

A QUANTUM OPTIMIZATION FRAMEWORK FOR DATA–ASSIMILATION–AUGMENTED PARAMETER ESTIMATION*

MUHAMMAD JALIL AHMAD[†], MOHAMMADHOSSEIN MOHAMMADISIAHROUDI[‡],
ANIMIKH BISWAS[§], AND KATHLEEN HOFFMAN[§]

Abstract. Parameter estimation is a fundamental challenge in the calibration of ordinary differential equation (ODE) models, where repeated numerical integration can lead to high computational cost. In this work, we investigate whether quantum algorithms can be leveraged to assist parameter estimation in nonlinear dynamical systems. We develop a hybrid classical–quantum framework that reformulates a data-assimilation-augmented parameter estimation problem as a combinatorial optimization task. Model dynamics and data assimilation are enforced entirely on the classical side, while the resulting parameter estimation cost functional is discretized and approximated by a quadratic unconstrained binary optimization (QUBO) surrogate. This surrogate is mapped to an Ising Hamiltonian, and quantum optimizers are used to search for low-energy configurations corresponding to candidate parameter estimates. We apply the framework to SIS and SIR epidemic models, the chaotic Lorenz–63 system, and a high-dimensional two-layer Lorenz–96 system. In this setting, the method is used to recover classical system parameters from partial state observations across steady-state, chaotic, and high-dimensional multiscale dynamical systems. Numerical experiments with synthetic data show that the proposed approach accurately recovers parameters while requiring data-assimilation solves only on a prescribed coarse grid. The framework avoids quantum state tomography, illustrating a viable pathway for integrating quantum optimization into data-driven parameter estimation for nonlinear dynamical systems.

Key words. Quantum Optimization, Data Assimilation, Parameter Estimation, Nonlinear Dynamical Systems, QUBO

MSC codes. 34D06, 34A55, 65P99, 65K10, 68Q12

1. Introduction. Ordinary differential equation (ODE) models play a central role in the mathematical modeling of time-dependent processes across the natural and applied sciences. In epidemiology, compartmental models such as the susceptible–infected–susceptible (SIS) and susceptible–infected–recovered (SIR) systems provide interpretable descriptions of disease transmission, recovery, and population-level progression through epidemiological states [1, 2]. In atmospheric science and nonlinear dynamics, the Lorenz–63 and Lorenz–96 systems are standard benchmark models for studying predictability, chaos, and multiscale dynamical behavior [3, 4, 5]. These models are widely used because their parameters often have direct physical, biological, or epidemiological interpretations, and because their trajectories can be compared with observational data.

The practical use of such models is critically dependent on the accurate estimation of unknown parameters. Even when the governing equations are known, uncertainty in parameters can lead to substantial errors in predicted dynamics. In epidemiological models, transmission and recovery rates determine the timing, magnitude, and duration of outbreaks, and inaccurate parameter values may lead to unreliable forecasts or misleading assessments of intervention strategies [6, 7]. In chaotic systems, the challenge is even more pronounced: small errors in parameters or initial conditions can grow rapidly over time, producing trajectories that diverge from the observed

*Submitted to the editors 07.27.2026.

[†] Department of Mathematics and Statistics, University of Maryland Baltimore County, USA (LS47576@umbc.edu).

[‡] Department of Mathematics and Statistics & Quantum Science Institute, University of Maryland Baltimore County, USA.

[§] Department of Mathematics and Statistics, University of Maryland Baltimore County, USA.

dynamics even when the underlying model is deterministic [3, 5]. Parameter estimation is therefore a fundamental component of forward prediction and uncertainty quantification.

Parameter estimation in dynamical systems is naturally formulated as an inverse problem. Given a collection of observations, one seeks parameter values that produce model trajectories consistent with the available data. A common approach is to pose this task as an optimization problem constrained by governing differential equations. In particular, the unknown parameters are determined by minimizing a data-misfit cost functional subject to the ODE dynamics. This leads to an ODE-constrained optimization problem in which each evaluation of the cost functional requires the numerical integration of the underlying dynamical system. Since the observable-to-parameter map is often nonlinear and nonconvex, classical solution methods may require many repeated forward simulations before convergence. The resulting computational burden can become significant when the parameter space is high-dimensional, when observations are partial or noisy, or when the underlying dynamics exhibit sensitive dependence on parameters and initial conditions. Consequently, the development of efficient computational strategies for parameter estimation remains an important challenge in scientific computing and inverse problems [8, 9, 10].

Recent advances in quantum computing have motivated the exploration of quantum algorithms as alternative computational tools for challenging optimization tasks. In particular, hybrid quantum–classical methods seek to exploit the strengths of both paradigms by combining conventional numerical computation with quantum optimization subroutines. Within this context, the present work investigates whether quantum optimization techniques can assist parameter estimation in nonlinear dynamical systems. The goal is not to replace classical time integration, but to reformulate the parameter-search component in a form that can be treated by quantum-compatible optimization methods. This provides a pathway for incorporating quantum optimization into inverse problems governed by differential equations while remaining compatible with quantum devices [11, 12, 13].

1.1. Related Work.

Conventional Parameter Estimation. Conventional parameter estimation methods include nonlinear least-squares [6], Bayesian [14, 15, 16], data-assimilation [17, 18, 19, 20], and machine-learning-based approaches [21, 22, 23, 24, 25]. These methods have been applied successfully to many dynamical systems, but their performance can be affected by limited or partial observations and by the computational expense of repeated model integration.

Data assimilation provides a natural framework for incorporating observational data into dynamical models [26, 27, 28, 29]. These methods combine model dynamics with available observations to improve state reconstruction and parameter estimation, including in systems with partial or uncertain data [18, 20, 17, 30, 19]. In parameter estimation, data assimilation can be used to construct an observation-informed cost functional that measures the agreement between model-generated trajectories and the available data. This provides a stable basis for estimating unknown parameters, particularly when the initial state is uncertain or only a subset of the system variables is observed. A useful feature of this formulation is that the nudged system need not be initialized at the true initial condition; the feedback term can stabilize the observed components and produce a meaningful cost functional landscape even when the initial state is uncertain [19].

Quantum Differential Equation Solvers. Quantum algorithms for differential equations have attracted increasing attention because many scientific and engineering applications are governed by dynamical systems. Several approaches have been proposed for solving ordinary and partial differential equations on quantum computers. One major direction relies on quantum linear algebra, where differential equations are discretized and reformulated as linear systems that can be solved using quantum linear system algorithms (QLSAs) [31, 32]. Another line of work studies nonlinear differential equations through linear representations and Carleman-type embeddings, enabling quantum algorithms for selected classes of nonlinear systems [33, 34]. More recently, Schrödingerization techniques have been introduced to transform differential equations into Hamiltonian evolution problems that are amenable to quantum simulation [35]. Variational and time-marching approaches have also been explored, where hybrid quantum–classical algorithms propagate the solution through a sequence of optimization steps using parameterized quantum circuits [36].

Despite these advances, directly solving nonlinear ODEs on quantum hardware remains challenging. Existing methods often require restrictive assumptions on sparsity, block encodings, state preparation, measurement access, smoothness of the dynamics, or the availability of sufficiently deep quantum circuits [33, 34]. Furthermore, parameter estimation problems typically require repeated evaluations of the underlying dynamical system for many candidate parameter values, which can significantly amplify the computational cost of quantum simulation. These limitations make direct quantum time integration difficult for the nonlinear and partially observed systems considered in this work. Consequently, rather than using quantum resources to solve the differential equations themselves, we employ classical simulation and data assimilation to construct the parameter-estimation cost functional and use quantum optimization only for the resulting combinatorial search problem. This strategy avoids the challenges associated with quantum simulation of nonlinear dynamics while remaining compatible with quantum optimization methods.

Quantum Optimization. Recent developments in quantum computing have motivated the study of quantum and hybrid quantum–classical algorithms for optimization, inverse problems, and scientific computing. Quantum systems provide computational mechanisms that differ fundamentally from classical digital computation. Superposition allows an n -qubit register to represent amplitudes over 2^n computational basis states, entanglement provides nonclassical correlations among qubits, and tunneling effects have motivated quantum optimization strategies for exploring complicated energy landscapes [37, 38, 39]. These features do not imply an automatic speedup for every problem, but they provide a foundation for designing quantum algorithms that may offer advantages for selected structured computational tasks.

Among quantum algorithms, the Quantum Approximate Optimization Algorithm (QAOA) has received substantial attention as a hybrid variational method for combinatorial optimization [40]. QAOA approximately minimizes a cost functional encoded as an Ising Hamiltonian by alternating between a problem Hamiltonian and a mixing Hamiltonian, with variational parameters optimized by a classical outer loop. Since many combinatorial optimization problems can be written as quadratic unconstrained binary optimization (QUBO) problems and mapped to equivalent Ising Hamiltonians, QUBO provides a natural bridge between classical binary optimization and QAOA-based quantum optimization [41]. This makes QAOA relevant for parameter estimation only after the original continuous inverse problem has been reformulated into a finite-dimensional binary optimization problem.

Quantum annealing provides another optimization paradigm for QUBO and Ising-

type problems. In quantum annealing, the optimization problem is encoded into an energy landscape, and the algorithm seeks low-energy configurations by evolving the system from an initial Hamiltonian to a problem Hamiltonian. This approach is especially natural for QUBO formulations because binary variables can be mapped directly to Ising spins, and low-energy spin configurations correspond to candidate minimizers of the original binary cost functional [38, 39]. In this work, we include simulated quantum annealing (SQA) as an additional QUBO solver, allowing comparison with the QAOA-based IBM simulator and IBM Kingston QPU implementations.

Beyond QAOA, several other quantum optimization paradigms have been proposed for continuous and structured optimization problems arising in scientific computing. Quantum Gibbs-sampling methods prepare thermal states of suitably designed Hamiltonians and have been studied for optimization, machine learning, and probabilistic inference, where low-energy states correspond to high-quality solutions. In addition, Quantum Hamiltonian Descent (QHD) has recently been proposed as a quantum counterpart of classical gradient-based optimization [42]. Derived from the path-integral formulation of dynamical systems associated with the continuous-time limit of gradient descent, QHD describes optimization as a quantum Hamiltonian evolution and exploits quantum-mechanical effects such as tunneling to explore nonconvex landscapes. The resulting dynamics can be implemented on both digital and analog quantum platforms, and empirical studies have demonstrated promising performance on nonconvex optimization problems [42]. More recently, theoretical and computational developments have extended QHD to non-smooth optimization problems and software frameworks for practical deployment on quantum hardware [43, 44].

For large-scale constrained optimization, quantum interior point methods (QIPMs) combine classical interior point frameworks with quantum linear algebra subroutines to accelerate the solution of the Newton systems that arise during each iteration [45, 46]. These methods have been developed for linear, semidefinite, and conic optimization problems and provide some of the strongest theoretical complexity guarantees currently known for quantum optimization in continuous domains. Unlike QAOA and other variational methods, QIPMs are primarily designed for fault-tolerant quantum computers and rely heavily on quantum linear system algorithms and efficient state preparation. Together, QAOA, Gibbs-sampling approaches, QHD, and QIPMs illustrate the diversity of current quantum optimization research, ranging from variational algorithms to asymptotically efficient quantum algorithms for large-scale optimization.

Motivation for Quantum Optimization. The preceding discussion suggests that the main computational bottleneck is not the lack of a differential equation solver, but the repeated search over candidate parameter values. Classical optimization methods can require many forward model evaluations, while direct quantum time integration is not yet practical for the nonlinear ODE systems considered here. This motivates a hybrid formulation in which data assimilation is used to construct a stabilized parameter-estimation cost functional, and quantum optimization is applied only after this cost functional has been converted into a finite-dimensional discrete search problem.

Retaining the data-assimilation component is important because full-state observations are rarely available in practical parameter estimation problems. Full-state observation represents the ideal setting for trajectory-based estimation, since every component of the model state can be compared directly with the model output. In many applications, however, only a subset of the state variables is observed. Data assimilation uses these partial observations, together with the model dynamics, to

stabilize the observed components and recover information about the unobserved components. Therefore, we retain the data-assimilation formulation in the proposed quantum optimization framework so that the resulting parameter estimator remains applicable to partially observed dynamical systems.

QUBO is a natural formulation for this discrete optimization stage. The cost functional is built from squared residuals and therefore has an intrinsic least-squares structure. Although the dependence of these residuals on the unknown parameters is generally nonlinear, the region of the parameter domain with small cost functional values can be approximated by a quadratic surrogate fitted from coarse-grid data-assimilation evaluations. Once the refined-grid parameter search is encoded using binary variables, this quadratic surrogate leads directly to a QUBO problem. Moreover, QUBO cost functionals admit a standard mapping to Ising Hamiltonians, making them compatible with QAOA and other quantum optimization methods. Thus, the QUBO formulation is not introduced as an arbitrary discretization; it is the mechanism that connects the classical cost functional to a quantum-compatible optimization problem. This also separates the role of the quantum algorithm from the dynamical simulation: the quantum routine is not used to solve the ODE, but only to search the binary-encoded surrogate landscape obtained from classical data-assimilation evaluations.

1.2. Contributions. The main contribution of this paper is a hybrid data-assimilation and quantum-optimization framework for parameter estimation in nonlinear ODE systems. The method uses coarse-grid cost functional values to construct a quadratic surrogate, encodes the refined parameter search as a QUBO, maps the resulting binary cost functional to an Ising Hamiltonian, and applies quantum optimizers to obtain parameter estimates.

The main contributions of this work are as follows:

- We develop a hybrid classical–quantum framework for parameter estimation in nonlinear ODE systems by combining nudging-based data assimilation with QUBO-based quantum optimization.
- We introduce a coarse-to-refined surrogate strategy in which expensive cost functional values are computed only on a coarse grid, while a QUBO-based quantum optimization stage searches over a refined binary-encoded parameter grid.
- We formulate the parameter estimation problem so that all ODE integrations remain classical, avoiding the need for quantum simulation of the underlying differential equations.
- We obtain parameter estimates directly from measured bitstrings, so the proposed approach does not require quantum state tomography [47].
- We provide theoretical statements and fitting-error diagnostics that clarify how surrogate approximation, QUBO fitting, and refined-grid resolution affect the recovered parameter estimate.
- We test the method on four model problems: the SIS and SIR epidemic models, whose trajectories approach stable equilibria under the regimes considered; the Lorenz–63 system in a chaotic regime; and a high-dimensional, multiscale two-layer Lorenz–96 system.

The remainder of the paper is organized as follows. In Section 2, we introduce the general parameter estimation formulation and describe the data-assimilation-augmented cost functional construction that serves as the starting point for the proposed method. In Section 3, we present the proposed quantum parameter estimation

framework, including coarse-grid cost functional evaluation, continuous quadratic surrogate fitting, refined-grid binary encoding, QUBO construction, Ising mapping, and quantum optimization. We also establish theoretical results describing how surrogate approximation error and refined-grid resolution affect the recovered parameter estimate. In Section 4, we present numerical experiments for the SIS, SIR, Lorenz-63, and two-layer Lorenz-96 systems using QAOA on an IBM simulator, QAOA on the IBM Kingston QPU, and SQA. Through these examples, we assess the method across epidemiological, chaotic, and high-dimensional multiscale dynamical systems, with particular attention to partial observations and refined-grid parameter recovery. Finally, in Section 5, we summarize the main findings and outline directions for future work.

2. Preliminaries. This section introduces the general dynamical-system and observation setting used throughout the paper. It then presents the data-assimilation-augmented parameter estimation formulation developed in [19], which serves as the starting point for the proposed quantum optimization framework.

Consider a nonlinear dynamical system governed by the ordinary differential equation

$$(2.1) \quad \dot{x}(t) = F(x(t), \theta), \quad x(0) = x_0,$$

where $x(t) \in \mathbb{R}^d$ denotes the state vector, $\theta \in \Theta \subset \mathbb{R}^{d_\theta}$ is the unknown parameter vector, and $F : \mathbb{R}^d \times \Theta \rightarrow \mathbb{R}^d$ is assumed to be sufficiently smooth. Let the observations be defined through the linear observation operator

$$(2.2) \quad y(t) = Hx(t),$$

where $y(t) \in \mathbb{R}^m$, m is the number of observed quantities, and $H \in \mathbb{R}^{m \times d}$ maps the full state to the observation space. In general, H may select individual state components or form linear combinations of the state variables. In the numerical examples considered in this work, H is a coordinate projection that selects the observed state components. Observations are available at times $\{t_k\}_{k=1}^{N_t}$ and are denoted by $y^{\text{data}}(t_k)$. The goal of parameter estimation is to determine parameter values for which the model output agrees with the observed data.

Data-Assimilation-Augmented Parameter Estimation. The parameter estimation approach developed in [19] uses a nudged dynamical system to stabilize the estimation process by incorporating observational data through a feedback term. For a candidate parameter vector $\tilde{\theta}$, the nudged system is defined by

$$(2.3) \quad \dot{\tilde{x}}(t) = F(\tilde{x}(t), \tilde{\theta}) + \mu H^T (y^{\text{data}}(t) - H(\tilde{x}(t))),$$

where $\mu > 0$ is the nudging coefficient. The initial condition for the nudged system need not coincide with the true initial condition. Instead, the nudging term drives the observed components of the nudged trajectory toward the available data.

Using the nudged trajectory, we define the data-misfit cost functional

$$(2.4) \quad C_{\text{nudge}}(\tilde{\theta}; \mu) = \sum_{k=1}^{N_t} \left\| y^{\text{data}}(t_k) - H(\tilde{x}(t_k; \tilde{\theta}, \mu)) \right\|^2,$$

where N_t is the number of observation times. The data-assimilation-augmented parameter estimation problem is then formulated as

$$(2.5) \quad \tilde{\theta}^* \in \arg \min_{\tilde{\theta} \in \Theta_s} C_{\text{nudge}}(\tilde{\theta}; \mu),$$

where $\Theta_s \subset \Theta$ is a bounded parameter search domain over which the minimization is performed. The resulting data-assimilation-augmented parameter estimation procedure is summarized in Algorithm 2.1.

Algorithm 2.1 Data–Assimilation–Augmented Parameter Estimation

Require: Observation times $\{t_k\}_{k=1}^{N_t}$, data $\{y^{\text{data}}(t_k)\}_{k=1}^{N_t}$, bounded parameter search domain $\Theta_s \subset \Theta$, nudging coefficient $\mu > 0$, initial guess $\tilde{\theta}^{(0)}$, stopping tolerance ε

- 1: Define the nudged ODE system (2.3) with parameter vector $\tilde{\theta}$ and nudging coefficient μ
- 2: Define the cost functional $C_{\text{nudge}}(\tilde{\theta}; \mu)$ in (2.4)
- 3: Initialize the optimizer at $\tilde{\theta}^{(0)}$
- 4: **while** optimizer has not converged **do**
- 5: Propose a candidate parameter vector $\tilde{\theta}$
- 6: Solve the nudged system (2.3) to obtain $\tilde{x}(t_k; \tilde{\theta}, \mu)$
- 7: Evaluate $C_{\text{nudge}}(\tilde{\theta}; \mu)$
- 8: Update the optimizer state
- 9: **end while**
- 10: **return** $\tilde{\theta}^*$

In practice, the minimization in (2.5) may be performed using a numerical optimization method such as the Nelder–Mead simplex method [48]. Each evaluation of the cost functional requires solving the nudged ODE system forward in time, so repeated numerical integration is the dominant computational cost. For further details on this data-assimilation-augmented parameter estimation method, we refer the reader to [19].

3. Proposed Quantum Optimization Framework. We propose a quantum-enabled extension of the data-assimilation-augmented optimization framework by reformulating the minimization of $C_{\text{nudge}}(\tilde{\theta}; \mu)$ as a combinatorial optimization problem. In contrast to a direct grid-based formulation, we distinguish between a coarse grid used for model-based cost functional evaluations and a refined grid used for the QUBO-based quantum search.

Let

$$(3.1) \quad \Theta_s = \prod_{j=1}^{d_\theta} [\theta_j^{\min}, \theta_j^{\max}]$$

denote the bounded parameter search domain. Let M_c and M_f denote the number of coarse and refined grid points per parameter, respectively, with $M_f \geq M_c$. The coarse grid for the j -th parameter is given by

$$(3.2) \quad \tilde{\theta}_{j,c}^{(i)} = \theta_j^{\min} + \frac{i}{M_c - 1} (\theta_j^{\max} - \theta_j^{\min}), \quad i = 0, \dots, M_c - 1.$$

For d_θ unknown parameters, this gives $L_c = M_c^{d_\theta}$ total coarse-grid points. Each coarse grid point defines a candidate parameter vector $\tilde{\theta}_{c,\ell}$, $\ell = 1, \dots, L_c$. For each candidate, we solve the nudged system (2.3) and compute $C_\ell = C_{\text{nudge}}(\tilde{\theta}_{c,\ell}; \mu)$. Thus, the expensive forward integrations and cost functional evaluations are performed only on the coarse grid.

The resulting coarse-grid cost functional values are then used to fit a continuous quadratic surrogate over the parameter domain. Let $\theta = (\theta_1, \dots, \theta_{d_\theta}) \in \Theta_s$ denote

a generic parameter vector, where θ_j is the j -th unknown parameter. For example, in the SIR model, $\theta = (\beta, \gamma)$, while in the Lorenz-63 model, $\theta = (\sigma, \rho, \beta)$. We approximate the cost functional by a quadratic function of the parameter components:

$$(3.3) \quad \widehat{C}(\theta) = c_0 + \sum_{j=1}^{d_\theta} c_j \theta_j + \sum_{1 \leq i \leq j \leq d_\theta} c_{ij} \theta_i \theta_j.$$

Here, c_0 is the constant coefficient, c_j are the linear coefficients, and c_{ij} are the quadratic interaction coefficients. The terms with $i = j$ represent squared parameter contributions, θ_i^2 , while the terms with $i < j$ represent pairwise interactions between different parameters. The coefficients $\{c_0, c_j, c_{ij}\}$ are determined by weighted least-squares fitting using the coarse-grid data $\left\{ \left(\tilde{\theta}_{c,\ell}, C_\ell \right) \right\}_{\ell=1}^{L_c}$. The use of least-squares fitting is appropriate in this setting for two related reasons. First, the cost functional is itself defined as a sum of squared residuals, so a least-squares surrogate is consistent with the structure of the underlying misfit functional. Second, the quadratic surrogate is linear in the unknown coefficients $\{c_0, c_j, c_{ij}\}$, which means that the surrogate coefficients can be computed by solving a linear least-squares problem rather than a nonlinear optimization problem.

To define the weights, let $C_{\min} = \min_{1 \leq \ell \leq L_c} C_\ell$, $C_{\max} = \max_{1 \leq \ell \leq L_c} C_\ell$. We first normalize the coarse-grid cost functional values by

$$\tilde{C}_\ell = \frac{C_\ell - C_{\min}}{C_{\max} - C_{\min} + \delta},$$

where $\delta > 0$ is a small numerical constant used to avoid division by zero. The least-squares weights are then chosen as $w_\ell = \exp(-\lambda \tilde{C}_\ell)$. In the numerical experiments, we use $\lambda = 8$. This weighting assigns the largest weights to the smallest cost functional values and progressively downweights regions with large cost functional values. Thus, the fitted quadratic surrogate is biased toward accuracy in the region with small cost functional values where the parameter minimizer is expected to lie.

Equivalently, let

$$q = 1 + d_\theta + \frac{d_\theta(d_\theta + 1)}{2}$$

denote the total number of constant, linear, squared, and pairwise interaction features in the quadratic surrogate. Define the quadratic feature vector by $\phi(\theta) = (1, \theta_1, \dots, \theta_{d_\theta}, \{\theta_i \theta_j\}_{1 \leq i \leq j \leq d_\theta})^T \in \mathbb{R}^q$. Thus, $\phi(\theta)$ contains the constant term, all linear terms, all squared terms, and all pairwise interaction terms. Let $\Phi_c \in \mathbb{R}^{L_c \times q}$ denote the coarse-grid design matrix, whose ℓ -th row is $\phi(\tilde{\theta}_{c,\ell})^T$. Let $C^{\text{sc}} \in \mathbb{R}^{L_c}$ denote the vector of normalized coarse-grid cost functional values, with entries $C_\ell^{\text{sc}} = \tilde{C}_\ell$. The coefficient vector $c \in \mathbb{R}^q$ is computed from

$$c = \arg \min_{\bar{c} \in \mathbb{R}^q} \left\| W_c^{1/2} (\Phi_c \bar{c} - C^{\text{sc}}) \right\|_2^2,$$

where $W_c = \text{diag}(w_1, \dots, w_{L_c})$. This gives the fitted continuous surrogate $\widehat{C}(\theta) = c^T \phi(\theta)$.

Next, we introduce the refined grid

$$(3.4) \quad \tilde{\theta}_{j,f}^{(i)} = \theta_j^{\min} + \frac{i}{M_f - 1} (\theta_j^{\max} - \theta_j^{\min}), \quad i = 0, \dots, M_f - 1.$$

The surrogate $\widehat{C}$ is evaluated on this refined grid, producing a refined discrete cost functional landscape without additional nudged ODE solves. Thus, the coarse grid controls the number of expensive model integrations, while the refined grid controls the resolution of the final binary search.

Each refined grid point is encoded by a binary decision vector $z \in \{0, 1\}^n$. If $b = \lceil \log_2 M_f \rceil$ bits are used per parameter, then $n = d_\theta b$. For the j -th parameter, let $z^{(j)} = (z_{j,0}, z_{j,1}, \dots, z_{j,b-1}) \in \{0, 1\}^b$ denote the block of bits assigned to that parameter. The corresponding refined-grid index is

$$(3.5) \quad i_j(z) = \sum_{r=0}^{b-1} 2^{b-1-r} z_{j,r}, \quad j = 1, \dots, d_\theta.$$

The decoded refined-grid parameter value is therefore

$$(3.6) \quad \tilde{\theta}_{j,f}(z) = \theta_j^{\min} + \frac{i_j(z)}{M_f - 1} (\theta_j^{\max} - \theta_j^{\min}), \quad j = 1, \dots, d_\theta.$$

Thus, $\tilde{\theta}_f(z) = (\tilde{\theta}_{1,f}(z), \dots, \tilde{\theta}_{d_\theta,f}(z))$ denotes the refined-grid parameter vector decoded from z . When M_f is a power of two, each b -bit block corresponds to a valid refined-grid index. In the numerical experiments below, $M_f = 32$, so each parameter is represented by five binary variables.

The refined surrogate values are then represented by a quadratic binary model of the form

$$(3.7) \quad E(z) = a_0 + \sum_{i=1}^n a_i z_i + \sum_{1 \leq i < j \leq n} a_{ij} z_i z_j, \quad z_i \in \{0, 1\}.$$

Here, $z = (z_1, \dots, z_n)$ is the binary vector encoding a refined-grid parameter vector, a_0 is the constant term, a_i are the linear QUBO coefficients, and a_{ij} are the pairwise quadratic interaction coefficients between binary variables. These coefficients are not model parameters; they are coefficients of the binary surrogate used to represent the refined surrogate landscape on the encoded parameter grid.

The QUBO coefficients $\{a_0, a_i, a_{ij}\}$ are obtained by weighted least-squares fitting so that $E(z) \approx \widehat{C}(\tilde{\theta}_f(z))$ over the refined grid. After the continuous surrogate is evaluated on the refined grid, the refined surrogate values are shifted and normalized to $[0, 1]$. If $\widehat{C}_f(z)$ denotes the normalized refined-grid surrogate value associated with bit-string z , then the QUBO least-squares weights are chosen as $w_q(z) = \exp(-\lambda \widehat{C}_f(z))$, again with $\lambda = 8$.

Because the continuous surrogate is quadratic in the parameters and the binary decoding in (3.6) is affine in the binary variables, the refined surrogate admits an exact QUBO representation. The weighted least-squares step is therefore used as a numerical procedure for recovering the coefficients of this binary quadratic representation. As a result, the QUBO fitting error is expected to be near machine precision, as observed in the numerical experiments.

The continuous surrogate coefficients in (3.3) are fitted using the normalized coarse-grid cost functional values, whereas the QUBO coefficients in (3.7) represent the normalized continuous surrogate on the refined binary grid. This produces the QUBO problem used in the final quantum optimization stage.

Using the standard transformation from binary variables to Ising spin variables,

$$z_i = \frac{1 - s_i}{2}, \quad z_i \in \{0, 1\}, \quad s_i \in \{-1, +1\},$$

the QUBO cost functional is mapped to an Ising Hamiltonian

$$(3.8) \quad \hat{H}_C = \kappa I + \sum_{i=1}^n h_i Z_i + \sum_{1 \leq i < j \leq n} J_{ij} Z_i Z_j,$$

where Z_i denotes the Pauli- Z operator acting on qubit i . Here, κ is a constant energy shift, h_i are the one-qubit Ising coefficients, and J_{ij} are the two-qubit coupling coefficients. These Ising coefficients are obtained algebraically from the QUBO coefficients $\{a_0, a_i, a_{ij}\}$. The constant shift κI does not affect the minimizer, but it is included in (3.8) for completeness.

The final step is to search for a low-energy configuration of the Ising Hamiltonian $\hat{H}_C$ using a quantum optimizer $\mathcal{S}$. In this work, the optimizer is treated as a solver for the binary energy landscape defined by the QUBO/Ising formulation. The low-energy bitstring obtained from this optimization step is decoded through (3.6) to obtain the corresponding refined-grid parameter estimate $\tilde{\theta}_Q$.

This formulation is independent of the particular optimization procedure used in the final QUBO/Ising search. In the numerical experiments, we use QAOA on an IBM simulator, QAOA on the IBM Kingston QPU, and SQA. After optimization, the solvers return candidate bitstrings. The returned low-energy bitstring is decoded to obtain a refined-grid parameter vector and hence the quantum-assisted estimate $\tilde{\theta}_Q$. The proposed hybrid procedure is summarized in Algorithm 3.1.

Least-Squares and QUBO Fitting Errors. The continuous quadratic surrogate introduces an approximation error because the cost functional is generally not exactly quadratic in the unknown parameters. We measure this error on the coarse training grid using the weighted root-mean-square error

$$\text{RMSE}_{\text{LS}} = \left(\frac{\sum_{\ell=1}^{L_c} w_{\ell} \left(\hat{C}(\tilde{\theta}_{c,\ell}) - \tilde{C}_{\ell} \right)^2}{\sum_{\ell=1}^{L_c} w_{\ell}} \right)^{1/2}.$$

We also monitor the maximum coarse-grid fitting error $e_{\infty, \text{LS}} = \max_{1 \leq \ell \leq L_c} \left| \hat{C}(\tilde{\theta}_{c,\ell}) - \tilde{C}_{\ell} \right|$. These quantities measure how accurately the continuous quadratic surrogate approximates the normalized cost functional values used for fitting.

After evaluating the continuous surrogate on the refined binary grid, the QUBO coefficients are recovered through weighted least squares. Since the continuous surrogate is quadratic in the parameters and the binary parameter decoding is affine, an exact binary quadratic representation exists. Let $\mathcal{Z} \subseteq \{0, 1\}^n$ denote the set of bitstrings corresponding to the refined-grid parameter points. The numerical accuracy of the recovered QUBO representation is measured by

$$\text{MSE}_{\text{QUBO}} = \frac{1}{|\mathcal{Z}|} \sum_{z \in \mathcal{Z}} \left(E(z) - \hat{C}_f(z) \right)^2,$$

where $|\mathcal{Z}|$ is the number of refined-grid bitstrings, $\hat{C}_f(z)$ is the normalized value of $\hat{C}(\tilde{\theta}_f(z))$, and $E(z)$ is the recovered QUBO energy. Thus, MSE_{QUBO} measures numerical coefficient-recovery error rather than a separate model-approximation error.

Conditioning of the Least-Squares Fits. A potential numerical challenge in the surrogate construction is the conditioning of the least-squares design matrices. Both the continuous quadratic surrogate and the QUBO surrogate are fitted using

Algorithm 3.1 Proposed Quantum Optimization Framework for Data-Assimilation-Augmented Parameter Estimation

Require: Observation times $\{t_k\}_{k=1}^{N_t}$, data $\{y^{\text{data}}(t_k)\}_{k=1}^{N_t}$, bounded parameter search domain $\Theta_s \subset \Theta$, nudging coefficient $\mu > 0$, coarse grid size M_c , refined grid size M_f , QUBO/Ising optimizer $\mathcal{S}$

- 1: Discretize Θ_s using an M_c -point coarse grid per parameter
- 2: **for** each coarse grid point $\theta_{c,\ell}$ **do**
- 3: Solve the nudged system (2.3) with parameter vector $\tilde{\theta}_{c,\ell}$ and gain μ
- 4: Compute $C_\ell = C_{\text{nudge}}(\tilde{\theta}_{c,\ell}; \mu)$
- 5: **end for**
- 6: Normalize the coarse-grid cost functional values to obtain $\{\tilde{C}_\ell\}$
- 7: Define weights $w_\ell = \exp(-\lambda \tilde{C}_\ell)$
- 8: Fit the continuous quadratic surrogate $\hat{C}$ in (3.3) by weighted least squares
- 9: Monitor the continuous surrogate fitting errors RMSE_{LS} and $e_{\infty, \text{LS}}$
- 10: Discretize $\hat{C}$ using an M_f -point refined grid per parameter
- 11: Evaluate $\hat{C}$ on the refined grid
- 12: Encode refined grid points by bitstrings $z \in \{0, 1\}^n$, where $n = d_\theta \lceil \log_2 M_f \rceil$
- 13: Normalize the refined surrogate values on the refined grid
- 14: Define QUBO weights $w_q(z) = \exp(-\lambda \hat{C}_f(z))$
- 15: Recover the coefficients of the QUBO representation $E(z)$ in (3.7) by weighted least squares
- 16: Monitor the numerical QUBO representation error MSE_{QUBO}
- 17: Map $E(z)$ to an Ising Hamiltonian $\hat{H}_C$
- 18: Use the selected QUBO/Ising optimizer $\mathcal{S}$ to approximately minimize the energy of $\hat{H}_C$ and obtain a bitstring z^*
- 19: Decode z^* to obtain the corresponding parameter estimate $\tilde{\theta}_Q$
- 20: **return** $\tilde{\theta}_Q$

feature matrices whose columns may be correlated. For example, the continuous surrogate includes linear, squared, and pairwise interaction terms in the parameters, while the QUBO surrogate includes binary variables and their pairwise products. On a structured grid, and especially after weighting the low-cost region more heavily, these features can exhibit multicollinearity, so the corresponding design matrix can become ill-conditioned or, in degenerate cases, rank deficient [49, 50, 51]. Such multicollinearity can make individual fitted coefficients sensitive to perturbations and difficult to interpret. However, in predictive or optimization-oriented uses of least-squares models, multicollinearity does not necessarily imply poor performance of the fitted response surface, since its main effect is often on the stability and interpretation of individual fitted coefficients rather than on the overall fitted response [52, 53]. In the present framework, the fitted coefficients are not interpreted as physical or model parameters; they are used only to represent the surrogate cost functional landscape and identify low-cost regions. In the present experiments, the fitted surrogates appeared numerically stable, as indicated by the monitored values of RMSE_{LS} , $e_{\infty, \text{LS}}$, and MSE_{QUBO} . Thus, although multicollinearity is a possible challenge of the least-squares fitting step, it did not prevent stable recovery of the low-cost parameter region in the examples considered in this work.

The construction above separates expensive model evaluation from the resolution

of the binary optimization problem. The nudged system is solved only on the coarse grid, while the QUBO/Ising optimization stage searches over the refined grid through the learned surrogate. Since the quantum optimizer minimizes the QUBO representation rather than the original cost functional in (2.4), it is important to understand how closely the minimizer of the surrogate problem approximates the minimizer of the target discrete cost functional.

Here, C denotes the target cost functional on the refined binary grid, while E denotes the QUBO representation minimized by the quantum optimizer. In the present quadratic construction, the dominant approximation arises from fitting the continuous quadratic surrogate to the coarse-grid cost functional values. The subsequent QUBO coefficient-recovery step reproduces the refined surrogate up to numerical roundoff, as measured by MSE_{QUBO} . Therefore, the uniform error parameter ε in Theorem 3.1 primarily reflects the continuous surrogate approximation error, together with any numerical error introduced when recovering the QUBO coefficients.

THEOREM 3.1. *Let $\mathcal{Z}$ be a nonempty finite set. Let $C : \mathcal{Z} \rightarrow \mathbb{R}$ denote the target refined-grid cost functional and let $E : \mathcal{Z} \rightarrow \mathbb{R}$ denote a surrogate cost functional. Assume that*

$$(3.9) \quad \sup_{z \in \mathcal{Z}} |E(z) - C(z)| \leq \varepsilon$$

for some $\varepsilon \geq 0$. Let $z_C \in \arg \min_{z \in \mathcal{Z}} C(z)$, $z_E \in \arg \min_{z \in \mathcal{Z}} E(z)$. Then

$$(3.10) \quad C(z_E) \leq \min_{z \in \mathcal{Z}} C(z) + 2\varepsilon.$$

Moreover, assume z_C is the unique minimizer of C on $\mathcal{Z}$, and define the optimality gap

$$(3.11) \quad \Delta := \min_{z \in \mathcal{Z} \setminus \{z_C\}} (C(z) - C(z_C)).$$

If $\Delta > 0$ and $\varepsilon < \Delta/2$, then z_C is the unique minimizer of E on $\mathcal{Z}$, and hence

$$(3.12) \quad z_E = z_C.$$

Proof. From Eq. (3.9), for all $z \in \mathcal{Z}$,

$$(3.13) \quad C(z) - \varepsilon \leq E(z) \leq C(z) + \varepsilon.$$

Since z_E minimizes E on $\mathcal{Z}$, $E(z_E) \leq E(z_C)$. Using Eq. (3.13) at $z = z_E$ and $z = z_C$, we obtain

$$C(z_E) - \varepsilon \leq E(z_E) \leq E(z_C) \leq C(z_C) + \varepsilon.$$

Thus, $C(z_E) \leq C(z_C) + 2\varepsilon$. Since $z_C \in \arg \min_{\mathcal{Z}} C$, we have $C(z_C) = \min_{z \in \mathcal{Z}} C(z)$, which proves Eq. (3.10).

Now assume that z_C is the unique minimizer of C and that Δ is defined by Eq. (3.11). Then, for every $z \in \mathcal{Z} \setminus \{z_C\}$,

$$(3.14) \quad C(z) \geq C(z_C) + \Delta.$$

Using Eq. (3.13) and Eq. (3.14), for $z \neq z_C$,

$$E(z) \geq C(z) - \varepsilon \geq C(z_C) + \Delta - \varepsilon, \quad E(z_C) \leq C(z_C) + \varepsilon.$$

Therefore, $E(z) - E(z_C) \geq \Delta - 2\varepsilon$. If $\varepsilon < \Delta/2$, then $\Delta - 2\varepsilon > 0$, and hence $E(z) > E(z_C)$ for all $z \neq z_C$. Thus, z_C is the unique minimizer of E on $\mathcal{Z}$, and Eq. (3.12) follows. $\square$

The discretization of the parameter domain introduces an approximation error because the true minimizer of the continuous cost functional may not lie exactly on the refined grid. Theorem 3.2 quantifies how this error depends on the refined grid spacing. Its assumptions are local regularity conditions near an identifiable minimizer and are intended to describe the effect of grid resolution, rather than to provide a global characterization of arbitrary nonconvex cost functional landscapes.

THEOREM 3.2. *Let*

$$\Theta_s = \prod_{j=1}^{d_\theta} [\theta_j^{\min}, \theta_j^{\max}] \subset \Theta \subset \mathbb{R}^{d_\theta},$$

and let $C : \Theta_s \rightarrow \mathbb{R}$ be continuously differentiable with a unique minimizer $\theta^* \in \text{int}(\Theta_s)$. Assume there exist constants $\alpha, L > 0$ such that, for all $\theta \in \Theta_s$,

$$C(\theta) \geq C(\theta^*) + \frac{\alpha}{2} \|\theta - \theta^*\|^2, \quad C(\theta) \leq C(\theta^*) + \frac{L}{2} \|\theta - \theta^*\|^2.$$

Let $\Theta_{h_f} \subset \Theta_s$ be the uniform refined grid induced by (3.4), and set

$$h_f := \max_{1 \leq j \leq d_\theta} \frac{\theta_j^{\max} - \theta_j^{\min}}{M_f - 1}.$$

Let $\theta_{h_f} \in \arg \min_{\theta \in \Theta_{h_f}} C(\theta)$, and assume $\text{dist}_\infty(\theta^*, \partial\Theta_s) \geq h_f/2$. Then

$$0 \leq C(\theta_{h_f}) - C(\theta^*) \leq \frac{L}{8} d_\theta h_f^2, \quad \|\theta_{h_f} - \theta^*\| \leq \sqrt{\frac{L}{\alpha}} \frac{\sqrt{d_\theta}}{2} h_f.$$

Proof. The assumption $\text{dist}_\infty(\theta^*, \partial\Theta_s) \geq \frac{h_f}{2}$ ensures that θ^* lies at least half a refined-grid spacing away from the boundary of Θ_s . Therefore, there exists a grid point $\tilde{\theta} \in \Theta_{h_f}$ such that

$$|\tilde{\theta}_j - \theta_j^*| \leq \frac{h_f}{2}, \quad j = 1, \dots, d_\theta.$$

Hence,

$$\|\tilde{\theta} - \theta^*\|^2 \leq d_\theta (h_f/2)^2 = \frac{d_\theta h_f^2}{4}.$$

Using the upper quadratic bound gives

$$C(\tilde{\theta}) - C(\theta^*) \leq \frac{L}{2} \frac{d_\theta h_f^2}{4} = \frac{L}{8} d_\theta h_f^2.$$

By optimality of θ_{h_f} on Θ_{h_f} , we have $C(\theta_{h_f}) \leq C(\tilde{\theta})$. Since $\Theta_{h_f} \subset \Theta_s$ and θ^* is the minimizer of C on Θ_s , we also have $C(\theta_{h_f}) \geq C(\theta^*)$. Therefore,

$$0 \leq C(\theta_{h_f}) - C(\theta^*) \leq \frac{L}{8} d_\theta h_f^2.$$

The parameter error bound follows by combining this estimate with the lower quadratic growth condition:

$$\frac{\alpha}{2} \|\theta_{h_f} - \theta^*\|^2 \leq C(\theta_{h_f}) - C(\theta^*) \leq \frac{L}{8} d_\theta h_f^2.$$

Taking square roots gives

$$\|\theta_{h_f} - \theta^*\| \leq \sqrt{\frac{L}{\alpha}} \frac{\sqrt{d_\theta}}{2} h_f. \quad \square$$

4. Numerical Results. This section evaluates Algorithm 3.1 on four model problems: the SIS model, the SIR model, the Lorenz-63 system, and the two-layer Lorenz-96 system. The SIS and SIR examples represent epidemiological models whose trajectories approach stable equilibria under the parameter regimes considered. The Lorenz-63 example provides a chaotic test problem, while the two-layer Lorenz-96 example provides a high-dimensional, multiscale dissipative system. Together, these examples evaluate the proposed framework across steady-state, chaotic, and high-dimensional dynamical systems.

Common Numerical and Optimization Settings. In all experiments, synthetic observations are generated by numerically solving the corresponding dynamical system with prescribed true parameter values at a finite collection of observation times. The nudged systems are initialized from prescribed initial states that differ from the reference initial conditions. During integration of the nudged systems, the discrete observations are linearly interpolated to provide observation values at the internal time points selected by the ODE solver. The model-specific reference initial conditions, observation intervals, observed components, nudging coefficients, and parameter search domains are stated in the corresponding subsections.

For each unknown parameter, the cost functional is evaluated at $M_c = 8$ coarse-grid points. The resulting coarse-grid cost functional values are shifted and normalized to $[0, 1]$ and used to fit a continuous quadratic surrogate by weighted least squares with weighting parameter $\lambda = 8$. The continuous surrogate is then evaluated on a refined grid containing $M_f = 32$ points per unknown parameter without requiring additional data-assimilation solves. Since $32 = 2^5$, each unknown parameter is represented using five binary variables. The coefficients of the corresponding refined-grid QUBO representation are recovered by weighted least squares using the same weighting parameter $\lambda = 8$. The numerical QUBO representation error MSE_{QUBO} is monitored to verify that the binary quadratic model reproduces the refined continuous surrogate up to numerical precision.

The resulting QUBO problems are solved using QAOA on an IBM quantum simulator, QAOA on the IBM Kingston QPU, and SQA implemented using D-Wave’s `PathIntegralAnnealingSampler`. The same QAOA implementation and execution settings are used across all four model problems. For the SQA experiments, 5000 reads and 2000 schedule points are used with random seed 123, and the lowest-energy sampled bitstring is selected as the final solution. For each optimization procedure, the returned low-energy bitstring is decoded on the refined parameter grid to obtain the corresponding parameter estimate.

For the two-parameter examples (SIS, SIR, and Lorenz-96), the coarse grid contains $8^2 = 64$ data-assimilation solves and the refined grid contains $32^2 = 1024$ binary-encoded parameter points. Since each parameter is represented by five binary variables, the refined search uses $2 \times 5 = 10$ binary variables. For the Lorenz-63 model, the coarse grid contains $8^3 = 512$ data-assimilation solves and the refined grid contains $32^3 = 32768$ binary-encoded parameter points, corresponding to $3 \times 5 = 15$ binary variables. The common grid sizes, observations, and binary encodings are summarized in Table 1.

Model	Unknowns	Observed data	DA solves	Binary variables
SIS	β, γ	$I(t)$	$8^2 = 64$	10
SIR	β, γ	$I(t)$	$8^2 = 64$	10
Lorenz-63	σ, ρ, β	$x(t)$	$8^3 = 512$	15
Lorenz-96	$d_{u,5}, d_{u,15}$	$u_5(t), u_{15}(t)$	$8^2 = 64$	10

Table 1: Summary of the numerical experiments. In all cases, cost functional values are computed on a coarse parameter grid, while the refined binary parameter search is represented as a QUBO and solved using the selected optimization procedures.

4.1. SIS Model. We first consider the susceptible–infected–susceptible (SIS) model [1]. The population is divided into susceptible and infected classes, represented by the population fractions $S(t)$ and $I(t)$, respectively. Thus, $S(t) + I(t) = 1$. The dynamics are

$$(4.1) \quad \begin{cases} \dot{S} = -\beta SI + \gamma I, \\ \dot{I} = \beta SI - \gamma I, \end{cases}$$

where $\beta > 0$ is the transmission rate and $\gamma > 0$ is the recovery rate.

Synthetic observations are generated using $S(0) = 0.99875, I(0) = 0.00125$, with true parameters $\beta_{\text{true}} = 0.65, \gamma_{\text{true}} = 0.25$. The system is solved on $[0, 40]$ using 200 observation times, and only the infected prevalence $I(t)$ is treated as observed data.

For each candidate parameter pair, we solve the nudged SIS system

$$(4.2) \quad \begin{cases} \dot{\tilde{S}} = -\beta \tilde{S} \tilde{I} + \gamma \tilde{I}, \\ \dot{\tilde{I}} = \beta \tilde{S} \tilde{I} - \gamma \tilde{I} + \mu_I (I_{\text{data}}(t) - \tilde{I}), \end{cases}$$

with $\mu_I = 0.5$. The cost functional is

$$(4.3) \quad C(\beta, \gamma) = \sum_{i=1}^{N_t} \left| I_{\text{data}}(t_i) - \tilde{I}(t_i; \beta, \gamma) \right|^2.$$

The parameter search domain is $\Theta_s = [0.55, 0.75] \times [0.18, 0.32]$, corresponding to the parameters (β, γ) .

Table 2 reports the SIS parameter estimates obtained from the IBM simulator, IBM Kingston QPU, and SQA. All three implementations recover parameter values close to the true generating parameters. The estimated value of β has relative error approximately 1.48%, while the estimated value of γ has relative error approximately 2.70%.

Parameter	True value	IBM Simulator	IBM QPU	SQA
β	0.65	0.659677 (1.48%)	0.659677 (1.48%)	0.653226 (0.49%)
γ	0.25	0.256774 (2.70%)	0.256774 (2.70%)	0.252258 (0.90%)

Table 2: Estimated SIS parameters obtained from the IBM simulator, IBM Kingston QPU, and SQA. Relative percentage errors are shown in parentheses.

Figure 1 compares the susceptible and infected population-fraction trajectories generated using the estimated parameters with the true SIS trajectories. The estimated trajectories closely overlap with the true solutions for both compartments over

the full time interval. This shows that the recovered parameters reproduce the SIS dynamics accurately, even though the final parameter search is performed through the QUBO/Ising optimization formulation.

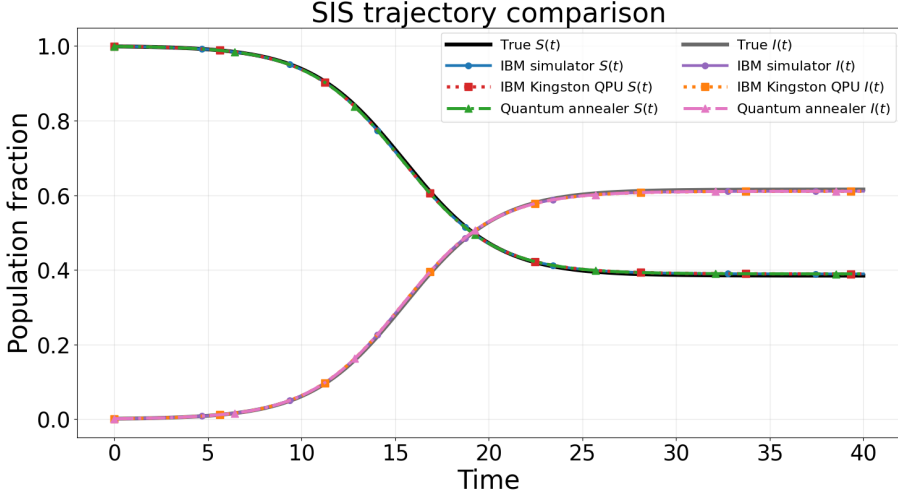


Fig. 1: SIS trajectory comparison for the susceptible and infected population fractions using the true parameters and the parameters estimated by the IBM simulator, IBM Kingston QPU, and SQA.

4.2. SIR Model. We consider the classical susceptible–infected–recovered (SIR) model, which is commonly used to describe the spread of an infectious disease in a closed population [1]. The population is divided into three compartments: susceptible $S(t)$, infected $I(t)$, and recovered $R(t)$. The dynamics are governed by

$$(4.4) \quad \begin{cases} \dot{S} = -\beta SI, \\ \dot{I} = \beta SI - \gamma I, \\ \dot{R} = \gamma I, \end{cases}$$

where $\beta > 0$ is the transmission rate and $\gamma > 0$ is the recovery rate. Throughout this subsection, the state variables are normalized population fractions, so that $S(t), I(t), R(t) \in [0, 1]$, and the total population satisfies $S(t) + I(t) + R(t) = 1$.

Synthetic infected-prevalence observations are generated by numerically solving Eq. (4.4) with $\beta_{\text{true}} = 0.32$, $\gamma_{\text{true}} = 0.12$, and initial condition $S(0) = 0.99, I(0) = 0.01, R(0) = 0$. The system is solved over the time interval $[0, 30]$. The infected component of the trajectory, denoted by $I_{\text{data}}(t)$, is treated as the observed prevalence data. Thus, the estimation procedure uses only observations of the infected population fraction, while β and γ are treated as unknown.

The parameter search domain is $\Theta_s = [0.10, 0.50] \times [0.05, 0.25]$, corresponding to the parameters (β, γ) . For each coarse-grid parameter pair, the nudged SIR system is solved with nudging gain $\mu = 0.5$. Since the infected compartment is observed, the

nudging term is applied to the infected equation:

$$(4.5) \quad \begin{cases} \dot{\tilde{S}} = -\beta\tilde{S}\tilde{I}, \\ \dot{\tilde{I}} = \beta\tilde{S}\tilde{I} - \gamma\tilde{I} + \mu(I_{\text{data}}(t) - \tilde{I}), \\ \dot{\tilde{R}} = \gamma\tilde{I}. \end{cases}$$

The data-misfit cost functional is evaluated using only the observed infected-prevalence component:

$$(4.6) \quad C(\beta, \gamma) = \sum_{i=1}^{N_t} \left| I_{\text{data}}(t_i) - \tilde{I}(t_i; \beta, \gamma) \right|^2,$$

where $N_t = 200$ is the number of observation times.

Table 3 reports the SIR parameter estimates obtained from the IBM simulator, IBM Kingston QPU, and SQA. All three implementations recover the same parameter values, which are close to the true generating parameters. The estimated transmission rate β has relative error approximately 0.20%, while the estimated recovery rate γ has relative error approximately 0.81%.

Parameter	True value	IBM Simulator	IBM QPU	SQA
β	0.32	0.319355 (0.20%)	0.319355 (0.20%)	0.319355 (0.20%)
γ	0.12	0.120968 (0.81%)	0.120968 (0.81%)	0.120968 (0.81%)

Table 3: Estimated SIR parameters obtained from the IBM simulator, IBM Kingston QPU, and SQA. Relative percentage errors are shown in parentheses.

Figure 2 compares the susceptible, infected, and recovered population-fraction trajectories generated using the estimated parameters with the true SIR trajectories. The estimated trajectories closely overlap with the true solutions for all three compartments over the full time interval. This shows that the recovered parameters reproduce the SIR dynamics accurately from infected-prevalence observations, even though the final parameter search is performed through the QUBO/Ising optimization formulation.

4.3. Lorenz-63 System. The Lorenz-63 system was introduced by Edward Lorenz in 1963 as a simplified mathematical model for atmospheric convection [3]. It is a three-dimensional nonlinear system of ordinary differential equations given by

$$(4.7) \quad \begin{cases} \dot{x} = \sigma(y - x), \\ \dot{y} = x(\rho - z) - y, \\ \dot{z} = xy - \beta z, \end{cases}$$

where $x(t)$, $y(t)$, and $z(t)$ denote the state variables, and σ , ρ , and β are model parameters. In the physical interpretation of the model, x is proportional to the intensity of convective motion, y represents the horizontal temperature variation, and z represents the vertical temperature variation [5]. The parameter σ is the Prandtl number, ρ is related to the Rayleigh number, and β depends on the geometry of the convective layer.

For the Lorenz-63 experiment, we choose the true parameter values $\sigma_{\text{true}} = 10$, $\rho_{\text{true}} = 30$, $\beta_{\text{true}} = \frac{8}{3} \approx 2.666667$, for which the system exhibits chaotic dynamics.

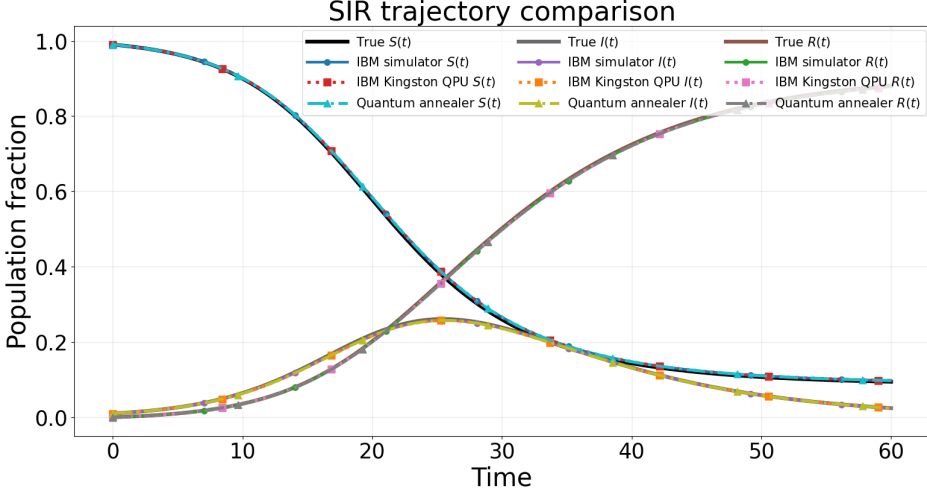


Fig. 2: SIR trajectory comparison for the susceptible, infected, and recovered population fractions using the true parameters and the parameters estimated by the IBM simulator, IBM Kingston QPU, and SQA.

We use the initial condition $x(0) = 1, y(0) = 1, z(0) = 1$. Synthetic observations are generated by numerically solving Eq. (4.7) over $[0, 20]$ using 400 uniformly spaced time points. Only the x -component of the resulting trajectory, denoted by $x_{\text{data}}(t)$, is treated as observed data. Therefore, although the full Lorenz–63 system is used to generate the synthetic trajectory, the estimation procedure has access only to partial observations. The parameters σ , ρ , and β are all treated as unknown.

The parameter search domain is $\Theta_s = [8, 12] \times [28, 32] \times [2.4, 2.9]$, corresponding to the parameters (σ, ρ, β) . For each coarse-grid parameter triple, the nudged Lorenz–63 system is solved with nudging gain $\mu = 30$. Since only the x -component is observed, the nudging term is applied only to the x -equation:

$$(4.8) \quad \begin{cases} \dot{\tilde{x}} = \sigma(\tilde{y} - \tilde{x}) + \mu(x_{\text{data}}(t) - \tilde{x}), \\ \dot{\tilde{y}} = \tilde{x}(\rho - \tilde{z}) - \tilde{y}, \\ \dot{\tilde{z}} = \tilde{x}\tilde{y} - \beta\tilde{z}. \end{cases}$$

The data-misfit cost functional is evaluated using only the observed x -component:

$$(4.9) \quad C(\sigma, \rho, \beta) = \sum_{i=1}^{N_t} |x_{\text{data}}(t_i) - \tilde{x}(t_i; \sigma, \rho, \beta)|^2,$$

where $N_t = 400$. The continuous quadratic surrogate fitted from the 8^3 coarse-grid cost functional values is evaluated on a refined 32^3 grid and encoded as a QUBO cost functional.

Table 4 reports the Lorenz–63 parameter estimates obtained from the IBM simulator, IBM Kingston QPU, and SQA. The estimates are close to the true parameter values, with relative errors ranging from about 2.80% to 4.56%. These discrepancies are expected in this chaotic and partially observed setting, where only the x -component is used in the cost functional.

Parameter	True value	IBM simulator	IBM QPU	SQA
σ	10.0	9.67741 (3.23%)	9.67741 (3.23%)	9.67741 (3.23%)
ρ	30.0	31.22580 (4.09%)	30.70967 (2.80%)	31.09677 (2.36%)
β	2.67	2.54516 (4.56%)	2.57741 (3.35%)	2.593548 (2.74%)

Table 4: Estimated Lorenz–63 parameters obtained from the IBM simulator, IBM Kingston QPU, and SQA using only x -component observations. Relative percentage errors are shown in parentheses.

Figure 3 compares the phase-space trajectories generated using the estimated parameters with the true Lorenz–63 trajectory. The estimated trajectories reproduce the overall structure of the Lorenz attractor and remain close to the true attractor despite small differences in the recovered parameter values. Because the Lorenz–63 system is chaotic, pointwise agreement over long time intervals is not expected; instead, the phase-space comparison demonstrates that the estimated parameters preserve the characteristic geometry of the underlying chaotic dynamics.

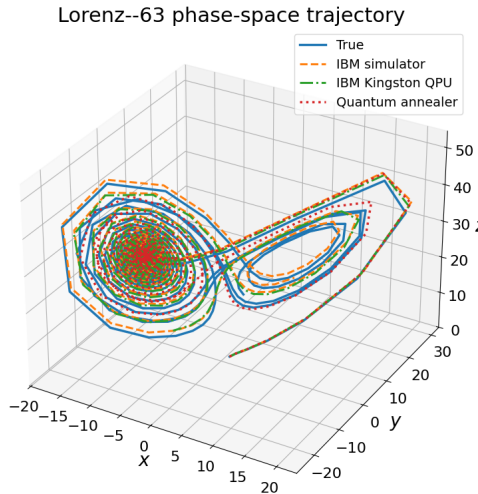


Fig. 3: Lorenz–63 phase-space trajectory comparison using the true parameters and the parameters estimated by the IBM simulator, IBM Kingston QPU, and SQA.

4.4. Lorenz–96 System. We finally consider a high-dimensional, multiscale two-layer Lorenz–96 system. The original Lorenz–96 model is a standard simplified model for atmospheric dynamics and predictability [4]. In the one-layer setting, the dynamics are

$$(4.10) \quad \frac{du_k}{dt} = u_{k-1}(u_{k+1} - u_{k-2}) - d_k u_k + F,$$

where the indices are interpreted periodically, $u_k(t)$ denotes the large-scale state variable, d_k is a damping coefficient, and F is a constant forcing term.

Following the two-layer Lorenz–96 formulation in [20], we use a model with K slow variables $u_k(t)$, $k = 0, \dots, K - 1$, and J fast variables $v_{k,j}(t)$, $j = 0, \dots, J - 1$,

associated with each slow variable. The two-layer system is

$$(4.11) \quad \begin{cases} \dot{u}_k = u_{k-1}(u_{k+1} - u_{k-2}) + \sum_{j=0}^{J-1} \gamma_j v_{k,j} u_k - d_{u,k} u_k + F, \\ \dot{v}_{k,j} = -d_{v,k,j} v_{k,j} - \gamma_j u_k^2, \end{cases}$$

where the indices in the slow variables are interpreted periodically. Here γ_j controls the coupling between the slow and fast variables, while $d_{u,k}$ and $d_{v,k,j}$ are damping parameters for the slow and fast variables, respectively.

In this experiment, we use $K = 40, J = 5, F = 5$, and the coupling vector $\gamma = (0.10, 0.15, 0.20, 0.25, 0.35)$. The true slow damping parameters are generated by

$$(4.12) \quad d_{u,k}^{\text{true}} = 1 + 0.7 \cos\left(\frac{2\pi(k+1)}{5}\right), \quad k = 0, \dots, K-1,$$

and the true fast damping parameters are

$$d_{v,k,j}^{\text{true}} = (0.2, 0.5, 1.0, 2.0, 5.0), \quad k = 0, \dots, K-1.$$

The goal is to estimate the two slow damping parameters $d_{u,5}$ and $d_{u,15}$. The corresponding true values are

$$d_{u,5}^{\text{true}} = 1.216312, \quad d_{u,15}^{\text{true}} = 1.216312.$$

All other parameters are fixed at their true values. The corresponding slow components $u_5(t)$ and $u_{15}(t)$ are observed.

Synthetic observations are generated by solving Eq. (4.11) on $[0, 40]$ using 600 uniformly spaced time points. The deterministic initial condition is

$$u_k(0) = F + 0.2 \sin\left(\frac{2\pi k}{K}\right), \quad v_{k,j}(0) = 0.1.$$

For each candidate parameter pair $\theta = (d_{u,5}, d_{u,15})$, we solve a nudged two-layer Lorenz-96 system. The nudging terms are applied only to the observed slow equations u_5 and u_{15} . Thus,

$$(4.13) \quad \begin{cases} \dot{\tilde{u}}_k = \tilde{u}_{k-1}(\tilde{u}_{k+1} - \tilde{u}_{k-2}) + \sum_{j=0}^{J-1} \gamma_j \tilde{v}_{k,j} \tilde{u}_k - d_{u,k} \tilde{u}_k + F, & k \notin \{5, 15\}, \\ \dot{\tilde{u}}_5 = \tilde{u}_4(\tilde{u}_6 - \tilde{u}_3) + \sum_{j=0}^{J-1} \gamma_j \tilde{v}_{5,j} \tilde{u}_5 - d_{u,5} \tilde{u}_5 + F + \mu(u_{5,\text{data}}(t) - \tilde{u}_5), \\ \dot{\tilde{u}}_{15} = \tilde{u}_{14}(\tilde{u}_{16} - \tilde{u}_{13}) + \sum_{j=0}^{J-1} \gamma_j \tilde{v}_{15,j} \tilde{u}_{15} - d_{u,15} \tilde{u}_{15} + F + \mu(u_{15,\text{data}}(t) - \tilde{u}_{15}), \\ \dot{\tilde{v}}_{k,j} = -d_{v,k,j} \tilde{v}_{k,j} - \gamma_j \tilde{u}_k^2. \end{cases}$$

In the numerical experiment, the nudging gain is $\mu = 15$.

The cost functional is computed using the two observed slow components:

$$(4.14) \quad C(d_{u,5}, d_{u,15}) = \sum_{i=1}^{N_t} |u_{5,\text{data}}(t_i) - \tilde{u}_5(t_i)|^2 + \sum_{i=1}^{N_t} |u_{15,\text{data}}(t_i) - \tilde{u}_{15}(t_i)|^2.$$

Here $N_t = 600$. The parameter search interval for each unknown parameter is chosen as $\theta_j \in [\theta_j^{\text{true}} - 0.5, \theta_j^{\text{true}} + 0.5]$, with a positive lower bound imposed when necessary. The coarse data-assimilation grid contains $8^2 = 64$ expensive model evaluations, and the refined QUBO search uses a 32×32 grid. Since each parameter is encoded using five bits, the final binary optimization problem uses 10 binary variables.

Table 5 reports the recovered parameter values obtained from the IBM simulator, IBM Kingston QPU, and SQA. All three implementations recover the same estimates for $d_{u,5}$ and $d_{u,15}$, with relative errors approximately 1.33%.

Parameter	True value	IBM simulator	IBM QPU	SQA
$d_{u,5}$	1.216312	1.200183 (1.33%)	1.200183 (1.33%)	1.200183 (1.33%)
$d_{u,15}$	1.216312	1.200183 (1.33%)	1.200183 (1.33%)	1.200183 (1.33%)

Table 5: Estimated two-layer Lorenz-96 parameters obtained from the IBM simulator, IBM Kingston QPU, and SQA. Relative percentage errors are shown in parentheses.

4.5. Surrogate and QUBO Representation Errors. Table 6 summarizes the continuous-surrogate fitting errors and numerical QUBO representation errors for the four model problems. The quantities RMSE_{LS} and $e_{\infty, \text{LS}}$ measure the accuracy of the continuous quadratic surrogate in approximating the normalized coarse-grid cost functional values, while MSE_{QUBO} measures the numerical accuracy of the QUBO representation on the refined grid.

Model	RMSE_{LS}	$e_{\infty, \text{LS}}$	MSE_{QUBO}
SIS	4.258770×10^{-3}	9.251285×10^{-2}	1.711676×10^{-30}
SIR	2.034177×10^{-2}	1.216511×10^{-1}	1.379741×10^{-31}
Lorenz-63	9.213976×10^{-2}	8.081225×10^{-1}	1.367646×10^{-28}
Lorenz-96	2.371029×10^{-3}	2.344146×10^{-2}	7.624507×10^{-31}

Table 6: Continuous-surrogate fitting errors and numerical QUBO representation errors for the four numerical experiments.

The QUBO representation errors are near numerical precision for the reported models, indicating that the QUBO coefficient-recovery step introduces no meaningful additional approximation.

5. Conclusions. This work developed a hybrid classical-quantum framework for parameter estimation in nonlinear dynamical systems. The proposed approach combines data-assimilation-augmented cost functional construction with a QUBO optimization stage. All dynamical simulations are performed classically: the nudged ODE system is solved only on a prescribed coarse parameter grid, and the resulting data-misfit cost functional values are used to construct a quadratic surrogate. This surrogate is then evaluated on a refined grid, encoded as a QUBO, mapped to an Ising Hamiltonian, and approximately minimized using quantum optimizers. In this way, the quantum component is used only for the discrete optimization stage, while the potentially difficult task of simulating nonlinear ODEs remains entirely classical. This separation makes the framework compatible with quantum optimization methods and avoids the need for quantum state tomography or direct quantum simulation of nonlinear dynamics.

The main contribution of the paper is a coarse-to-refined quantum-assisted parameter estimation strategy that reduces the number of expensive data-assimilation solves while still allowing a refined binary search over the parameter domain. The method provides a systematic connection between nudging-based inverse problems, quadratic surrogate modeling, QUBO formulations, Ising Hamiltonians, and quantum optimizers. Theoretical results were also presented to clarify the effect of surrogate approximation error and refined-grid discretization on the recovered parameter estimate. These results show that, when the surrogate uniformly approximates the refined-grid target cost functional and the grid resolution is sufficiently fine, the minimizer of the QUBO surrogate provides a near-optimal estimate for the underlying discrete parameter estimation problem.

Numerical experiments demonstrated the performance of the proposed framework on four representative systems: the SIS and SIR epidemic models, the Lorenz–63 system, and the two-layer Lorenz–96 system. For the SIS model, the recovered parameters had relative errors of about 1.48% for β and 2.70% for γ . For the SIR model, the relative errors were below 1% for both parameters. In the Lorenz–63 example, where only the $x(t)$ component was observed, the parameter errors were larger, as expected for a chaotic and partially observed system, but the recovered trajectories still reproduced the overall Lorenz attractor structure. In the two-layer Lorenz–96 example, the two damping parameters $d_{u,5}$ and $d_{u,15}$ were recovered with relative errors of about 1.33%, using observations only from $u_5(t)$ and $u_{15}(t)$.

An important feature of these experiments is that the estimation procedure uses only partial and time-discrete observations. The SIS and SIR examples use infected-prevalence observations rather than full-state observations. The Lorenz–63 experiment uses only the $x(t)$ component to estimate all three model parameters, and the Lorenz–96 experiment uses only two observed slow variables, $u_5(t)$ and $u_{15}(t)$, to estimate the corresponding damping parameters. Moreover, the observations are available only at finitely many time points, not continuously in time. Thus, the results demonstrate that the proposed framework can be used in realistic observation settings where only selected components of the state are measured at discrete sampling times. Additional numerical tests, not reported here, indicated that increasing the number of observation time points generally improved the accuracy of the recovered parameter values. This suggests that the amount and temporal resolution of the available data can influence the accuracy of the proposed parameter estimation framework.

The IBM Kingston experiments show that the approach can be executed on real quantum hardware, although the recovered estimates may be affected by sampling noise and hardware noise. The SQA results provide an additional comparison for the same QUBO energy landscape. The computational time of the final optimization stage depends on the selected solver and its execution settings. The classical setup stage, including data-assimilation solves, surrogate construction, QUBO fitting, circuit construction, and transpilation, remains part of the total computational cost. Therefore, runtime comparisons should be interpreted in the context of the full hybrid workflow.

Another important point is that the present proof-of-concept experiments use a relatively small number of qubits. The two-parameter examples (SIS, SIR, Lorenz–96) use 10 binary variables, corresponding to five bits per parameter, while the Lorenz–63 example uses 15 binary variables for three unknown parameters. Even with this modest number of qubits, the refined search corresponds to $32^2 = 1024$ candidate parameter values for two-parameter problems and $32^3 = 32768$ candidate parameter values for the three-parameter Lorenz–63 problem. This illustrates the potential benefit of the binary encoding: increasing the number of available qubits would allow

finer parameter grids, larger parameter spaces, or both. Thus, as quantum hardware improves, the same framework could support higher-resolution searches and more complex inverse problems.

Future work will focus on improving the scalability, robustness, and practical applicability of the framework. In particular, we plan to develop adaptive coarse-grid sampling and local refinement strategies so that data-assimilation solves are concentrated in promising regions with small cost functional values, especially when the cost functional landscape is nonquadratic, multimodal, or difficult to resolve on a uniform grid. We also plan to explore more expressive surrogate models, such as higher-order, piecewise, sparse, or physics-informed surrogates, while preserving a QUBO-compatible formulation. Another important direction is to study scalability with respect to the number of unknown parameters and the number of bits used per parameter, since larger quantum devices would allow finer parameter grids and higher-dimensional inverse problems. It will also be important to investigate identifiability more systematically for partially observed dynamical systems, including the chaotic Lorenz-63 system and the high-dimensional two-layer Lorenz-96 system, and to assess the effect of noisy observations, model error, correlated parameters, hardware noise, queue time, circuit depth, sampling variability, and shot count. More broadly, applying the framework to noisy real-world data, larger-scale dynamical systems, and PDE-constrained inverse problems would further clarify the practical role of quantum optimization in parameter estimation.

Acknowledgments. The authors gratefully acknowledge support from the University of Maryland, Baltimore County (UMBC) Strategic Awards for Research Transitions (START) grant. The quantum computing experiments reported in this work used IBM Quantum services, including simulator and hardware access. The SQA experiments were performed using the D-Wave Ocean software and its `PathIntegralAnnealingSampler`.

REFERENCES

- [1] W. O. Kermack and A. G. McKendrick. A contribution to the mathematical theory of epidemics. *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character*, 115(772):700–721, 08 1927.
- [2] H. W. Hethcote. The mathematics of infectious diseases. *SIAM Review*, 42(4):599–653, 2000.
- [3] E. N. Lorenz. Deterministic nonperiodic flow. *Journal of Atmospheric Sciences*, 20(2):130–141, 1963.
- [4] E. N. Lorenz. Predictability: A problem partly solved. In *Proceedings of the Seminar on Predictability*, volume 1, Reading, United Kingdom, 1996.
- [5] C. Sparrow. *The Lorenz Equations: Bifurcations, Chaos, and Strange Attractors*. Springer, 1982.
- [6] Z. Li, M. R. Osborne, and T. Prvan. Parameter estimation of ordinary differential equations. *IMA Journal of Numerical Analysis*, 25(2):264–285, 2005.
- [7] W. Roda, M. Varughese, D. Han, and M. Li. Why is it difficult to accurately predict the COVID-19 epidemic? *Infectious Disease Modelling*, 5, 03 2020.
- [8] H. T. Banks and K. Kunisch. *Estimation Techniques for Distributed Parameter Systems*. Birkhäuser, Boston, MA, 1989.
- [9] U. M. Ascher. *Numerical Methods for Evolutionary Differential Equations*, volume 5 of *Classics in Applied Mathematics*. SIAM, Philadelphia, PA, 2008.
- [10] A. Tarantola. *Inverse Problem Theory and Methods for Model Parameter Estimation*. SIAM, Philadelphia, PA, 2005.
- [11] E. Farhi, J. Goldstone, and S. Gutmann. A quantum approximate optimization algorithm. *arXiv preprint arXiv:1411.4028*, 2014.
- [12] J. Preskill. Quantum computing in the NISQ era and beyond. *Quantum*, 2:79, 2018.
- [13] A. Lucas. Ising formulations of many np problems. *Frontiers in Physics*, 2:5, 2014.

- [14] P. H. Boersch-Supan, S. J. Ryan, and L. R. Johnson. deBinfer: Bayesian inference for dynamical models of biological systems. *Methods in Ecology and Evolution*, 8:511–518, 2017.
- [15] S. Springer, H. Haario, J. Susiluoto, A. Bibov, A. Davis, and Y. Marzouk. Efficient Bayesian inference for large chaotic dynamical systems. *Geoscientific Model Development*, 14(7):4319–4333, 2021.
- [16] O. Ghasemi, M. L. Lindsey, T. Yang, N. Nguyen, Y. Huang, and Y.-F. Jin. Bayesian parameter estimation for nonlinear modelling of biological pathways. *BMC Systems Biology*, 5(Suppl 3):S9, 2011.
- [17] J. Newey, J. P. Whitehead, and E. Carlson. Model discovery on the fly using continuous data assimilation. *Journal of Computational Physics*, 537:114121, 2025.
- [18] E. Carlson, J. Hudson, A. Larios, V. R. Martinez, E. Ng, and J. P. Whitehead. Dynamically learning the parameters of a chaotic system using partial observations. *Discrete and Continuous Dynamical Systems*, 42(8):3809–3839, 2022.
- [19] M. J. Ahmad, A. Biswas, and K. Hoffman. A data-assimilation-augmented optimization framework for parameter estimation in dynamical systems. *arXiv preprint arXiv:2606.29291*, 2026.
- [20] Vincent R Martinez, Jacob Murri, and Jared P Whitehead. Relaxation-based schemes for on-the-fly parameter estimation in dissipative dynamical systems. *Inverse Problems*, 41(5):055001, 2025.
- [21] Y. Yang and H. Li. Neural ordinary differential equations for robust parameter estimation in dynamic systems with physical priors. *Applied Soft Computing*, 169:112649, 2025.
- [22] M. J. Ahmad and K. Günel. Can neural networks estimate parameters in epidemiology models using real observed data? *Applied Intelligence*, 55(2):133, 2024.
- [23] Q. He, L. Wang, and B. Liu. Parameter estimation for chaotic systems by particle swarm optimization. *Chaos, Solitons & Fractals*, 34(2):654–661, 2007.
- [24] Vighnesh V Alavani, Palla Danumjaya, and Padmanabhan Seshaiyer. Fractional-order prey-predator models with parameter estimation via fractional physics-informed neural networks (fpinns). *Journal of Machine Learning for Modeling and Computing*, 7(1), 2026.
- [25] H. Jo, K. Josić, and J. K. Kim. Neural network-based parameter estimation for nonautonomous differential equations with discontinuous signals. *SIAM Journal on Applied Mathematics*, 86(1):327–347, 2026.
- [26] K. J. H. Law, A. M. Stuart, and K. C. Zygalakis. *Data Assimilation: A Mathematical Introduction*. Springer, Cham, 2015.
- [27] E. Kalnay. *Atmospheric Modeling, Data Assimilation and Predictability*. Cambridge University Press, 2003.
- [28] M. Asch, M. Bocquet, and M. Nodet. *Data Assimilation: Methods, Algorithms, and Applications*, volume 11 of *Fundamentals of Algorithms*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 2016.
- [29] A. Carrassi, M. Bocquet, L. Bertino, and G. Evensen. Data assimilation in the geosciences: An overview of methods, issues, and perspectives. *WIREs Climate Change*, 9(5):e535, 2018.
- [30] A. Azouani, E. Olson, and E. S. Titi. Continuous data assimilation using general interpolant observables. *Journal of Nonlinear Science*, 24(2):277–304, 2014.
- [31] D. W. Berry, A. M. Childs, A. Ostrander, and G. Wang. Quantum algorithm for linear differential equations with exponentially improved dependence on precision. *Communications in Mathematical Physics*, 356(3):1057–1081, 2017.
- [32] A. M. Childs, J.-P. Liu, and A. Ostrander. High-precision quantum algorithms for partial differential equations. *Quantum*, 5:574, 2021.
- [33] J.-P. Liu, H. O. Kolden, H. K. Krovi, N. F. Loureiro, K. Trivisa, and A. M. Wang. Efficient quantum algorithm for dissipative nonlinear differential equations. *Proceedings of the National Academy of Sciences*, 118(35):e2026805118, 2021.
- [34] S. Jin, N. Liu, and Y. Yu. Time complexity analysis of quantum algorithms via linear representations for nonlinear ordinary and partial differential equations. *Journal of Computational Physics*, 487:112149, 2023.
- [35] S. Jin, N. Liu, and Y. Yu. Quantum simulation of partial differential equations via Schrödingerization. *Physical Review Letters*, 133(23):230602, 2024.
- [36] O. Kyriienko, A. E. Paine, and V. E. Elfving. Solving nonlinear differential equations with differentiable quantum circuits. *Physical Review A*, 103(5):052416, 2021.
- [37] M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, Cambridge, 10th anniversary edition, 2010.
- [38] T. Kadowaki and H. Nishimori. Quantum annealing in the transverse ising model. *Physical Review E*, 58(5):5355–5363, 1998.
- [39] T. Albash and D. A. Lidar. Adiabatic quantum computation. *Reviews of Modern Physics*,

- 90(1):015002, 2018.
- [40] E. Farhi, J. Goldstone, and S. Gutmann. A quantum approximate optimization algorithm. *arXiv preprint*, 2014.
- [41] Y. Ruan et al. Quantum optimization for constrained combinatorial problems. *Information Sciences*, 2023.
- [42] J. Leng, E. Hickman, J. Li, and X. Wu. Quantum hamiltonian descent. *arXiv preprint arXiv:2303.01471*, 2023.
- [43] J. Leng, Y. Zheng, Z. Jia, C. Zhao, Y. Peng, and X. Wu. Quantum hamiltonian descent for non-smooth optimization. *arXiv preprint arXiv:2503.15878*, 2025.
- [44] S. Kushnir, J. Leng, Y. Peng, L. Fan, and X. Wu. QHDOPT: A software for nonlinear optimization with quantum hamiltonian descent. *INFORMS Journal on Computing*, 37(1):107–124, 2024.
- [45] M. Mohammadisiahroudi, R. Fakhimi, and T. Terlaky. Efficient use of quantum linear system algorithms in inexact infeasible IPMs for linear optimization. *Journal of Optimization Theory and Applications*, pages 1–38, 2024.
- [46] M. Mohammadisiahroudi, B. Augustino, P. Sampourmahani, and T. Terlaky. Quantum computing inspired iterative refinement for semidefinite optimization. *Mathematical Programming*, 2025.
- [47] M. S. Kaznady and D. F. V. James. Quantum state tomography: “the best” is the enemy of “good enough”. *arXiv preprint arXiv:0809.2376*, 2008.
- [48] J. A. Nelder and R. Mead. A simplex method for function minimization. *Computer Journal*, 7:308–313, 1965.
- [49] D. A. Belsley, E. Kuh, and R. E. Welsch. *Regression Diagnostics: Identifying Influential Data and Sources of Collinearity*. Wiley, 1980.
- [50] G. H. Golub and C. F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, 4 edition, 2013.
- [51] Å. Björck. *Numerical Methods for Least Squares Problems*. SIAM, 1996.
- [52] R. M. O’Brien. A caution regarding rules of thumb for variance inflation factors. *Quality & Quantity*, 41(5):673–690, 2007.
- [53] K. P. Vatcheva, M. Lee, J. B. McCormick, and M. H. Rahbar. Multicollinearity in regression analyses conducted in epidemiologic studies. *Epidemiology*, 6(2):227, 2016.