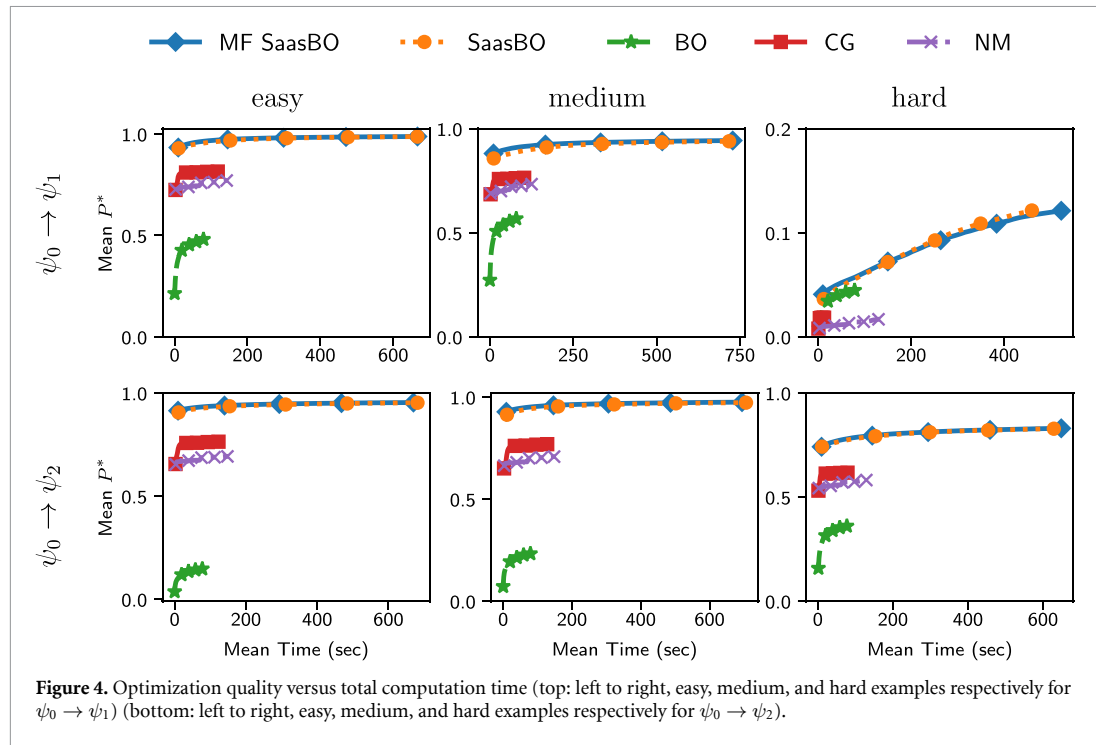
Figure 3 shows the average (across different potential functions) best value found as a function of the total amount of computation time spent in simulation. For both the ($\psi_0 \rightarrow \psi_1$) and ($\psi_0 \rightarrow \psi_2$) case, and for all easy, medium, and hard problem sets, MFSaasBO distinctly beats all other methods, requiring less simulation time. Though equal iterations cost equal function evaluation for all methods, MFSaasBO has the advantage that a low-fidelity evaluation takes less time. It is interesting to note that SaasBO, BO, and NM methods take almost similar simulation times as they all have equal number of high fidelity evaluations.

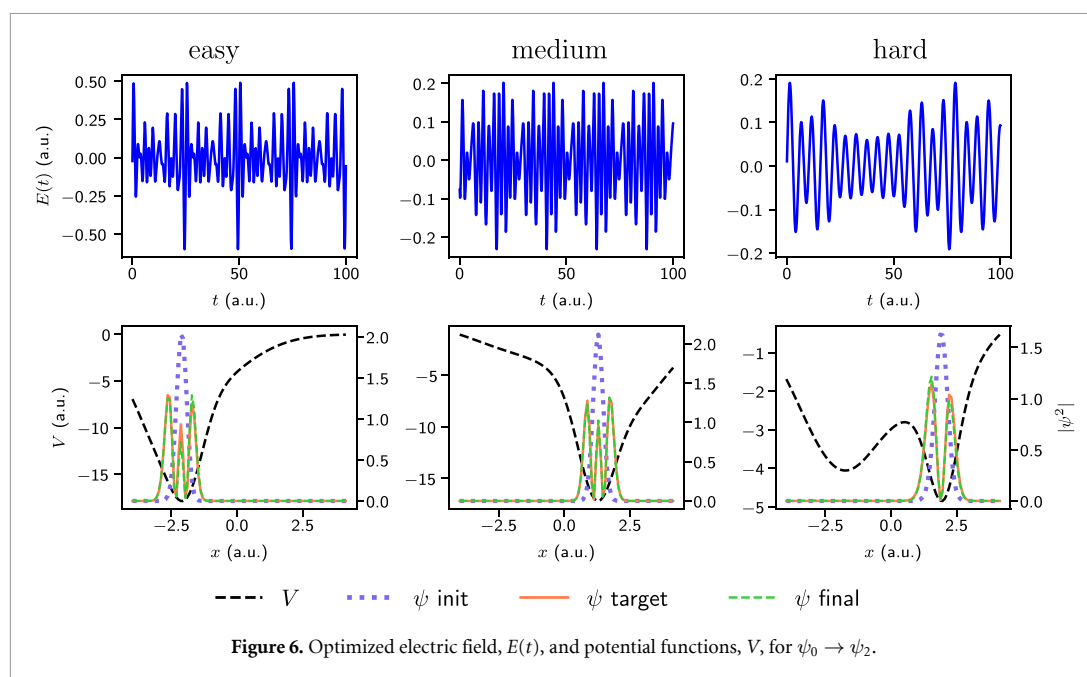
CG methods take more simulation time because it is a gradient method and, therefore, requires additional time for the gradient calculations. While CG does slightly better than NM in figure 2, in terms of simulation time NM is better. Both of these methods quickly get stuck in local minima. NIC-CAGE was able to do better using a gradient method for the ‘easy’ problems because it optimizes in the full space of (discretized) electric field functions. For these experiments, we are restricted to electric fields of the form of equation (16). For ‘hard’ problems, neither NIC-CAGE nor gradient methods in this reduced function space are able to achieve transition probabilities better than 0.01, whereas our method is able to optimize most of these problems to at least a level of 0.1 (and some to 0.2).

Figure 4 shows the average quality of the found solution as a function of the total computation time (both simulation time and time spent selecting the next query point). This shows that SaasBO and MFSaasBO take significantly more total time than other methods, although in accuracy these two methods are the best in a good margin. This is due to the additional time necessary to decide which point to next query. The simulations for our one-dimensional quantum system are relatively fast. With higher-dimensional quantum simulations (such time-dependent density functional theory calculations on material systems), we expect the simulation time will dominate all other aspects of the optimization (which do not scale with the complexity of the simulation), and high-dimensional multi-fidelity BO methods will not only produce better optima, but also produce them faster.



Figures 5 and 6 show examples of the electric fields found (top three plots for both figures) and corresponding double bell potential function. We have used same set of examples for both transition cases ($\psi_0 \rightarrow \psi_1$ and $\psi_0 \rightarrow \psi_2$). From both figures it is clear that our known initial state (blue) is completely distinct from our desired or target state (orange) and except for one case ($\psi_0 \rightarrow \psi_2$, hard set), our final propagated wave function i.e. final state (green) closely matches the target state (orange).





6. Conclusions

We have presented a high-dimensional multi-fidelity BO approach for successful quantum control of time-dependent systems. This approach is particularly efficient and well-suited for the quantum control problem as it considers all previously evaluated points (instead of just the single previous one or its gradient) when deciding which point to query next. By employing a recent high-dimensional BO surrogate model (SaasBO) and combining it with our multi-fidelity acquisition function, we are able to drastically increase the quality of the found solutions for a fixed simulation computational budget.

BO methods require more computational cost between function evaluations than other optimization methods. However, as the complexity of the simulations necessary for the evaluations grows, the extra computational burden of BO relative to gradient methods becomes negligible. For these reasons, we anticipate that our BO approach would be even more efficient for higher-dimensional quantum systems (or systems with many-body quantum interactions), which would intrinsically require more function evaluations. As such, we advocate the use of BO in optimal quantum control and other time-dependent inverse problems that have a heavy simulation cost.

Data availability statement

No new data were created or analysed in this study.

Acknowledgments

This work was supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, Scientific Discovery through Advanced Computing (SciDAC) program under Award Number DE-SC0022209.

ORCID iDs

Marjuka F Lazin  <https://orcid.org/0009-0008-1746-9047>

Christian R Shelton  <https://orcid.org/0000-0001-6698-7838>

Bryan M Wong  <https://orcid.org/0000-0002-3477-8043>

References

- [1] Huang G M, Tarn T J and Clark J W 1983 *J. Math. Phys.* **24** 2608–18
- [2] Oviedo M B and Wong B M 2016 *J. Chem. Theory Comput.* **12** 1862–71

- [3] Ilawe N V, Oviedo M B and Wong B M 2017 *J. Chem. Theory Comput.* **13** 3442–54
- [4] Ilawe N V, Oviedo M B and Wong B M 2018 *J. Mater. Chem. C* **6** 5857–64
- [5] Maiuri M, Oviedo M B, Dean J C, Bishop M, Kudisch B, Toa Z S D, Wong B M, McGill S A and Scholes G D 2018 *J. Phys. Chem. Lett.* **9** 5548–54
- [6] Kudisch B, Maiuri M, Moretti L, Oviedo M B, Wang L, Oblinsky D G, Prud'homme R K, Wong B M, McGill S A and Scholes G D 2020 *Proc. Natl Acad. Sci. USA* **117** 11289–98
- [7] Nowack K C, Koppens F H L, Nazarov Y V and Vandersypen L M K 2007 *Science* **318** 1430–3
- [8] Kues M et al 2017 *Nature* **546** 622–6
- [9] Fortunato E M, Pravia M A, Boulant N, Teklemariam G, Havel T F and Cory D G 2002 *J. Chem. Phys.* **116** 7599–606
- [10] Williams H J, Caldwell L, Fitch N J, Truppe S, Rodewald J, Hinds E A, Sauer B E and Tarbutt M R 2018 *Phys. Rev. Lett.* **120** 163201
- [11] Bartana A, Kosloff R and Tannor D J 2001 *Chem. Phys.* **267** 195–207
- [12] Brown B L, Dicks A J and Walmsley I A 2006 *Phys. Rev. Lett.* **96** 173002
- [13] Wright M J, Pechkis J A, Carini J L, Kallush S, Kosloff R and Gould P L 2007 *Phys. Rev. A* **75** 051401
- [14] Brumer P and Shapiro M 1989 *Acc. Chem. Res.* **22** 407–13
- [15] Somló J, Kazakov V A and Tannor D J 1993 *Chem. Phys.* **172** 85–98
- [16] Zhu W, Botina J and Rabitz H 1998 *J. Chem. Phys.* **108** 1953–63
- [17] Khaneja N, Reiss T, Kehlet C, Schulte-Herbrüggen T and Glaser S J 2005 *J. Magn. Reson.* **172** 296–305
- [18] James M R, Nurdin H I and Petersen I R 2007 H_∞ control of linear quantum stochastic systems *Quantum-Atom Optics Downunder* (Optica Publishing Group) p QME32
- [19] Frey V, Norris L M, Viola L and Biercuk M J 2020 *Phys. Rev. Appl.* **14** 024021
- [20] Youssry A, Paz-Silva G A and Ferrie C 2020 *npj Quantum Inf.* **6** 95
- [21] Dong D and Petersen I 2010 *IET Control Theory Appl.* **4** 2651–71
- [22] Zhu W and Rabitz H 1998 *J. Chem. Phys.* **109** 385–91
- [23] Wang X, Kumar A, Shelton C R and Wong B M 2020 *Phys. Chem. Chem. Phys.* **22** 22889–99
- [24] Gao Y, Wang X, Yu N and Wong B M 2022 *Phys. Chem. Chem. Phys.* **24** 24012–20
- [25] Duffield S, Benedetti M and Rosenkranz M 2023 *Mach. Learn.: Sci. Technol.* **4** 025007
- [26] Tamiya S and Yamasaki H 2022 *npj Quantum Inf.* **8** 90
- [27] Tian J, Said R S, Jelezko F, Cai J and Xiao L 2023 *Sensors* **23** 3244
- [28] Sauvage F and Mintert F 2020 *PRX Quantum* **1** 020322
- [29] Hoff P V D, Thallmair S, Kowalewski M, Siemering R and Vivie-Riedle R D 2012 *Phys. Chem. Chem. Phys.* **14** 14460–85
- [30] Thallmair S, Keefer D, Rott F and de Vivie-Riedle R 2017 *J. Phys. B: At. Mol. Opt. Phys.* **50** 082001
- [31] Brixner T and Gerber G 2003 *ChemPhysChem* **4** 418–38
- [32] Dantus M and Lozovoy V V 2004 *Chem. Rev.* **104** 1813–60
- [33] Raza A, Hong C, Wang X, Kumar A, Shelton C and Wong B 2020 *Comput. Phys. Commun.* **258** 107541
- [34] Wong B M and Raman S 2007 *J. Comput. Chem.* **28** 759–66
- [35] Wong B M, Fadri M M and Raman S 2008 *J. Comput. Chem.* **29** 481–7
- [36] Bechtel H A, Steeves A H, Wong B M and Field R W 2008 *Angew. Chem., Int. Ed.* **47** 2969–72
- [37] Wong B M 2008 *Phys. Chem. Chem. Phys.* **10** 5599–606
- [38] Wong B M, Thom R L and Field R W 2006 *J. Phys. Chem. A* **110** 7406–13
- [39] Reinisch G, Miki K, Vignoles G L, Wong B M and Simmons C S 2012 *J. Chem. Theory Comput.* **8** 2713–24
- [40] Wong B M, Steeves A H and Field R W 2006 *J. Phys. Chem. B* **110** 18912–20
- [41] Prozument K, Shaver R G, Ciuba M A, Muentert J S, Park G B, Stanton J F, Guo H, Wong B M, Perry D S and Field R W 2013 *Faraday Discuss.* **163** 33–57
- [42] Mockus J, Tiesis V and Zilinskas A 1978 *Towards Global Optimization* vol 2 (Elsevier)
- [43] Garnett R 2023 *Bayesian Optimization* (Cambridge University Press)
- [44] Rasmussen C E 2004 Gaussian processes in machine learning *Advanced Lectures on Machine Learning (ML 2003) (Revised Lectures)* ed O Bousquet, U von Luxburg and G Rätsch (Springer) pp 63–71
- [45] Rasmussen C E and Williams C K I 2006 *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)* (MIT Press)
- [46] Jones D, Schonlau M and Welch W 1998 *J. Glob. Optim.* **13** 455–92
- [47] Kushner H J 1964 *J. Basic Eng.* **86** 97–106
- [48] Srinivas N, Krause A, Kakade S M and Seeger M W 2012 *IEEE Trans. Inf. Theory* **58** 3250–65
- [49] Wu J, Toscano-Palmerin S, Frazier P I and Wilson A G 2019 Practical multi-fidelity Bayesian optimization for hyperparameter tuning (arXiv:1903.04703)
- [50] Frazier P, Powell W and Dayanik S 2009 *INFORMS J. Comput.* **21** 599–613
- [51] Mockus J and Mockus L 1991 *J. Optim. Theory Appl.* **70** 157–72
- [52] Garnett R, Osborne M A and Hennig P 2013 Active learning of linear embeddings for Gaussian processes (arXiv:1310.6740)
- [53] Wang Z, Hutter F, Zoghi M, Matheson D and de Freitas N 2013 Bayesian optimization in a billion dimensions via random embeddings (arXiv:1301.1942)
- [54] Nayeibi A, Munteanu A and Poloczek M 2019 A framework for Bayesian optimization in embedded subspaces *Proc. 36th Int. Conf. on Machine Learning (Proc. Machine Learning Research vol 97)* ed K Chaudhuri and R Salakhutdinov (PMLR) pp 4752–61
- [55] Kirschner J, Mutný M, Hiller N, Ischebeck R and Krause A 2019 Adaptive and safe Bayesian optimization in high dimensions via one-dimensional subspaces *CoRR* (arXiv:1902.03229)
- [56] Moriconi R, Deisenroth M P and Kumar K S S 2020 High-dimensional Bayesian optimization using low-dimensional feature spaces (arXiv:1902.10675)
- [57] Kandasamy K, Schneider J and Poczos B 2015 High dimensional Bayesian optimisation and bandits via additive models *Proc. 32nd Int. Conf. on Machine Learning (Proc. Machine Learning Research vol 37)* ed F Bach and D Blei (PMLR) pp 295–304
- [58] Gardner J, Guo C, Weinberger K, Garnett R and Grosse R 2017 Discovering and Exploiting Additive Structure for Bayesian Optimization *Proc. 20th Int. Conf. on Artificial Intelligence and Statistics (Proc. Machine Learning Research vol 54)* ed A Singh and J Zhu (PMLR) pp 1311–9
- [59] Wang Z, Gehring C, Kohli P and Jegelka S 2018 Batched large-scale Bayesian optimization in high-dimensional spaces *Proc. 21st Int. Conf. on Artificial Intelligence and Statistics (Proc. Machine Learning Research vol 84)* ed A Storkey and F Perez-Cruz (PMLR) pp 745–54

- [60] Mutny M and Krause A 2018 Efficient high dimensional Bayesian optimization with additivity and quadrature fourier features *Advances in Neural Information Processing Systems* vol 31, ed S Bengio, H Wallach, H Larochelle, K Grauman, N Cesa-Bianchi and R Garnett (Curran Associates, Inc.)
- [61] Rolland P, Scarlett J, Bogunovic I and Cevher V 2018 High-dimensional Bayesian optimization via additive models with overlapping groups *CoRR* (arXiv:1802.07028)
- [62] Oh C, Gavves E and Welling M 2018 BOCK : Bayesian optimization with cylindrical kernels *Proc. 35th Int. Conf. on Machine Learning (Proc. Machine Learning Research* vol 80) ed J Dy and A Krause (PMLR) pp 3868–77
- [63] Li C, Gupta S, Rana S, Nguyen V, Venkatesh S and Shilton A 2018 High dimensional Bayesian optimization using dropout (arXiv:1802.05400)
- [64] Jaquier N and Rozo L 2020 High-dimensional Bayesian optimization via nested riemannian manifolds (arXiv:2010.10904)
- [65] Eriksson D, Pearce M, Gardner J, Turner R D and Poloczek M 2019 Scalable global optimization via local Bayesian optimization *Advances in Neural Information Processing Systems* vol 32, ed H Wallach, H Larochelle, A Beygelzimer, F d'Alché-Buc, E Fox and R Garnett (Curran Associates, Inc.)
- [66] Eriksson D and Jankowiak M 2021 High-dimensional Bayesian optimization with sparse axis-aligned subspaces (arXiv:2103.00349)
- [67] Hoffman M D and Gelman A 2014 *J. Mach. Learn. Res.* **15** 1593–623
- [68] Byrd R H, Lu P, Nocedal J and Zhu C 1995 *SIAM J. Sci. Comput.* **16** 1190–208
- [69] Li S, Kirby R and Zhe S 2021 Batch multi-fidelity Bayesian optimization with deep auto-regressive networks *Advances in Neural Information Processing Systems* ed A Beygelzimer, Y Dauphin, P Liang and J W Vaughan
- [70] Kandasamy K, Dasarathy G, Oliva J B, Schneider J and Poczos B 2016 Gaussian process bandit optimisation with multi-fidelity evaluations *Advances in Neural Information Processing Systems* vol 29, ed D Lee, M Sugiyama, U Luxburg, I Guyon and R Garnett (Curran Associates, Inc.)
- [71] Song J, Chen Y and Yue Y 2018 A general framework for multi-fidelity Bayesian optimization with Gaussian processes (arXiv:1811.00755)
- [72] Li S, Xing W W, Kirby M and Zhe S 2020 Multi-fidelity Bayesian optimization via deep neural networks *CoRR* (arXiv:2007.03117)
- [73] Nelder J A and Mead R 1965 *Comput. J.* **7** 308–13
- [74] Gao F and Han L 2012 *Comput. Optim. Appl.* **51** 259–77
- [75] Jorge Nocedal S J W 2006 *Numerical Optimization* 2nd edn (Springer)
- [76] Dembo R S and Steihaug T 1983 *Math. Program.* **26** 190–212
- [77] Bartholomew-Biggs M 2005 The steepest descent method *Nonlinear Optimization with Financial Applications* (Springer) pp 51–64