Discover Quantum Science

BRIEF REPORT Open Access

Quantum distortion model for running variational quantum algorithms without error corrections

Ammar Daskin[1*]

*Correspondence:
Ammar Daskin
adaskin25@gmail.com
[1]Department of Computer Engineering, Istanbul Medeniyet University, 34000 Istanbul, Turkey

Abstract

The classical distortion models similarly to temporal data analysis provide a way to predict the trends in the output of the algorithms and discard the anomalies in the output which may impact the accuracy of the method. In this paper, we introduce a quantum distortion modeling framework that enables variational quantum algorithms to operate effectively in the presence of errors without traditional error correction. Drawing inspirations from classical distortion-tolerant computing, we describe different distortions measures and formulation which can be used in variational quantum algorithms. In particular, we develop mathematical foundations for distortion metrics including energy progression, parameter stability, and state fidelity distortions, and demonstrate their applicability to variational quantum eigensolver, quantum approximate optimization algorithm, and quantum power method. We believe this paper will provide a milestone for distortion-aware quantum computing which can expand the practical applicability of the pre-fault-tolerant era devices to problems where approximate solutions provide sufficient value, representing a paradigm shift from exact error elimination to managed accuracy degradation.

Keywords Variational quantum algorithms, Quantum optimization, Quantum distortion model, Quantum error correction, Quantum error mitigation

1 Introduction

Quantum computers are inherently probabilistic machines that generates classical output with probabilities. While they may solve computationally challenging problems that are intractable for classical computers, current technologies, in the noisy intermediate-scale quantum (NISQ) era, operate with quantum gates and measurements that are inherently error-prone due to decoherence, control imperfections, and environmental noise [1]. To address these challenges and manage errors, two main commonly used approaches are quantum error correction (QEC) [2, 3], which provides a pathway to fault-tolerant quantum computation but requires substantial qubit overhead, and quantum error mitigation (QEM) [4–6], which employs statistical and algorithmic techniques

© The Author(s) 2026. **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

to reduce errors in expectation values without the qubit overhead of full error correction. Recent advances in quantum error mitigation include dynamical decoupling [7], proposed for decoherence in open quantum systems; zero-noise extrapolation [8]; probabilistic error cancellation [9]; learning-based mitigation techniques [10]; machine learning-driven methods [11] and artificial neural network-based approaches [12]; and variational techniques [13] that adopt a dynamic strategy for error mitigation. While these and similar methods have shown substantial improvements in algorithmic performance on NISQ devices, they typically operate under the assumption that the objective is to recover the exact, error-free computational result. However, in many practical applications-particularly in optimization and approximation algorithms-users may be willing to tolerate a degree of inaccuracy in exchange for reduced computational cost or improved resilience to failure.

This perspective, combined with the inherent-probabilistic nature of quantum computing, finds a parallel in classical computing, where probabilistic accuracy bounds and distortion models have been successfully employed to ensure computational resilience in the presence of errors and faults. Rinard [14] introduced a formal distortion model in which computations are partitioned into independent tasks so that tasks that encounter errors can be discarded or their effect can be minimized in the model. That means the results of the computations are accepted or can be corrected within the model as long as the distortion remains within acceptable bounds. The concept of intentionally tolerating computational inaccuracy to enhance robustness, reduce execution time, or lower resource consumption is well-established in fault-tolerant computing (e.g. fault tolerant neural networks [15] or see approximate computing survey [16] for further examples). This paradigm can be distilled into following key components [14]:

1. Task Decomposition: The computation is partitioned into task blocks, where each block represents an atomic unit of computation that can be independently executed and, If necessary, discarded: That means if there is a failure or error in one particular task-block that impacts the output of the whole computation, then this block can be discarded to minimize the error in the output. We should note here that the task decomposition is used in many different areas such as distributed computing (e.g. [17]) and machine learning (e.g. [18]) and may not be easy in every case since it would require new algorithmic paradigms.
2. Criticality Testing: Through systematic sampling, every task-block is classified as either critical or failable: Indicating whether failure in the block leads to some unacceptable distortion or is within acceptable distortion range.
3. Probabilistic Distortion Modeling: Using regression techniques on empirical data from previous runs or historical data of the current run, the output distortion is modeled as a function of task failure rates:

$$\hat{d}(x_1, \ldots, x_n) = c_0 \pm e_0 + \sum_{i=1}^{n} (c_i \pm e_i) x_i, \tag{1}$$

where x_i denotes the failure rate of task block i, c_i are regression coefficients, and e_i represent confidence intervals. This is similar to time series analysis such as autoregressive models of temporal data which can be seen in finance and other areas [19].

4. Timing Modeling: The execution time is modeled as a function of failure rates in order to enable and analyze explicit trade-offs between accuracy and performance. This can be also used to design a type of selective execution model used in deep neural networks [20] where only a selective part of the model is run based on the input.

The distortion-tolerant computing framework integrates techniques from statistical learning [21] and regression analysis [22] to construct predictive models of output quality [23]; methods from approximate computing [24, 25], which trade computational accuracy for improvements in performance, energy efficiency, or resource utilization-such as loop perforation and precision scaling; and principles of algorithm resilience [26, 27], which emphasize designing algorithms that produce acceptable results even in the presence of errors like rounding or approximation. Probabilistic bounds in distortion models are grounded in concentration inequalities [28], such as Hoeffding's inequality [29], which provides rigorous guarantees on the deviation of the sample mean from its expected value.

Since the distortion model offers key advantages by enabling explicit trade-offs between accuracy and execution time, it can be used to determine which hardware faults and software errors to tolerate or which tasks can be intentionally discarded in order to reduce execution time while maintaining acceptable accuracy. The distortion model and related approximate computing techniques have found successful applications across various domains: In computer graphics and image processing, algorithms often discard computations associated with pixels or triangles that contribute minimally to final visual quality [30], thereby enabling real-time rendering of complex scenes by concentrating computational resources on perceptually significant elements [31]. Large-scale search engines and recommendation systems [32] employ approximate indexing and retrieval techniques [33] to achieve substantial improvements in response time and resource efficiency. Scientific simulations frequently utilize adaptive precision and approximate numerical methods [34, 35] to balance computational cost with solution accuracy, especially in parameter studies where exact solutions are not essential. In distributed computing environments, where partial failures are common, task discarding and recomputation serve as fundamental fault-tolerance mechanisms, as exemplified by MapReduce [36].

1.1 Contribution

Although classical distortion models operate under assumptions that may not directly translate to quantum computing, they offer several insights that are particularly relevant to the quantum domain:

- Error Classification: Differentiating between critical and failable tasks to identify which quantum operations are most sensitive to errors and which can be approximated or skipped. As an example quantum operations written as a sum of permutations with positive coefficients are error-resilient to bit flips [37].
- Probabilistic Guarantees: Applying statistical bounds to quantify the output quality of quantum computations and measurements.
- Resource-Accuracy Tradeoffs: Explicitly modeling the trade-offs between accuracy and performance to manage the limited resources available on near-term quantum devices.

- Robust Algorithm Design: Establishing design principles that enhance the robustness of quantum algorithms against the error characteristics of quantum hardware.

By bridging classical distortion modeling with the unique characteristics of quantum algorithms, this work provides a practical pathway for enhancing the applicability of variational quantum algorithms to real-world problems in the pre-fault-tolerant era. Specifically, we adapt the classical distortion model [14] to develop a primary mathematical framework for characterizing and bounding output distortions in variational quantum algorithms. The key contributions of this paper are as follows:

- While prior works have explored quantum-to-classical rate distortion coding [38], signal distortion correction in quantum sensing [39], and quantum state redistribution among multiple parties [40], to the best of our knowledge, this paper is the first to introduce a quantum distortion modeling framework that adapts classical concepts of probabilistic accuracy bounds [14] to the quantum domain. Although related to error mitigation techniques and machine learning based error correction approaches, our framework provides a way to consider how variational quantum algorithms can withstand hardware faults and software errors by giving priority to quantitative guarantees on output quality, rather than striving for complete error elimination.
- We formulate a generic mathematical distortion model for variational quantum algorithms that captures the evolution of inaccuracies across iterations. This model can be used to detect anomalies in output and optimization parameters. To quantify distortion, we introduce rigorous formulations for quantum-specific metrics, including a composite distortion measure that integrates multiple metrics with configurable weights: energy progression distortion for identifying optimization anomalies, parameter convergence distortion for monitoring landscape smoothness, and state fidelity distortion for tracking quantum state degradation.
- We develop specialized distortion models for three widely used variational algorithms in quantum optimization: the variational quantum eigensolver (VQE), the quantum approximate optimization algorithm (QAOA), and the variational quantum power method (VQPM). These models provide a foundational framework for applying distortion-aware principles across diverse quantum computing applications.

The remainder of this paper is organized as follows: Sect. 2 introduces the general framework and formulates distortion measures. Section 3 discusses how these model can be trained and validated. Section 4 presents specialized distortion models for the variational quantum power method (VQPM), variational quantum eigensolver (VQE), and quantum approximate optimization algorithm (QAOA). Section 5 provides a numerical simulation for VQE to illustrate the practical application of the proposed framework. Finally, Sect. 6 discusses the results, outlines limitations of the model, gives some future research directions, and concludes the paper.

2 Distortion model framework for variational quantum algorithms

In this section, we present a mathematical framework for modeling distortion in variational quantum algorithms, drawing primarily from classical distortion modeling concepts [14, 22]. In particular, to capture the evolution of quantum states under imperfect and error-prone conditions, we adapt the general classical distortion formulations to the

quantum domain. Since quantum operations may include both unitary and non-unitary processes, instead of using unitary operator, we employ quantum channels [41] to represent the full range of quantum dynamics, including error processes. Additionally, we use the density matrix formalism [42, 43] to describe mixed states that arise due to decoherence and imperfect operations.

We lay the foundation of quantum distortion model based on the evolution of quantum states under noisy conditions. The quantum state at iteration t can be represented by a density matrix ρ_t. Its evolution can be modeled as:

$$\rho_t = \mathcal{E}_t(\rho_{t-1}, \theta_t, \epsilon_t), \tag{2}$$

where θ_t denotes the variational parameters at iteration t, ϵ_t captures error parameters such as gate infidelities, decoherence, and measurement noise, and $\mathcal{E}_t$ is the quantum channel representing the algorithmic iteration, incorporating both ideal operations and error processes. The primary focus of classical distortion models lies in analyzing convergence patterns. Following this philosophy, we define a relative distortion metric that quantifies deviations from expected convergence behavior:

$$D_t = \sum_{i=1}^{k} w_i \cdot d_i(\rho_t, \rho_{t-1}, \ldots, \rho_{t-m}), \tag{3}$$

where d_i denotes component distortion measures, w_i are their associated weights satisfying $\sum_{i=1}^{k} w_i = 1$, and m specifies the window size for historical context. In many practical applications where the exact solution is unknown, similar relative metrics with historical context are used to measure deviation from failure-free execution [44] or to predict future behavior, as seen in time series analysis models [19].

2.1 Possible component distortion measures

We can define several component distortion measures, d_is, in order to capture different aspects of quantum algorithm performance under noisy conditions. These measures can be combined into an overall distortion metric using weighted contributions. For example, given distortion components such as energy progression distortion d_E, state fidelity distortion d_F, and parameter convergence distortion d_θ, the overall distortion at iteration t can be expressed as:

$$D_t = w_E d_E(t) + w_F d_F(t) + w_\theta d_\theta(t) + \cdots, \tag{4}$$

where the weights $w_E, w_F, w_\theta, \ldots$ determine the relative importance of each component in the overall distortion measure, and satisfy $\sum w_i = 1$.

2.1.1 Energy progression distortion

In many variational quantum algorithms, the energy-typically the lowest eigenvalue-is formulated as the optimal solution to the underlying problem. The algorithms we consider in this paper include the variational quantum eigensolver (VQE) [45], quantum approximate optimization algorithm (QAOA) [46], and other related methods such as the variational quantum power method (VQPM) [47, 48] and variational imaginary time evolution algorithm [49].

As a generic model for these variational algorithms, we can define the energy-based distortion at iteration t as:

$$d_E(t) = \left| \frac{E_t - \hat{E}_t}{\sigma_E} \right|, \tag{5}$$

where E_t is the observed energy, and $\hat{E}_t$ is the predicted energy obtained via linear regression over recent energy values: $\hat{E}_t = \alpha + \beta \cdot t$, with regression coefficients α and β computed from the historical window $\{E_{t-m}, \ldots, E_{t-1}\}$. The normalization term σ_E is the standard deviation of energy values in the same window:

$$\sigma_E = \sqrt{\frac{1}{m-1} \sum_{i=t-m}^{t-1} (E_i - \bar{E})^2}, \tag{6}$$

where $\bar{E} = \frac{1}{m} \sum_{i=t-m}^{t-1} E_i$ is the mean energy over the window. The distortion measure in Equation (5) detects anomalies in energy convergence, which may indicate error accumulation or convergence to suboptimal local minima. The normalization by σ_E ensures scale invariance, making the metric robust across different problem instances.

2.1.2 State fidelity distortion

The stability and quality of quantum state evolution can be quantified using fidelity [50]. The convergence of this algorithms is done through the convergence of the state to the eigenstate. This convergence behavior in general is gradual process and without sudden changes. Therefore, we can also define a fidelity-based distortion at iteration t to capture sudden changes in the fidelity as:

$$d_F(t) = 1 - F(\rho_t, \rho_{t-1}) = 1 - \left(\mathrm{Tr} \sqrt{\sqrt{\rho_{t-1}}\, \rho_t \, \sqrt{\rho_{t-1}}} \right)^2, \tag{7}$$

where $F(\rho, \sigma)$ is the fidelity between states ρ and σ [50, 51]. Fidelity satisfies $F(\rho, \sigma) = F(\sigma, \rho)$, $F(\rho, \sigma) \in [0, 1]$, $F(\rho, \sigma) = 1$ iff $\rho = \sigma$, and $F(\rho, \sigma) = 0$ for orthogonal states [43]. Fidelity is non-decreasing under any completely positive trace-preserving map $\mathcal{E}$: $F(\mathcal{E}(\rho), \mathcal{E}(\sigma)) \geq F(\rho, \sigma)$.

For pure states $\rho = |\psi\rangle\langle\psi|$ and $\sigma = |\phi\rangle\langle\phi|$,

$$d_F(t) = 1 - |\langle\psi_t|\psi_{t-1}\rangle|^2. \tag{8}$$

When ρ and σ are diagonal in the same basis,

$$\rho = \sum_i p_i |i\rangle\langle i|, \quad \sigma = \sum_i q_i |i\rangle\langle i|, \quad F(\rho, \sigma) = \left(\sum_i \sqrt{p_i q_i} \right)^2. \tag{9}$$

Computing fidelity exactly is infeasible for large systems due to exponential scaling. In practice fidelity can be estimated via the swap test [52] or classical-shadow techniques that reduce measurement overhead [53].

As indicated, tn variational algorithms, the evolution of $d_F(t)$ is a useful diagnostic: Because, $\lim_{t\to\infty} d_F(t) = 0$ indicates stable convergence and sudden drops in fidelity

typically signal significant errors, unwanted state transitions, convergence to incorrect solutions, or hardware-induced state corruption.

2.1.3 Parameter stability distortion

Parameter stability reflects the smoothness of the optimization landscape [54]. Large, erratic parameter changes may indicate numerical instability, poor gradient estimates, or the presence of barren plateaus. We can define parameter distortion as:

$$d_\theta(t) = \frac{\|\theta_t - \theta_{t-1}\|}{\|\theta_{t-1}\|}. \tag{10}$$

This metric can also help to distinguish between healthy optimization progress and problematic behavior. Note that, to avoid division by zero, one can use $\|\theta_{t-1}\| + \delta$ with a small $\delta > 0$.

2.1.4 Convergence rate distortion

For certain variational quantum algorithms such as QAOA [46] or VQPM [48], the expected convergence rate r_{expected} can be derived from theoretical analysis or estimated from similar failure-free executions. To capture deviations from expected convergence behavior-analogous to classical methods that monitor numerical optimization progress [55]-we can define the following convergence-based distortion measure:

$$d_C(t) = \begin{cases} \left| \frac{\Delta E_t}{\Delta E_{t-1}+\delta} - r_{\text{expected}} \right| & \text{if } |\Delta E_{t-1}| > \delta_{\min} \\ 0 & \text{otherwise} \end{cases}, \tag{11}$$

where $\Delta E_t = E_t - E_{t-1}$, and δ is a small constant used to avoid division by zero.

Here, note that additional distortion measures may also be formulated based on other quantum-specific metrics, such as entanglement measures, quantum Fisher information, or other indicators commonly used in quantum computing.

2.2 Failure-incorporated evolution model

While relative distortion metrics provide essential monitoring capabilities, the underlying failure mechanisms that cause distortion can also be explicitly incorporated into the model. Quantum hardware errors can be linked to the distortion model to predict distortion under various error conditions. This mirrors the classical approach that relates task failure rates to output distortion [14], and is also similar to error mitigation techniques applied to hybrid variational algorithms [56].

To explicitly model the impact of errors, we can decompose the quantum channel into ideal and failed components as:

$$\rho_t = (1 - \epsilon_t) \cdot \rho_t^{\text{ideal}} + \epsilon_t \cdot \rho_t^{\text{failed}}, \tag{12}$$

where the total failure rate ϵ_t aggregates error contributions from different sources-primarily gate error rates ϵ_g, measurement errors ϵ_m, and decoherence effects ϵ_d. The failure rate ϵ_t can be estimated from device calibration data or theoretical error models (e.g., [4]). This formulation can also be expressed using quantum channel operators $\mathcal{E}_t(\dots)$, which is fundamentally equivalent.

2.3 Distortion-failure relationship

Building upon the failure-incorporated evolution model, we can also establish a quantitative relationship between failure rates and observed distortion. This relationship is crucial for predicting how specific error conditions affect algorithmic performance and for managing detected distortions. The prediction part can be done by extending the classical regression approach to quantum algorithms. In particular, the relationship between distortion and failure rates at iteration t can be modeled by using the total observed distortion D_t, measured from actual algorithm execution with failures, and the inherent algorithmic distortion $D_t^{\text{algorithm}}$, representing distortion that would occur even in a perfect, failure-free execution due to factors such as incomplete convergence, approximation errors, or suboptimal parameter choices:

$$D_t = D_t^{\text{algorithm}} + \Delta D_t^{\text{failure}}, \tag{13}$$

where $\Delta D_t^{\text{failure}}$ represents the additional distortion caused by failures. This failure-induced distortion can be modeled as:

$$\Delta D_t^{\text{failure}} = \alpha \cdot \epsilon_t + \beta \cdot \epsilon_t \cdot D_t^{\text{algorithm}} + \gamma \cdot \epsilon_t^2, \tag{14}$$

where the term $\alpha \cdot \epsilon_t$ captures baseline distortion caused by failures, independent of algorithmic state; $\beta \cdot \epsilon_t \cdot D_t^{\text{algorithm}}$ reflects how failures interact with the current algorithmic state-when the algorithm is already struggling (high $D_t^{\text{algorithm}}$), the same failures may cause more severe distortion; and $\gamma \cdot \epsilon_t^2$ accounts for potential quadratic effects where failure combinations lead to compounded distortion. For practical purposes, a simplified linear model may also be employed:

$$D_t = (1 + \beta \cdot \epsilon_t) \cdot D_t^{\text{algorithm}} + \alpha \cdot \epsilon_t. \tag{15}$$

This formulation can be interpreted as a linear combination of the inherent algorithmic distortion with a baseline component. Note that the coefficients α, β, and γ can be learned from training data or derived from theoretical considerations. This captures both the direct impact of failures (linear term) and their interaction with algorithmic convergence (interaction term). In practice, $D_t^{\text{algorithm}}$ can be estimated from simulation results on noiseless quantum simulators, theoretical convergence bounds for the specific algorithm, empirical baseline performance on simple problem instances, or extrapolation from early iterations before significant failure accumulation.

2.4 Dynamic threshold model for recovery triggers

Static distortion thresholds may be insufficient for variational quantum algorithms because we expect many early jumps at the early iterations and small changes in the later iterations in the convergence of these algorithms. Therefore, it may be necessary to evolve distortion levels throughout the optimization process. Using the previously established distortion-failure relationship, we can address the question of when to intervene in algorithm execution by modeling a dynamic threshold as follows:

$$\tau_t = \tau_{\min} + (\tau_{\max} - \tau_{\min}) \cdot \exp(-\lambda t) + \eta \cdot \epsilon_t, \tag{16}$$

where τ_t is the dynamic distortion threshold at iteration t; $\tau_{\min}$ and $\tau_{\max}$ are the minimum (strictest bound) and maximum (most lenient bound) acceptable distortion

thresholds, respectively; and λ is the decay rate controlling how quickly the threshold tightens. The term $\tau_{\min} + (\tau_{\max} - \tau_{\min}) \cdot \exp(-\lambda t)$ defines the expected distortion reduction as the algorithm converges, while the term $\eta \cdot \epsilon_t$ adjusts the threshold based on failure rates, with η representing the failure tolerance coefficient. In conclusion, this adjustment in a sense acknowledges that some additional distortion is unavoidable when failure rates are high. The decreasing threshold reflects the increasing expectation of convergence as the algorithm progresses, similar to classical approaches that adjust tolerance levels during computation [56]. The exponential decay provides a smooth transition from lenient early thresholds to strict late thresholds, while the derivative condition helps prevent unnecessary recovery actions during normal convergence.

Based on this threshold, we can determine when a recovery is triggered by using the changes in this threshold model:

$$D_t > \tau_t \quad \text{and} \quad \frac{dD_t}{dt} > \kappa \cdot \frac{d\tau_t}{dt}, \tag{17}$$

where κ is a rate sensitivity parameter, typically chosen such that $\kappa > 1$.

2.5 Further extensions to model

2.5.1 Correlation-aware distortion for multi-qubit systems

Quantum systems exhibit complex correlations due to entanglement and coherent error propagation. These correlations imply that distortions in different algorithm components can interfere with each other in non-trivial ways. To capture these quantum-specific effects, a correlation-aware distortion model-extending classical multivariate analysis [57]-can be defined for multi-qubit systems by using the following matrix formulation:

$$\mathrm{D}_t = \mathrm{w}^\top \mathrm{d}_t + \mathrm{d}_t^\top \mathrm{C} \mathrm{d}_t + \mathrm{f}_t^\top \mathrm{R} \mathrm{f}_t, \tag{18}$$

where $\mathrm{d}_t = [d_1(t), d_2(t), \ldots, d_m(t)]^\top$ is the component distortion vector of size $m \times 1$; w is the linear weight vector of size $m \times 1$, representing independent contributions; C is the distortion correlation matrix of size $m \times m$, capturing cross-component distortion interactions; f_t is the failure vector of size $p \times 1$, representing different failure types; and R is the failure correlation matrix of size $p \times p$, capturing correlated failure patterns.

For an n-qubit system with k distortion metrics per qubit, we have $m = n \times k$, and C is symmetric with diagonal elements C_{ii} representing self-correlation and off-diagonal elements C_{ij} capturing distortion coupling between components i and j. The values of C_{ij} can indicate different types of correlation, such as strong anti-correlation when $C_{ij} < 0$, or crosstalk correlations when $C_{ij} > 0$. The matrix R encodes spatial and temporal failure correlations, such as crosstalk between adjacent qubits or correlated measurement errors.

This formulation basically extends classical multivariate models to quantum systems by modeling entanglement effects through the quadratic term $\mathrm{d}_t^\top \mathrm{C} \mathrm{d}_t$ in order to emulate the interaction of distortions in entangled subsystems. Additionally, the failure correlation term addresses hardware-specific error patterns like crosstalk [58]. The matrix structure thus enables scalable distortion computation for large multi-qubit quantum systems.

2.5.2 Time-evolution as a dynamical system

While the distortion models presented so far provide static snapshots, as stated before, quantum algorithm execution is fundamentally dynamical (based on unitary time evolution or underlying Hamiltonian). To enable real-time distortion management and predictive control, distortion evolution can also be modeled as a dynamical system. This approach allows the application of control theory principles, such as adaptive control methods used in optimization [59], to actively steer the algorithm away from high-distortion trajectories. In particular, the temporal evolution of distortion can be modeled using a discrete-time dynamical system [60, 61]:

$$\mathrm{D}_{t+1} = \mathrm{A}\mathrm{D}_t + \mathrm{B}\mathrm{u}_t + \mathbf{\Gamma}\epsilon_t + \eta_t, \tag{19}$$

where D_t is the distortion state vector of size $m \times 1$ at iteration t; A is the state transition matrix of size $m \times m$, capturing natural distortion dynamics; B is the control input matrix of size $m \times r$, mapping control actions to distortion changes; u_t is the control vector of size $r \times 1$, representing algorithmic adjustments; $\mathbf{\Gamma}$ is the failure sensitivity matrix of size $m \times p$; ϵ_t is the failure vector of size $p \times 1$; and η_t is the process noise vector of size $m \times 1$, representing unmodeled effects.

This dynamical approach in a sense enables forecasting of future distortion based on the current state, supports the derivation of control policies that minimize distortion, and facilitates analysis of algorithm convergence under various control strategies. These strategies may be encoded in the control vector u_t, incorporating parameter adjustments $\Delta\theta_t$, changes in measurement strategy, or other error mitigation techniques.

2.5.3 Probabilistic bounds using concentration inequalities

Since the randomness of quantum measurements and the statistical nature of error processes can be used to draw probabilistic guarantees. We can augment the models presented so far with probabilistic bounds derived from different concentration inequalities. For instance, individual distortion components can be bounded using Hoeffding's inequality [29], which provides a probability bound for the deviation from the expected mean value of random variables bounded by a_i and b_i:

$$P\left(|\hat{d}_i - \mathbb{E}[d_i]| \geq \delta\right) \leq 2\exp\left(-\frac{2M\delta^2}{(b_i - a_i)^2}\right), \tag{20}$$

where M is the number of measurement shots. Building on this, the total distortion can be bounded using the multivariate Berry-Esseen theorem in higher dimensions [62], applied to the distortion vector:

$$P\left(\|\hat{\mathrm{D}}_t - \mathbb{E}[\mathrm{D}_t]\| \geq \delta\right) \leq \frac{C}{\sqrt{M}} + \text{higher-order terms}. \tag{21}$$

Total distortion can also be quantified through confidence intervals. For the estimated distortion $\hat{D}_t(\epsilon)$ at iteration t, with failure rates $\epsilon = [\epsilon_g, \epsilon_m, \epsilon_d]$ representing gate, measurement, and decoherence errors, the confidence interval can be expressed using the standard error σ_t of the distortion estimate:

$$\hat{D}_t(\epsilon) \in \left[\hat{D}_t - z_{\alpha/2} \cdot \sigma_t, \hat{D}_t + z_{\alpha/2} \cdot \sigma_t\right], \tag{22}$$

where $z_{\alpha/2}$ is the critical value and α is the significance level from the standard normal distribution. The standard error σ_t quantifies uncertainty in the distortion estimate and is given by $\sigma_t = \frac{s_t}{\sqrt{M}}$, where s_t is the sample standard deviation of distortion measurements and M is the number of measurement shots or independent trials. Here, this standard error decreases as $1/\sqrt{M}$ with more measurements and may include contributions from multiple sources. For example, it may combine measurement shot noise $\sigma_t^{\text{shot}} \propto 1/\sqrt{M}$, algorithmic uncertainty σ_t^{alg} due to variational parameter sensitivity, and hardware variability $\sigma_t^{\text{hardware}}$ from device calibration fluctuations. In such cases, the total uncertainty can be expressed as:

$$\sigma_t = \sqrt{(\sigma_t^{\text{shot}})^2 + (\sigma_t^{\text{alg}})^2 + (\sigma_t^{\text{hardware}})^2}. \tag{23}$$

3 Training and validation framework for variational quantum algorithms

The distortion models can be deployed as a learned framework by using historical data. This requires a specialized training and validation setup that accounts for the characteristics of quantum optimization landscapes, data availability, and device-specific error profiles. Here, classical machine learning methodologies can be adapted while considering these constraints.

Training such models begins with a preliminary data collection phase. This datasets can be from smaller systems or previous runs, or previous iterations in the current run. Datasets from multiple sources for all considered distortion measures can be used to capture diverse error conditions as follows:

$$\mathcal{D}_{\text{train}} = \left\{ (\epsilon^{(i)}, \mathrm{D}^{(i)}, \theta^{(i)}, t^{(i)}) \right\}_{i=1}^{N}, \tag{24}$$

where $\epsilon^{(i)} = [\epsilon_g^{(i)}, \epsilon_m^{(i)}, \epsilon_d^{(i)}]$ denotes gate, measurement, and decoherence error rates, respectively; $\mathrm{D}^{(i)}$ is the observed distortion vector at parameter set $\theta^{(i)}$ for algorithm iteration $t^{(i)}$.

As stated, these datasets can be generated using noisy quantum simulations with systematically varied error models on smaller systems, historical hardware executions across multiple quantum processors, controlled failure injection during algorithm optimization, or transfer learning from classical approximations of quantum dynamics.

For training, a regularized regression or other machine learning techniques can be applied to variational quantum algorithms. As an example by incorporating optimization landscape knowledge along with the training data, we can use the following regularized regression model:

$$\min_{\mathrm{w,C,R}} \sum_{i=1}^{N} \left(\mathrm{D}^{(i)} - \hat{\mathrm{D}}(\epsilon^{(i)}, \theta^{(i)}, t^{(i)}; \mathrm{w}, \mathrm{C}, \mathrm{R}) \right)^2 + \lambda_1 R_{\text{landscape}}(\theta) + \lambda_2 \|\mathrm{C}\|_*, \tag{25}$$

where $R_{\text{landscape}}(\theta)$ is a landscape-aware regularizer that penalizes models contradicting known variational optimization properties, such as smoothness in parameter space or convergence guarantees.

For validation, classical strategies can be adapted with slight modifications. As stated before; training on smaller problem instances and validating on larger ones, or

evaluating generalization across different quantum processors, can help ensure robustness and transferability of the learned distortion models.

4 Specified models for example variational algorithms

4.1 Variational quantum eigensolver (VQE)

The Variational Quantum Eigensolver (VQE) is a hybrid quantum-classical algorithm [54, 63] designed to find the ground state energy of quantum systems. Although this algorithm can be used for any eigenvalue-related problem, it is particularly applied to quantum chemistry and material science problems, where VQE employs a parameterized quantum circuit (ansatz) to prepare trial states. The energy expectation values of these states are measured and fed to a classical optimizer for iterative improvement.

A VQE-specific distortion model can be defined using the components in the state evolution formula, which is given by:

$$\rho_t^{\mathrm{VQE}} = U(\theta_t)\rho_0 U^\dagger(\theta_t), \tag{26}$$

with energy expectation $E_t = \mathrm{Tr}[H\rho_t^{\mathrm{VQE}}]$, where H is the underlying Hamiltonian and U is the parameterized unitary.

Considering each component in this formulation, the combined distortion measure can be expressed as:

$$D_t^{\mathrm{VQE}} = w_E \cdot d_E(t) + w_G \cdot d_G^{\mathrm{VQE}}(t) + w_W \cdot d_W^{\mathrm{VQE}}(t) + w_\Delta \cdot d_\Delta^{\mathrm{VQE}}(t), \tag{27}$$

where each term defines a different type of distortion. The energy distortion $d_E(t)$ is given in (5). The energy gradient distortion is defined as:

$$d_G^{\mathrm{VQE}}(t) = \left\| \nabla E_t - \nabla \hat{E}_t \right\|, \tag{28}$$

where $\nabla \hat{E}_t$ is the predicted gradient from previous iterations. A normalized version can also be used:

$$d_G^{\mathrm{VQE}}(t) = \frac{\|\nabla E_t - \nabla \hat{E}_t\|}{\|\nabla \hat{E}_t\|}. \tag{29}$$

The parameter update distortion $d_\Delta^{\mathrm{VQE}}(t)$ is defined as:

$$d_\Delta^{\mathrm{VQE}}(t) = \frac{\|\theta_t - \theta_{t-1}\|}{\|\theta_{t-1} - \theta_{t-2}\|} \cdot \mathbb{I}_{\{\|\theta_{t-1}-\theta_{t-2}\|>\delta\}}, \tag{30}$$

where $\mathbb{I}$ is the indicator function.

The wavefunction overlap (fidelity) distortion is given by:

$$d_W^{\mathrm{VQE}}(t) = 1 - |\langle\psi(\theta_t)|\psi(\theta_{t-1})\rangle|^2. \tag{31}$$

Near convergence, as $t \to \infty$ and $\theta_t \to \theta^*$, the total distortion can be approximated by:

$$D_t^{\mathrm{VQE}} \approx \|\theta_t - \theta^*\| \cdot \|\nabla^2 E(\theta^*)\|. \tag{32}$$

4.2 Quantum approximate optimization algorithm (QAOA)

The Quantum Approximate Optimization Algorithm (QAOA) [46] is a quantum algorithm that finds or approximates solutions for combinatorial optimization problems through an alternating sequence of problem-specific and mixer Hamiltonians with classically optimized parameters. The generic formulation of QAOA is defined with a depth parameter p as:

$$|\psi_t\rangle = \prod_{k=1}^{p} e^{-i\beta_k^{(t)} H_M} e^{-i\gamma_k^{(t)} H_C} |+\rangle^{\otimes n}. \tag{33}$$

As in VQE, we define distortion measures for each component and combine them to formulate the total distortion:

$$D_t^{\mathrm{QAOA}} = w_E \cdot d_E(t) + w_P \cdot d_P^{\mathrm{QAOA}}(t) + w_R \cdot d_R^{\mathrm{QAOA}}(t) + w_V \cdot d_V^{\mathrm{QAOA}}(t). \tag{34}$$

Here, the component-wise distortions include the layer-wise parameter distortion:

$$d_P^{\mathrm{QAOA}}(t) = \frac{1}{p} \sum_{k=1}^{p} \left[\left(\frac{\beta_k^{(t)} - \beta_k^{(t-1)}}{\beta_k^{(t-1)}} \right)^2 + \left(\frac{\gamma_k^{(t)} - \gamma_k^{(t-1)}}{\gamma_k^{(t-1)}} \right)^2 \right]^{1/2}. \tag{35}$$

The distortion in the approximation ratio is given by:

$$d_R^{\mathrm{QAOA}}(t) = \left| \frac{E_t - E_{\min}}{E_{\max} - E_{\min}} - \frac{E_{t-1} - E_{\min}}{E_{\max} - E_{\min}} \right|. \tag{36}$$

The constraint violation distortion (for constrained problems) is defined as:

$$d_V^{\mathrm{QAOA}}(t) = \sum_j w_j \cdot |C_j(\psi_t)|, \tag{37}$$

where C_j are constraint functions.

In QAOA, successful convergence behavior typically involves increasing the depth parameter p along with problem size. For increasing depth, the distortion scales as:

$$D_t^{\mathrm{QAOA}} \sim \frac{1}{\sqrt{p}} \cdot D_t^{\mathrm{base}}. \tag{38}$$

4.3 Variational quantum power method (VQPM)

The Variational Quantum Power Method (VQPM) adapts classical power iteration for quantum optimization of QUBO problems [47, 48]. It features a qubit locking mechanism that exponentially reduces the search space and accelerates convergence by fixing qubit values when their probability differences exceed a threshold.

The state evolution in VQPM with qubit locking can be defined as:

$$\rho_t^{\mathrm{VQPM}} = \mathcal{L}_{S_t} \circ \mathcal{U}_Q \circ \mathcal{P}_{\theta_t}(\rho_{t-1}), \tag{39}$$

where $\mathcal{L}_{S_t}$ is the locking operation on qubit set S_t.

As in the other algorithms, the total distortion is defined as the accumulation of component-wise distortion measures:

$$D_t^{\mathrm{VQPM}} = w_E \cdot d_E(t) + w_L \cdot d_L^{\mathrm{VQPM}}(t) + w_C \cdot d_C^{\mathrm{VQPM}}(t) + w_P \cdot d_P^{\mathrm{VQPM}}(t) + w_\theta \cdot d_\theta(t). \quad (40)$$

The qubit locking distortion is defined as:

$$d_L^{\mathrm{VQPM}}(t) = \frac{1}{|S_t \setminus S_{t-1}|} \sum_{q \in S_t \setminus S_{t-1}} \left(1 - P_{\mathrm{consistency}}(q,t)\right), \quad (41)$$

where the consistency probability of qubit q being in state $|0\rangle$ over the window from iteration $t-k$ to $t-1$ is:

$$P_{\mathrm{consistency}}(q,t) = \exp\left(-\lambda \cdot \mathrm{Var}\{P_q(0)_{t-k:t-1}\}\right). \quad (42)$$

The control qubit distortion is given by:

$$d_C^{\mathrm{VQPM}}(t) = \left| P_{\mathrm{control}}^{(t)}(0) - \frac{1+\cos(\hat{E}_t)}{2} \right|. \quad (43)$$

Finally, the probability distribution distortion is formulated as:

$$d_P^{\mathrm{VQPM}}(t) = \mathrm{JS}\left(\mathrm{p}^{(t)}, \mathrm{p}^{(t-1)}\right), \quad (44)$$

where JS denotes the Jensen-Shannon divergence [64].

5 Numerical experiments with distortion-aware VQE

5.1 Experimental set-up

To evaluate the effectiveness of the proposed quantum distortion model, we conducted a comprehensive series of simulations across multiple configurations. The experiments were designed to test the framework under varying conditions of noise exposure and recovery capabilities. For demonstration purposes, we employed VQE for an 3-qubit Hamiltonian and 7-qubit Hamiltonian generated using X, Y, and Z neighbor interactions with specified interaction coefficients. In addition, we employed three distinct noise patterns: short (2 noisy iterations per 10 iterations), medium (6 noisy iterations per 10 iterations), and long (9 noisy iterations per 10 iterations) noise durations. For each configuration, we compared four scenarios: (1) ideal execution on a noiseless device, (2) ideal execution with distortion monitoring and recovery, (3) noisy execution without recovery, and (4) noisy execution with distortion-aware recovery.

Furthermore, the recovery mechanism itself was tested under two distinct conditions: recovery operations performed on a noiseless device (simulating access to higher-fidelity quantum operations) and recovery performed on the same noisy device (representing more realistic hardware constraints). This allows us to distinguish between the fundamental capability of the model to detect anomalies and the practical effectiveness of recovery operations under different hardware conditions.

For the simulation, we used the Pennylane software package and apply different noise to each qubit after both single and entangling gates in every layer. Some of the other simulation settings are given in Table 1. Here note that the code used for the simulation is publicly available for download from the link.[1]

[1] The simulation code: https://github.com/adaskin/quantum-distortion-model

Table 1 Simulation parameters for distortion-aware VQE experiments

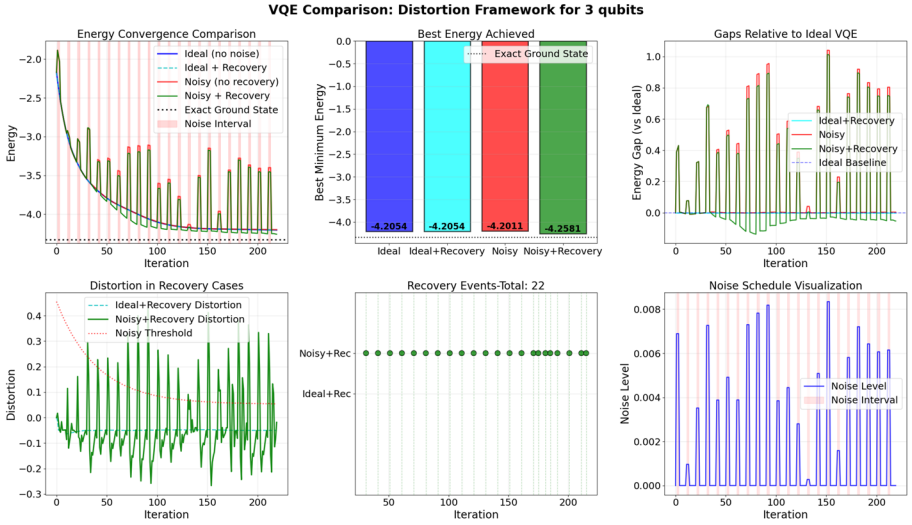
Parameter	Value/Configuration
Distortion range	$[-1, 0)$: Good distortion (energy improving) (0, 1]: Bad distortion (energy degrading)
Distortion weights	Energy: 0.85, Fidelity: 0.05, Parameter stability: 0.05, Convergence: 0.05
Distortion threshold	Dynamic: $\tau_{\min} + (\tau_{\max} - \tau_{\min}) \cdot e^{-\lambda t}$ High in early iterations, decreasing exponentially
Recovery strategy	Four-tiered approach based on distortion magnitude: • Severe (> 0.5): Reset with best parameters (50:50 blend) • Moderate (0.3-0.5): Blend best and smoothed parameters (50:50) • Mild (0.1-0.3): Small adjustment toward best (90:10) • Very mild (0.0-0.1): Minimal adjustment (95:05)
Learning rate	0.01 (fixed throughout optimization)
Qubit count	3 and 7 (two separate experiments)
Ansatz layers	3 layers of parameterized rotations and entangling gates
Recovery device	Noisy or noiseless (PennyLane `default.mixed`)
Noise model	After each single- and two-qubit gate: • Bit flip: `qml.BitFlip(noise_level)` • Phase damping: `qml.PhaseDamping(noise_level)` • Amplitude damping: `qml.AmplitudeDamping(noise_level)` • Depolarizing: `qml.DepolarizingChannel(noise_level)`
Noise schedule	Periodic noisy intervals: • Short: 2 iterations every 10 iterations • Medium: 6 iterations every 10 iterations • Long: 9 iterations every 10 iterations • Random noise level for each interval

The distortion range is normalized to $[-1, 1]$, where negative values indicate beneficial behavior (energy improving) and positive values indicate problematic behavior (energy degrading). Recovery strategies are triggered based on distortion magnitude relative to dynamic thresholds

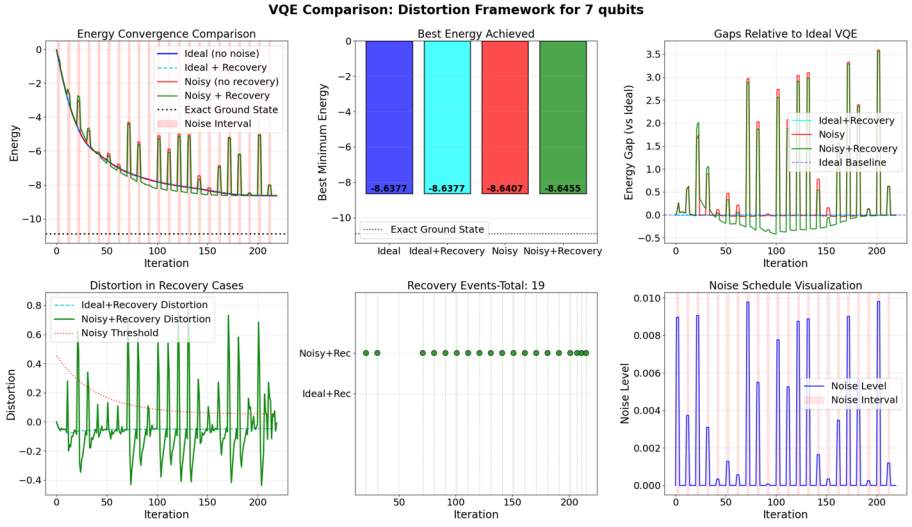
5.2 Key findings

The experimental results are given in Figs. 1, 2, 3, 4, 5, 6 which reveal several important patterns. First, when recovery operations can be performed on a noiseless device (Figs. 1, 2, 3 for short, medium, and long noise bursts), the distortion model demonstrates remarkable robustness across all noise durations and in some cases improves the optimization result over the ideal case (Here note that it is already known some noise in VQE are beneficial for optimization progress [65, 66]). In addition, even under near-continuous noise (9 out of 10 iterations), the system successfully detects anomalous behavior and triggers recovery operations which fully restore optimization progress. This is particularly evident in Fig. 3, where the"noisy + recovery"trajectories closely follow the ideal case for both 3-qubit and 7-qubit systems, regardless of noise duration.

However, as shown in Figs. 4, 5, 6 a critical limitation emerges when recovery operations are themselves performed on noisy hardware. For short noise intervals (2/10 iterations), the recovery mechanism remains functional, though with slightly reduced effectiveness compared to noiseless recovery. For medium noise durations (6/10 iterations), recovery operations provide only marginal improvement over the noisy baseline. Most strikingly, for long noise durations (9/10 iterations), the recovery mechanism offers virtually no benefit when performed on noisy hardware. Therefore, the effectiveness of recovery diminishes significantly with increasing noise duration. This highlights the significance of the fidelity of the hardware used to execute recovery operations.



(a) 3-qubit system. The distortion-recovery model successfully mitigates noise effects, maintaining performance close to the ideal case despite periodic noise intervals.

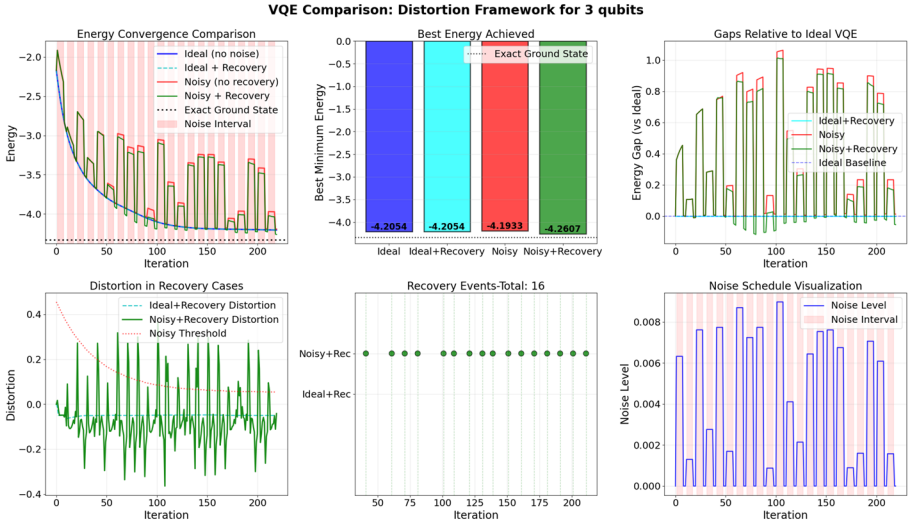


(b) 7-qubit system. The increased system size amplifies noise effects, yet the recovery mechanism preserves optimization trajectory and final energy accuracy.

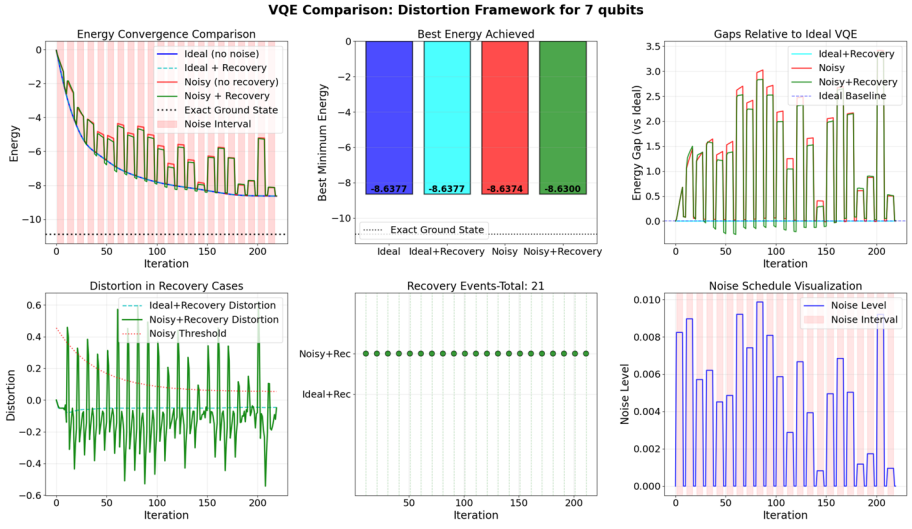
Fig. 1 Comparative analysis of distortion-aware VQE performance across four scenarios: (1) ideal noiseless execution, (2) ideal case with distortion monitoring and recovery, (3) noisy execution without recovery, and (4) noisy execution with distortion-recovery. Both systems experience short (2-iteration) noise bursts every 10 iterations with random noise levels. Recovery operations are performed on a noiseless device. The results demonstrate the effectiveness of the distortion model in detecting anomalies and triggering parameter recovery to maintain optimization progress

6 Discussion and future work

Although there are some quantum studies that describe error-resilient models (e.g., [37]) and machine learning based approaches for quantum control and output estimation [67, 68], there remains a gap in the research on such frameworks. The quantum distortion model introduced in this paper represents a paradigm shift from conventional error correction protocols. Its similarity to temporal data analysis techniques used in other fields provides both theoretical and practical grounding for designing distortion measurements and recovery schemes tailored to variational quantum algorithms. Rather than pursuing exact error elimination, the method enables approximate solutions and can be



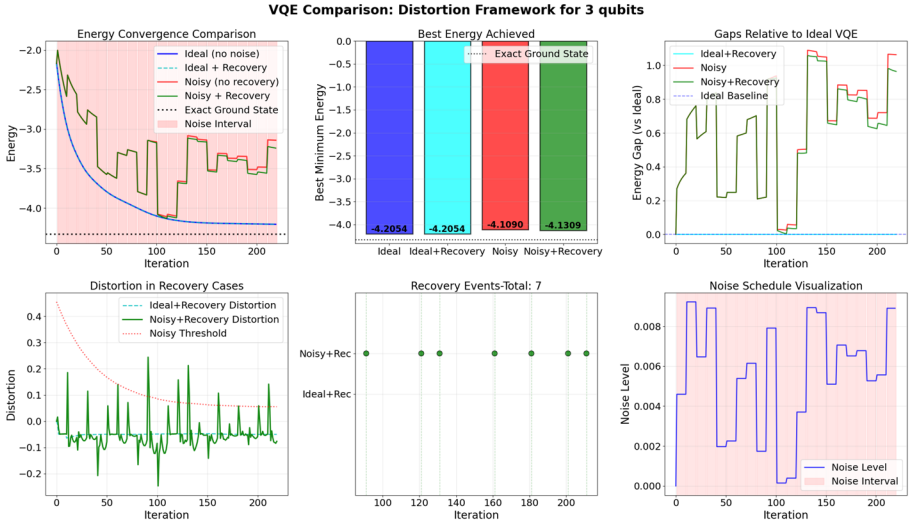
(a) 3-qubit system with medium noise duration. The distortion-recovery model effectively compensates for moderate noise exposure, maintaining convergence near the ideal trajectory.



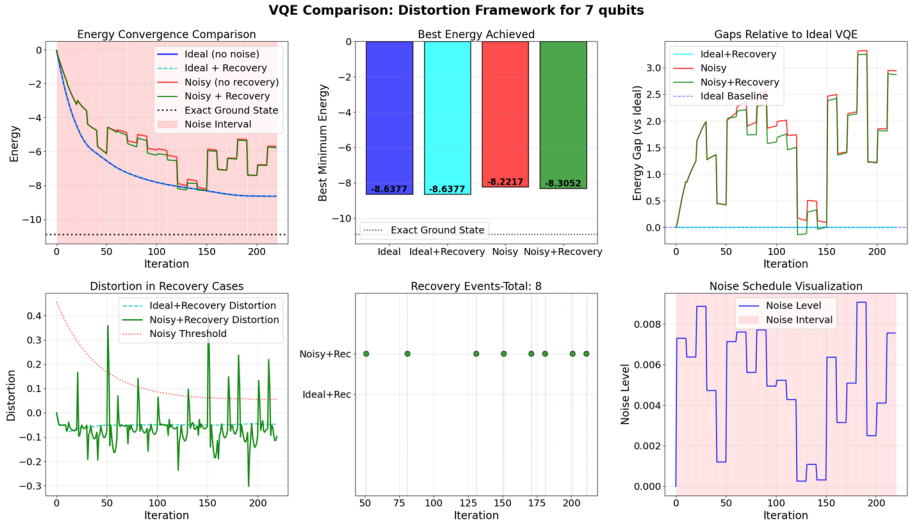
(b) 7-qubit system with medium noise duration. Despite increased system complexity, the recovery mechanism successfully counteracts noise effects when performed on a noiseless device.

Fig. 2 VQE optimization under medium-duration noise (6 noisy iterations every 10 iterations) with noiseless device recovery. The distortion model successfully detects anomalies during extended noise intervals and triggers recovery operations that restore optimization progress. Both system sizes demonstrate resilience to moderate noise exposure when recovery operations can be performed on a clean device

integrated with established quantum error mitigation (QEM) techniques such as zero-noise extrapolation [8] and probabilistic error cancellation [4], thereby enhancing the utility of current quantum devices for real-world applications. Moreover, the advantage of the quantum distortion model lies in its resource efficiency and its applicability to current technologies, where resources remain limited for large-scale computations. In domains where exact solutions are often unnecessary or infeasible, and algorithms are typically defined by parameterized circuits; probabilistic distortion bounds can be used to guarantee accuracy bounds for applications in optimization, machine learning, and approximate sampling. Moreover, the framework provides foundational tools which can be extended to other quantum computing contexts.



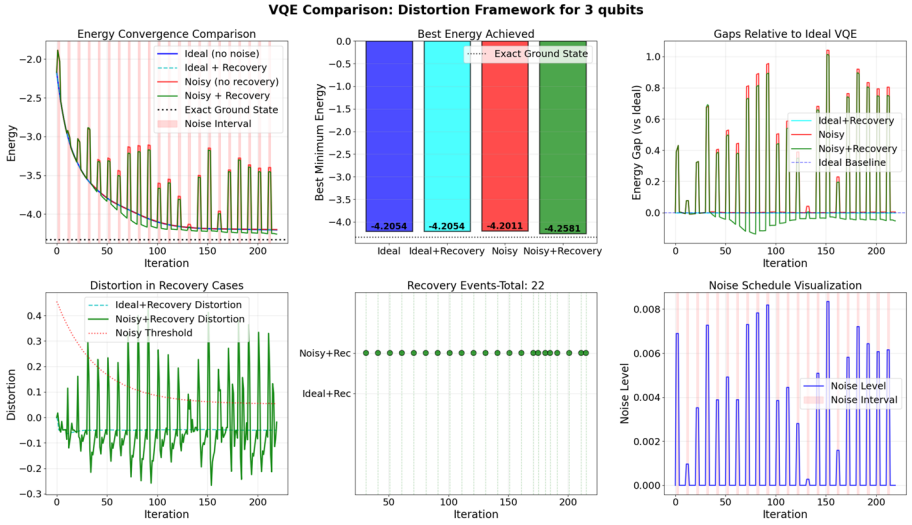
(a) 3-qubit system with long noise duration. The recovery model maintains functionality despite persistent noise, though performance degradation is more pronounced than with shorter noise intervals.



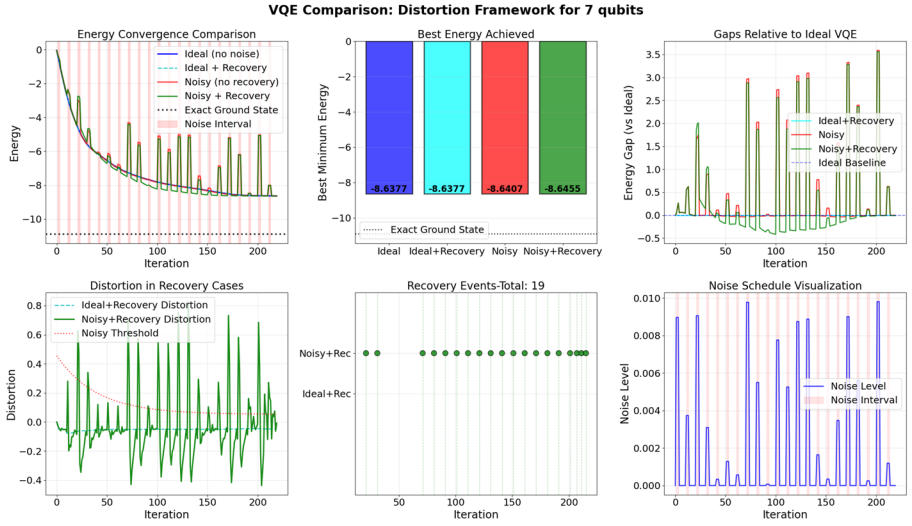
(b) 7-qubit system with long noise duration. The recovery mechanism continues to operate effectively, demonstrating the model's robustness even under near-continuous noise conditions.

Fig. 3 VQE optimization under long-duration noise (9 noisy iterations every 10 iterations) with noiseless device recovery. Despite near-continuous noise exposure, the distortion model successfully identifies problematic behavior and triggers recovery. This demonstrates the framework's effectiveness when recovery operations can access a noiseless device, even under extreme noise conditions

As demonstrated in the numerical experiments, the successful recovery under noiseless conditions demonstrates that the core principles studied in this paper are sound: By monitoring multiple distortion metrics and triggering parameter adjustments when thresholds are exceeded, it is possible to maintain optimization progress despite substantial noise exposure. The failure of noisy recovery under extended noise conditions suggests that the recovery operations themselves must be relatively error-free to be effective. This implies that practical implementations of distortion-aware quantum algorithms will require either (1) occasional access to higher-fidelity quantum operations specifically for recovery, or (2) additional error mitigation techniques applied to recovery operations.



(a) 3-qubit system with short noise and noisy recovery. Recovery operations remain effective despite being performed on noisy hardware, demonstrating robustness to mild noise conditions.



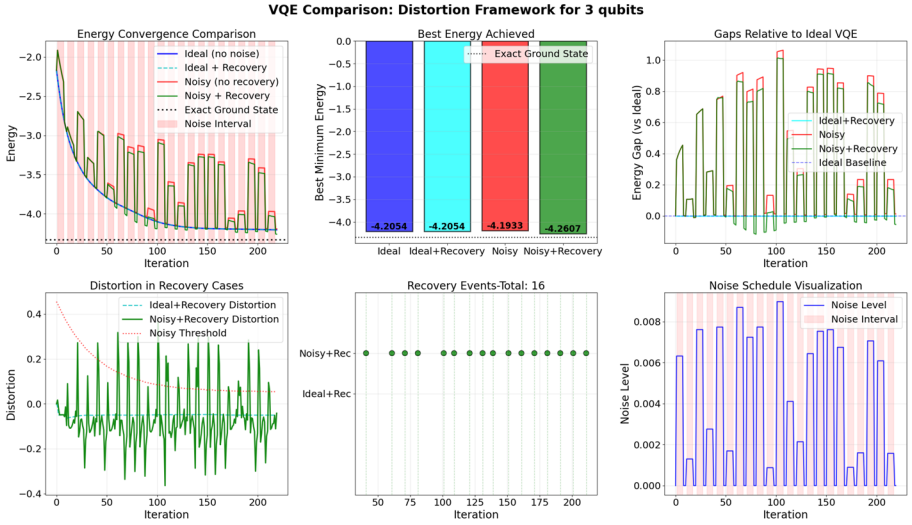
(b) 7-qubit system with short noise and noisy recovery. The recovery mechanism continues to function, though with slightly reduced effectiveness compared to noiseless recovery.

Fig. 4 VQE optimization under short-duration noise (2 noisy iterations every 10 iterations) with noisy device recovery. When recovery operations are performed on the same noisy hardware, the distortion model remains effective for short noise intervals. This represents a more realistic scenario where clean hardware is unavailable

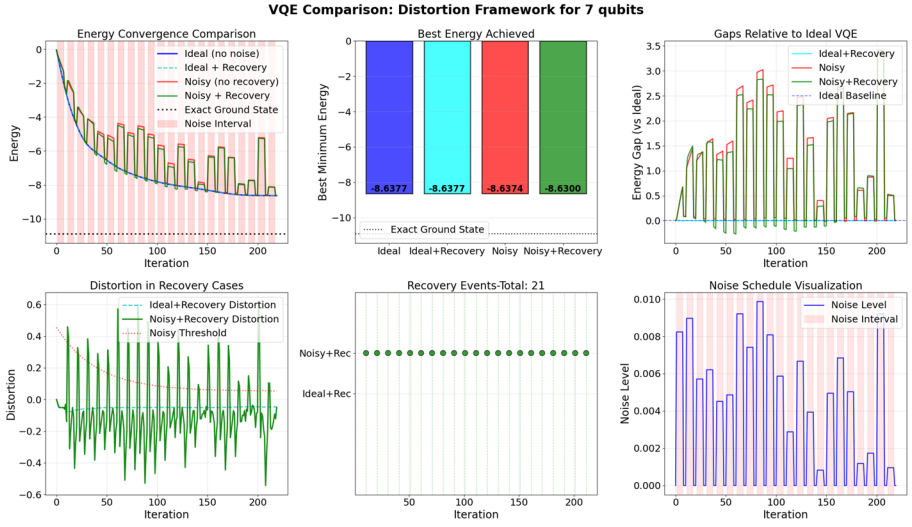
Since the 7-qubit systems exhibit similar patterns to the 3-qubit systems, the approach may generalize to larger problem instances. However, larger systems are more sensitive to noise because of the small spectral gap. Therefore, application to larger systems may require careful parameter adjustments.

6.1 Some limitations

Quantum distortion modeling provides a viable approach for maintaining optimization progress in noisy quantum environments, but with important practical constraints. As reported in the experiments, the framework successfully detects noise-induced anomalies and triggers appropriate recovery actions. However, the ultimate effectiveness of these recoveries depends on the fidelity of the hardware available for executing them.



(a) 3-qubit system with medium noise and noisy recovery. Recovery effectiveness diminishes as noise exposure increases, highlighting the limitations of performing recovery on noisy hardware.

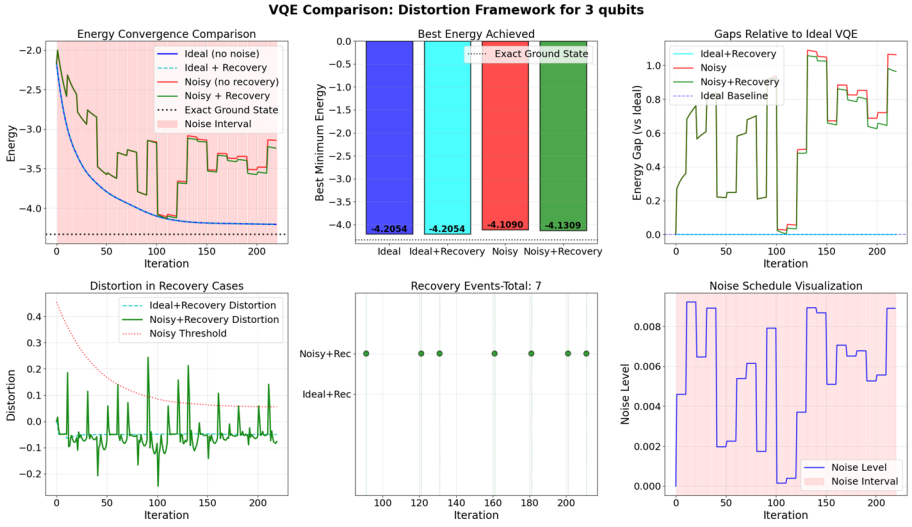


(b) 7-qubit system with medium noise and noisy recovery. The recovery mechanism struggles to counteract noise effects when both optimization and recovery operations are performed on noisy hardware.

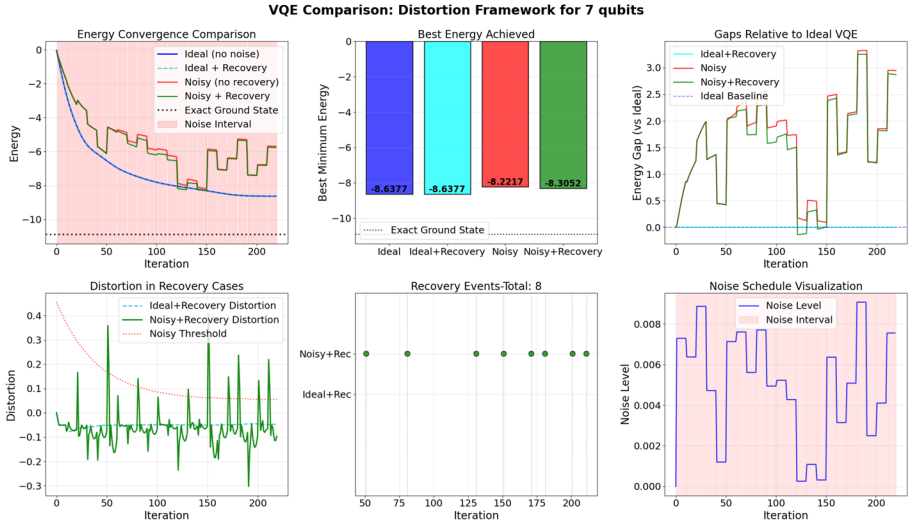
Fig. 5 VQE optimization under medium-duration noise (6 noisy iterations every 10 iterations) with noisy device recovery. The effectiveness of recovery operations degrades significantly when performed on noisy hardware. Distortion detection remains possible, but noisy recovery operations may introduce additional errors that compromise their effectiveness

This points to a hybrid architecture where most computations occur on noisy quantum hardware, but occasional recovery operations are performed on higher-fidelity quantum processors or with additional error mitigation. Future work should explore intermediate approaches where recovery operations employ more robust parameter adjustment strategies or incorporate additional error mitigation specifically for recovery computations.

Since the distortion model relies on historical context or learned parameters, it assumes similar trends or behaviors to those observed in known applications. This reliance may present limitations: In particular, when the number of iterations is too small to establish a meaningful historical context, or when no prior data from similar applications is available, the model may hinder the algorithm's ability to escape local minima.



(a) 3-qubit system with long noise and noisy recovery. Recovery operations fail to meaningfully improve optimization, demonstrating the breakdown of noisy recovery under persistent noise conditions.



(b) 7-qubit system with long noise and noisy recovery. The recovery mechanism provides minimal benefit, indicating that noisy recovery cannot overcome continuous hardware noise.

Fig. 6 VQE optimization under long-duration noise (9 noisy iterations every 10 iterations) with noisy device recovery. This configuration reveals the fundamental limitation of the distortion model: when recovery operations themselves are performed on noisy hardware and noise levels remain high toward the end of iterations, they cannot effectively counteract persistent noise

Additionally, if the historical context from previous iterations is insufficient, the model's effectiveness may depend on the availability of representative training data and could require retraining for significantly different problem instances or hardware configurations. In addition, when the historical context does not provide any noiseless state-i.e. when the noise is prevalent across the iterations (there is always noise)-, it may not be possible to recover from this noisy state by distortion metrics. We have also observed this behavior in our numerical experiments for VQE.

In addition, when the learning rate high or errors during a steep slope might cause the optimizer to take large steps. If a distortion is detected during such a step, the recovery strategy might reset the parameters to a previous state. In our numerical experiment,

we use best recent parameters in historical window. However, these parameters might be from a very different region of the optimization space and may revert optimization progress back to an earlier point, again losing the progress made. Therefore, one might need to adjust the distortion model to account for the expected energy change based on the step size.

Finally, we also note that our numerical experiments employ synthetic, periodic noise patterns (short, medium, and long bursts) rather than a detailed hardware-driven noise model derived from specific device calibration data. This choice was made to enable a controlled and systematic analysis of the distortion model's response to varying temporal profiles of error accumulation-a key factor in iterative algorithms like VQE-in isolation from the complex, static noise characteristics of any particular quantum processor. While hardware-driven models based on published calibration data (e.g., gate error rates, coherence times, crosstalk) are invaluable for predicting absolute performance on a specific device, our synthetic approach allows us to study the core detection and recovery logic of the framework under clearly defined noise durations and frequencies. Future work will indeed benefit from integrating detailed, device-specific noise models to tailor distortion thresholds and recovery strategies to the unique error profile of target hardware.

7 Conclusion

In conclusion, we have developed a primary quantum distortion model framework that bridges classical concepts of approximate computing and temporal analysis for the variational quantum algorithms. We have introduced various distortion measures and demonstrated their applicability across different scenarios and formulations. The proposed approach is general and adaptable to any quantum variational algorithm. Future work may focus on extending the distortion model to broader classes of quantum algorithms, developing more efficient training methodologies, integrating with existing error mitigation techniques, and providing experimental validation across diverse applications and quantum hardware platforms.

Acknowledgements
In the preparation of this paper, A.D. acknowledges the use of DeepSeek AI system [69] through private chats to proofread paper, improve the readability, and correct and edit some equations.

Author contributions
A.D conceptualized the whole research idea and wrote the manuscript.

Funding
This paper is not funded by any funding agency.

Data availability
The simulation code can be downloaded from a publicly available repository:https://github.com/adaskin/quantum-distortion-model

Declarations

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
The authors declare no Competing interests.

Received: 16 October 2025 / Revised: 6 January 2026 / Accepted: 12 January 2026
Published online: 27 January 2026

References

1. Preskill J. Quantum computing in the nisq era and beyond. Quantum. 2018;2:79.
2. Shor PW. Scheme for reducing decoherence in quantum computer memory. Phys Rev A. 1995;52(4):R2493.
3. Devitt SJ, Munro WJ, Nemoto K. Quantum error correction for beginners. Rep Progress Phys. 2013;76(7):076001.
4. Temme K, Bravyi S, Gambetta JM. Error mitigation for short-depth quantum circuits. Phys Rev Lett. 2017;119(18):180509.
5. Cai Z, Babbush R, Benjamin SC, Endo S, Huggins WJ, Li Y, et al. Quantum error mitigation. Rev Modern Phys. 2023;95(4):045005.
6. Bultrini D, Gordon MH, Czarnik P, Arrasmith A, Cerezo M, Coles PJ, et al. Unifying and benchmarking state-of-the-art quantum error mitigation techniques. Quantum. 2023;7:1034.
7. Viola L, Knill E, Lloyd S. Dynamical decoupling of open quantum systems. Phys Rev Lett. 1999;82(12):2417.
8. Kandala A, Temme K, Córcoles AD, Mezzacapo A, Chow JM, Gambetta JM. Error mitigation extends the computational reach of a noisy quantum processor. Nature. 2019;567(7749):491–5.
9. Van Den Berg E, Minev ZK, Kandala A, Temme K. Probabilistic error cancellation with sparse Pauli–Lindblad models on noisy quantum processors. Nat Phys. 2023;19(8):1116–21.
10. Strikis A, Qin D, Chen Y. Simon C Benjamin, and Ying Li. Learning Based Quantum Error Mitigation PRX Quantum. 2021;2(4):040330.
11. Liao H, Wang DS, Sitdikov I, Salcedo C, Seif A, Minev ZK. Machine learning for practical quantum error mitigation. Nat Mach Intell. 2024;6(12):1478–86.
12. Kim C, Park KD, Rhee JK. Quantum error mitigation with artificial neural network. IEEE Access. 2020;8:188853–60.
13. Ravi GS, Smith KN, Gokhale P, Mari A, Earnest N, Javadi-Abhari A, Chong FT. Vaqem: A variational approach to quantum error mitigation. In 2022 IEEE international symposium on high-performance computer architecture (HPCA), pages 288–303. IEEE, 2022.
14. Rinard M. Probabilistic accuracy bounds for fault-tolerant computations that discard tasks. In Proceedings of the 20th annual international conference on Supercomputing, pages 324–334, 2006.
15. Traiola M, Pappalardo S, Piri A, Ruospo A, Deveautour B, Sanchez E, Bosio A, Saeedi S, Carpegna A, Savino A. Approximate fault-tolerant neural network systems. In 29th IEEE European Test Symposium (ETS 2024). IEEE, 2024.
16. Leon V, Hanif MA, Armeniakos G, Jiao X, Shafique M, Pekmestzi K, et al. Approximate computing survey, part i: terminology and software & hardware approximation techniques. ACM Comput Surv. 2025;57(7):1–36.
17. Cai J, Liu W, Huang Z, Yu FR. Task decomposition and hierarchical scheduling for collaborative cloud-edge-end computing. IEEE Trans Serv Comput. 2024;17(6):4368–82.
18. Yang J, Xuanqi Liu Yu, Diao XC, Haikuo H. Adaptive task decomposition physics-informed neural networks. Comput Methods Appl Mech Eng. 2024;418:116561.
19. Hamilton JD. Time series analysis. Princeton university press, 2020.
20. Liu L,Deng J. Dynamic deep neural networks: Optimizing accuracy-efficiency trade-offs by selective execution. In Proceedings of the AAAI conference on artificial intelligence, volume 32, 2018.
21. Vapnik V. The nature of statistical learning theory. Springer science & business media, 2013.
22. Freund R,Littell R. SAS system for regression. John Wiley & Sons, 2000.
23. Zhang J, Zhou Y. Calibration procedures for linear regression models with multiplicative distortion measurement errors. Braz J Probab Stat. 2020;34(3):519–36.
24. Mittal S. A survey of techniques for approximate computing. ACM Comput Surv (CSUR). 2016;48(4):1–33.
25. Dalloo AM, Humaidi AJ, Al Mhdawi AK, Al-Raweshidy H. Approximate computing: concepts, architectures, challenges, applications, and future directions. IEEE Access. 2024;12:146022–88.
26. Huang KH, Abraham JA. Algorithm-based fault tolerance for matrix operations. IEEE Trans Comput. 1984;100(6):518–28.
27. Moskalenko V, Kharchenko V, Moskalenko A, Kuzikov B. Resilience and resilient systems of artificial intelligence: taxonomy, models and methods. Algorithms. 2023;16(3):165.
28. Massart P. Some applications of concentration inequalities to statistics. In Annales de la Faculté des sciences de Toulouse Mathématiques. 2000;9:245–303.
29. Hoeffding W. Probability inequalities for sums of bounded random variables. J Am Stat Assoc. 1963;58(301):13–30.
30. Kay TL, Kajiya JT. Ray tracing complex scenes. ACM SIGGRAPH Comput Gr. 1986;20(4):269–78.
31. Akenine-Moller T, Haines E, Hoffman N. Real-time rendering. AK Peters/crc Press, 2019.
32. Sarwar BM, Karypis G, Konstan J, Riedl J. Recommender systems for large-scale e-commerce: Scalable neighborhood formation using clustering. In Proceedings of the fifth international conference on computer and information technology. 2002;1:291–324.
33. Anh VN,Moffat A. Pruned query evaluation using pre-computed impacts. In Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval, pages 372–379, 2006.
34. Baboulin M, Buttari A, Dongarra J, Kurzak J, Langou J, Langou J, et al. Accelerating scientific computations with mixed precision algorithms. Comput Phys Commun. 2009;180(12):2526–33.
35. Abdelfattah A, Anzt H, Boman EG, Carson E, Cojean T, Dongarra J, Gates M, Grützmacher T, Higham NJ, Li S, et al. A survey of numerical methods utilizing mixed precision arithmetic. arXiv preprint arXiv:2007.06674, 2020.
36. Dean J, Ghemawat S. Mapreduce: simplified data processing on large clusters. Commun ACM. 2008;51(1):107–13.
37. Daskin A. Error analysis of quantum operators written as a linear combination of permutations: A. daskin. Quantum Inf Process. 2025;24(5):149.
38. Datta N, Hsieh MH, Wilde MM, Winter A. Quantum-to-classical rate distortion coding. J Math Phys. 2013;54(4).
39. Chaves KR, Wu X, Rosen YJ, DuBois JL. Nonlinear signal distortion corrections through quantum sensing. Appl Phys Lett. 2021;118(1).
40. Khanian ZB,Winter A. A rate-distortion perspective on quantum state redistribution. IEEE Trans Inf Theory, 2024.
41. Caruso F, Giovannetti V, Lupo C, Mancini S. Quantum channels and memory effects. Rev Mod Phys. 2014;86(4):1203–59.
42. Blum K. Density matrix theory and applications, volume 64. Springer Science & Business Media, 2012.
43. Nielsen MA,Chuang IL. Quantum computation and quantum information. Cambridge university press, 2010.
44. Misailovic S, Sidiroglou S, Hoffmann H, Rinard M. Quality of service profiling. In Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering-Volume 1, pages 25–34, 2010.

45. Kandala A, Mezzacapo A, Temme K, Takita M, Brink M, Chow JM, et al. Hardware-efficient variational quantum Eigensolver for small molecules and quantum magnets. Nature. 2017;549(7671):242–6.
46. Farhi E, Goldstone J, Gutmann S. A quantum approximate optimization algorithm. arXiv preprint arXiv:1411.4028, 2014.
47. Daskin A. Combinatorial optimization through variational quantum power method. Quantum Inf Process. 2021;20(10):336.
48. Daskin A. From theory to practice: Analyzing vqpm for quantum optimization of qubo problems. arXiv preprint arXiv:2505.12990, 2025.
49. McArdle S, Jones T, Endo S, Li Y, Benjamin SC, Yuan X. Variational ansatz-based quantum simulation of imaginary time evolution. NPJ Quantum Inf, 2019;5(1):75.
50. Jozsa R. Fidelity for mixed quantum states. J Mod Opt. 1994;41(12):2315–23.
51. Uhlmann A. The transition probability in the state space of a*-algebra. Rep Math Phys. 1976;9(2):273–9.
52. Buhrman H, Cleve R, Watrous J, De Wolf R. Quantum fingerprinting. Phys Rev Lett. 2001;87(16):167902.
53. Huang H-Y, Kueng R, Preskill J. Predicting many properties of a quantum system from very few measurements. Nat Phys. 2020;16(10):1050–7.
54. McClean JR, Romero J, Babbush R, Aspuru-Guzik A. The theory of variational hybrid quantum-classical algorithms. New J Phys. 2016;18(2):023023.
55. Nocedal J, Wright SJ. Numerical optimization. Springer, 2006.
56. Endo S, Cai Z, Benjamin SC, Yuan X. Hybrid quantum-classical algorithms and quantum error mitigation. J Phys Soc Japan. 2021;90(3):032001.
57. Johnson RA, Wichern DW, et al. Applied multivariate statistical analysis. 2002.
58. Sarovar M, Proctor T, Rudinger K, Young K, Nielsen E, Blume-Kohout R. Detecting crosstalk errors in quantum information processors. Quantum. 2020;4:321.
59. Bryson AE. Applied optimal control: optimization, estimation and control. Routledge, 2018.
60. Robinson RC. An introduction to dynamical systems: continuous and discrete, volume 19. Am Math Soc, 2012.
61. Belta C, Yordanov B, Aydin Gol E. Formal methods for discrete-time dynamical systems. Springer, 2017.
62. Chernozhukov V, Chetverikov D, Kato K. Central limit theorems and bootstrap in high dimensions. Ann Probab. 2017;45(4):2309–52.
63. Peruzzo A, McClean J, Shadbolt P, Yung MH, Zhou XQ, Love PJ, Aspuru-Guzik A, O'brien JL. A variational eigenvalue solver on a photonic quantum processor. Nat Commun, 2014;5(1):4213.
64. Briët J, Harremoës P. Properties of classical and quantum Jensen–Shannon divergence. Phys Rev A Atomic Mole Opt Phys. 2009;79(5):052311.
65. Gentini L, Cuccoli A, Pirandola S, Verrucchi P, Banchi L. Noise-assisted variational hybrid quantum-classical optimization. arXiv preprint arXiv:1912.06744, pages 1–8, 2019.
66. Liu J, Wilde F, Mele AA, Jin X, Jiang L, Eisert J. Stochastic noise can be helpful for variational quantum algorithms. Phys Rev A. 2025;111(5):052441.
67. Ma H, Qi B, Petersen IR, Wu RB, Rabitz H, Dong D. Machine learning for estimation and control of quantum systems. Natl Sci Rev. 2025;12(8):nwaf269.
68. Krenn M, Landgraf J, Foesel T, Marquardt F. Artificial intelligence and machine learning for quantum technologies. Phys Rev A. 2023;107(1):010101.
69. DeepSeek Chat. Private ai-assisted communication. Conversation with DeepSeek AI system. 2025.