# Benchmarking Adaptive Genetic Pulse-Level Optimization on Standard Quantum Algorithms

William Aguilar-Calvo
*School of Software Engineering*
*CENFOTEC University*
San José, Costa Rica
waguilarc@ucenfotec.ac.cr

Santiago Núñez-Corrales
*NCSA, IQUIST*
*University of Illinois Urbana-Champaign*
Urbana IL, USA
nunezco2@illinois.edu

***Abstract*—Noise in current quantum hardware (NISQ devices) poses a major obstacle by degrading gate operations and overall circuit fidelity toward FTQC. Traditional quantum error correction (QEC) is effective but resource-intensive, requiring several extra qubits and operations beyond the reach of near-term devices. Quantum error mitigation (QEM) offers a complementary path by reducing the impact of error prior to full correction. We present an adaptive pulse-level QEM using genetic algorithms to dynamically adjust control pulses in real-time, improving algorithm fidelity without modifying the logic structure of a quantum circuit. By targeting pulse parameters directly, this method reduces the impact of various noise sources, improving algorithm resilience in quantum circuits. To demonstrate the effectiveness of our methodology, we apply our protocol to five key quantum algorithms, including Bernstein-Vazirani, Deutsch-Jozsa, Grover's search, Quantum Fourier Transform (QFT), and its inverse (IQFT). Experimental results show that our pulse-level optimization strategy provides a flexible and efficient solution for increasing fidelity during the noisy execution of quantum circuits. Our work contributes to advance error mitigation techniques, essential for robust quantum computing, which will increasingly require HPC infrastructure as quantum systems scale.**

***Index Terms*—adaptive algorithm, NISQ, FTQC, HPC-QPU integration, noise resilience, pulse-level optimization, qubit fidelity, quantum computing, quantum error correction, quantum error mitigation**

## I. Introduction

Quantum computers promise to solve classically intractable problems, from factoring large numbers to simulating complex quantum systems [1]. However, today's devices operate in the Noisy Intermediate-Scale Quantum (NISQ) regime [2], where computation is constrained by short coherence times and gate errors. Quantum noise arises from various sources: e.g. *bit-flip* and *phase-flip* errors randomly flip qubit states or phases [3]. Decoherence processes, characterized by amplitude relaxation ($T_1$) and phase decoherence ($T_2$), degrade quantum information over time [4]. Multi-qubit systems encounter *crosstalk*, where unintended interactions between qubits cause extra errors during the execution of two- and multi-qubit gates [5]. These error mechanisms underscore the urgent need for scalable and robust strategies to mitigate noise in quantum systems. Since physical mechanisms responsible for the susceptibility of a quantum system to intrinsic and external noise sources tend to be harder to address, software-based strategies constitute the center of attention in this article. [6]

Conventional Quantum Error Correction (QEC) methods meet these challenges by encoding logical qubits into entangled states distributed across multiple physical qubits [7]. For instance, a single logical qubit state $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ can be transformed into an encoded state:

$$|\psi\rangle \rightarrow |\psi_L\rangle = \alpha|0_L\rangle + \beta|1_L\rangle,$$

where $|0_L\rangle$ and $|1_L\rangle$ represent logical states encoded across multiple physical qubits. Syndrome measurements identify errors and enable corrective operations to restore $|\psi_L\rangle$ to its intended state [7]. While effective, QEC demands substantial qubit overhead and extremely low physical error rates that are not yet available in NISQ devices. Consequently, near-term quantum computing must rely on methods that mitigate errors without full error correction.

Quantum error mitigation (QEM) techniques seek to reduce the impact of errors, trading increased sampling or computation for improved fidelity [6, 8]. Unlike QEC, QEM does not produce fully error-corrected logical qubits, but instead uses methods such as error extrapolation [9] or probabilistic error cancellation [10] to approximate error-free results. For instance, zero-noise extrapolation involves running circuits at inflated noise levels and extrapolating to zero noise. QEM can significantly improve algorithm outputs on NISQ hardware, but fundamental analyses have shown it cannot indefinitely suppress errors without incurring exponential overhead in sampling as circuits scale [8].

In this work, we build upon QEM by targeting the quantum control pulses that implement gate operations. Pulse-level control allows adjusting the continuous control waveforms driving the qubits' evolution [11]. By fine-tuning pulse amplitudes and durations, one can directly influence the Hamiltonian of the system and counteract noise at its source [6]. Prior quantum control methods like GRAPE and CRAB have optimized pulse shapes offline to achieve high-fidelity gates [12, 13, 14]. However, such methods typically assume a fixed noise model and do not adapt to run-time fluctuations. We propose an Adaptive Genetic Algorithm (AGA) that dynamically optimizes pulses during circuit execution [6], guided by live feedback on performance [15]. Evolutionary algorithms are well-suited to navigate complex, noisy optimization landscapes;

here we use a genetic algorithm to evolve a population of pulse configurations, selecting those that maximize the output state fidelity. Furthermore, employing AGAs for quantum error mitigation illustrates how the structured use of automated learning techniques enhances the separation of concerns across different abstraction layers in the quantum stack [16]. Specifically, improving fidelity is a distinct challenge from executing a quantum algorithm and, ideally, should remain transparent to quantum programmers.

Our proposal introduces no additional gates or qubits – it works within the original circuit by tweaking how each gate is physically realized. We validate our proposed method on multiple algorithms, showing that it substantially boosts final-state fidelity without altering the algorithm's logic (thus preserving the algorithm's design and intent)

### A. Contributions

This paper introduces a novel adaptive pulse-level quantum error mitigation algorithm comprising three main advances:

1) **Adaptive Pulse Parameter Optimization:** A new algorithm dynamically adjusts pulse parameters in response to real-time noise measurements, thereby enhancing its responsiveness to fluctuating noise conditions within quantum systems.
2) **Enhanced Fidelity Without Circuit Alteration:** The algorithm achieves substantial fidelity improvements in quantum algorithms without requiring modifications to the original circuits, thereby preserving their design and intent.
3) **Empirical Validation on Benchmark Algorithms:** The efficacy of the proposed method is empirically demonstrated through enhanced performance metrics in five benchmark quantum algorithms, specifically Bernstein-Vazirani, Deutsch-Jozsa, Grover's search, Quantum Fourier Transform (QFT), and its inverse (IQFT)

Collectively, our contributions underscore the potential of evolutionary strategies to refine pulse-level control and achieve error mitigation. These also suggest an effective methodology to enhance the reliability of quantum computations on Noisy Intermediate-Scale Quantum (NISQ) devices. By leveraging adaptive optimization techniques, our methodology effectively navigates the noise of current quantum systems, thereby enabling more robust and accurate computational results.

### B. Organization of the Paper

The remainder of this paper is organized as follows: **Section II** provides the theoretical and technical background essential to our study, including noise modeling in quantum computing, the distinction between error mitigation and error correction, and an overview of the QuTiP framework used for our simulations. In **Section III**, we describe the design and implementation of our adaptive genetic algorithm for pulse-level error mitigation. This section covers the pulse representation, the genetic algorithm structure and its key features (such as adaptive feedback, diversity control, and parallelization), as well as details on how the algorithm is integrated with quantum circuit simulations. **Section IV** outlines the experimental setup and simulation parameters, and introduces the five benchmark quantum algorithms used in this study: Grover's Search, Deutsch–Jozsa, Bernstein–Vazirani, Quantum Fourier Transform (QFT), and Inverse QFT (IQFT). **Section V** presents the experimental results, including fidelity evolution and comparative analyses between optimized and baseline pulses across the different algorithms and noise regimes. **Section VI** discusses the implications of our findings, assessing the strengths and limitations of our technique while proposing directions for future research. Finally, **Section VII** concludes the paper by summarizing our key contributions and highlighting the potential impact of our work on advancing quantum error mitigation strategies.

## II. Background and Preliminaries

We model quantum errors using a combination of continuous and discrete noise processes typical in superconducting qubit systems [17]. We provide the necessary theoretical and technical background behind our method. We begin with an overview of noise modeling in quantum computing, highlighting various types of errors and their mathematical representations. Finally, we introduce QuTiP (Quantum Toolbox in Python), the open-source framework employed for simulating open quantum systems and implementing pulse-level strategies.

### A. Noise in NISQ Systems

Noisy Intermediate-Scale Quantum (NISQ) devices are inherently susceptible to decoherence and operational errors that degrade qubit fidelity and gate performance [1, 2]. Such errors arise from unavoidable system-environment interactions and hardware imperfections.

*1) Lindblad and Kraus Representations:* A standard framework for modeling open-system dynamics is the Lindblad master equation:

$$\frac{d\rho}{dt} = -\frac{i}{\hbar}[H,\rho] + \sum_k \Big( L_k\,\rho\,L_k^\dagger - \frac{1}{2}\{L_k^\dagger L_k, \rho\}\Big), \quad (1)$$

where $H$ is the system Hamiltonian and $L_k$ are the collapse operators. Equivalently, one may use the Kraus representation:

$$\mathcal{E}(\rho) = \sum_\ell E_\ell \rho E_\ell^\dagger, \quad \sum_\ell E_\ell^\dagger E_\ell = I.$$

*2) Standard Noise Channels:* Key noise models include:

- **Amplitude Damping:** Models energy relaxation (characterized by $T_1$).
- **Dephasing:** Affects relative phase coherence (characterized by $T_2$).
- **Pauli Errors:** Bit-flip, phase-flip, and bit-phase-flip errors occurring with probability $p$.
- **Depolarizing Noise:** Replaces the state with the maximally mixed state with probability $p$.

### B. Error Mitigation vs. Error Correction

Quantum Error Correction (QEC) encodes logical qubits into multiple physical qubits and uses syndrome measurements to detect and correct errors [7]. However, QEC's overhead makes it impractical for NISQ devices. Instead, Quantum Error Mitigation (QEM) reduces noise effects via techniques such as zero-noise extrapolation and probabilistic error cancellation [18, 8]. Our method complements these strategies by optimizing the pulse-level control fields without altering the circuit structure.

### C. Pulse-Level Modeling and Simulation Tools

Incorporating noise directly into the pulse evolution of gates provides a realistic representation of device behavior [19]. The control Hamiltonian is given by:

$$H(t) = H_0 + \sum_i u_i(t)\, H_i,$$

where $H_0$ is the drift Hamiltonian and $u_i(t)$ are the control amplitudes.

*1) QuTiP and QuTiP-QIP:* We use the *Quantum Toolbox in Python* (QuTiP) [20] for simulating open quantum systems. QuTiP offers a unified `Qobj` class for representing states and operators, and solvers such as `mesolve` for integrating Eq. (1). The `qutip-qip` module extends these capabilities to pulse-level simulations, mapping quantum circuits to pulse sequences with parameters such as evolution time (`evo_time`) and number of time slices (`num_tslots`). This framework forms the basis for our adaptive pulse-level error mitigation strategy.

## III. Methodology

Building on our previous work [6], we propose an adaptive pulse-level error mitigation algorithm that optimizes control pulses in real time without modifying the circuit logic. Below we describe the key components of our research.

### A. Pulse Representation and Control

A pulse is represented as a time series of complex-valued amplitudes, $[d_0, \ldots, d_{n-1}]$, where each sample $d_j$ is applied during a cycle time $dt$ (defined by the waveform generator's sampling rate). The ideal output at time $f_j dt$ is given by:

$$D_j = \mathrm{Re}\Big[e^{i(2\pi f_j dt + \phi_j)}\Big], \tag{2}$$

at time $f_j dt$, where $f_j$ is a modulation frequency and $\phi_j$ is a phase. Pulse samples describe only the envelope of the signal produced, which is then mixed in hardware with a carrier signal defined by its frequency and phase [21]

### B. Adaptive Genetic Algorithm for Pulse Optimization

Our method employs an adaptive Genetic Algorithm (GA) ([6]), to fine-tune pulse parameters and mitigate noise-induced errors. In this framework, a population of candidate pulse configurations is evolved over successive generations, with each candidate evaluated based on its ability to steer the system toward the ideal target state.

A key metric for benchmarking quantum operations is fidelity ([1]), which quantifies the closeness between two quantum states. Given two density matrices $\rho$ and $\sigma$, the fidelity $F(\rho, \sigma)$ is defined as

$$F(\rho, \sigma) \equiv \left(\mathrm{Tr}\sqrt{\rho^{1/2}\sigma\rho^{1/2}}\right)^2.$$

In our method, the fitness of each candidate is determined by the fidelity between the simulated final state and the desired target state:

$$F(\rho_{\text{final}}, \rho_{\text{target}}) = \left(\mathrm{Tr}\sqrt{\sqrt{\rho_{\text{final}}}\,\rho_{\text{target}}\,\sqrt{\rho_{\text{final}}}}\right)^2. \tag{3}$$

This fidelity measure provides a quantitative assessment of how effectively the optimized pulses improve circuit performance under noisy conditions.

Key features include:

- **Adaptive Feedback:** Mutation and crossover probabilities ($p_{\text{mut}}$ and $p_{\text{cross}}$) are dynamically adjusted based on the change in average fitness ($\Delta\overline{F}$) over a fixed interval:

$$p_{\text{mut}}^{(g+1)} = \begin{cases} p_{\text{mut}}^{(g)} + \Delta p, & \text{if } \Delta\overline{F} < \delta, \\ p_{\text{mut}}^{(g)} - \Delta p, & \text{otherwise,} \end{cases}$$

  and similarly for $p_{\text{cross}}$ [15, 6].
- **Diversity Control:** Genetic diversity is monitored using the average Mahalanobis distance between individuals. If diversity $D$ falls below a threshold $\theta$, additional mutations or individual replacements are applied [22, 6].
- **Elitism and Early Stopping:** The best-performing individuals are preserved in each generation, and the optimization process stops when no significant improvement is observed over $R$ consecutive generations [6].
- **Parallelization:** Fitness evaluations (simulations of noisy quantum circuits) are parallelized using the SCOOP framework [23].

Algorithm 1 summarizes the overall GA workflow.

**Algorithm 1:** Adaptive Pulse-Level Error Mitigation Algorithm (Overview)

**Input :** Population size $N$, number of generations $G$, initial mutation probability $p_{\text{mut}}$, initial crossover probability $p_{\text{cross}}$, feedback threshold $\delta$, feedback interval $I$, early stopping rounds $R$, diversity threshold $\theta$, diversity action (e.g., 'mutate' or 'replace')
**Output:** Optimized pulse parameters maximizing fidelity
Initialize population $P_0$;
**for** $g \leftarrow 1$ **to** $G$ **do**
  Select parents (Tournament Selection);
  Apply Crossover and Mutation;
  Evaluate Fitness (using Eq. (3));
  Replace Population and apply Elitism;
  Enforce Diversity Control and adjust probabilities;
  Check Early Stopping Criteria;
**end**
**return** *Best individual (i.e., highest fidelity)*;

### C. Workflow and Integration with Quantum Circuits

The optimized pulse parameters are applied to pulse-level representations of quantum circuits using the `qutip-qip` framework [18]. Specifically, parameters such as evolution time (`evo_time`) and the number of time slices (`num_tslots`) are refined to maximize fidelity under realistic noise conditions. Figure 1 illustrates the overall workflow, which begins with a quantum circuit that is compiled into pulse sequences, followed by noise integration, GA-based pulse optimization, and final simulation with the optimized pulses.

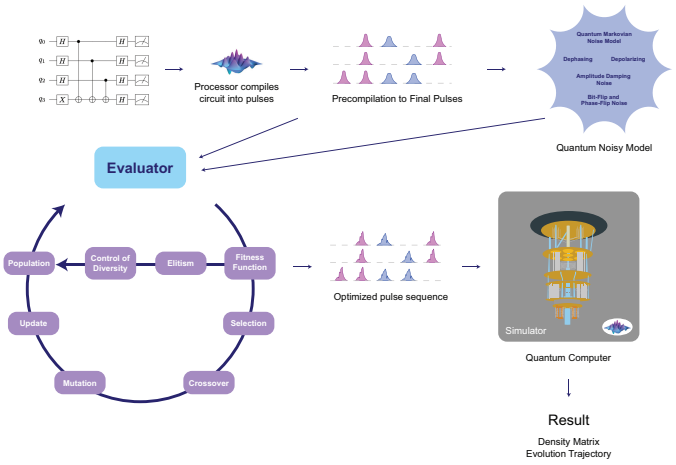


Fig. 1: Workflow of the Adaptive Pulse-Level Error Mitigation Algorithm.

### D. Implementation Details

Our implementation is developed in Python, using the DEAP library for the GA [24] and QuTiP for quantum simulation [20]. The SCOOP framework manages parallelization of fitness evaluations. For further technical details and the full pseudocode, please refer to the Appendix of our full paper [6].

## IV. Experimental Setup

All experiments were conducted on a MacBook Pro with Apple's M3 chip, which features an 8-core CPU (4 performance + 4 efficiency), a 10-core GPU, a 16-core Neural Engine, 100 GB/s memory bandwidth, and 24 GB of RAM. Table I summarizes the hardware specifications, demonstrating that our system can handle pulse-level quantum simulations and parallelized fitness evaluations using our GA framework.

TABLE I: MacBook Pro M3 Hardware Specifications

| Component | Specification |
|---|---|
| System on Chip (SoC) | Apple M3 |
| CPU Cores | 8 (4 performance + 4 efficiency) |
| GPU Cores | 10-core GPU |
| Neural Engine | 16-core |
| Memory Bandwidth | 100 GB/s |
| RAM | 24 GB |

### A. Benchmark Quantum Algorithms

To assess the effectiveness of our strategy, we evaluated performance on several well-established quantum algorithms. These benchmarks were chosen due to diversity in circuit structure, the frequency of appearance in literature and applications, and sensitivity to noise, making them suitable for testing improvements in fidelity.

*1) Bernstein-Vazirani:* The Bernstein-Vazirani algorithm efficiently determines a hidden bit-string [25]. Its shallow circuit and deterministic output make it an ideal benchmark for examining the impact of pulse-level optimization on error suppression.

*2) Deutsch-Jozsa:* The Deutsch-Jozsa algorithm distinguishes between constant and balanced Boolean functions in a single evaluation [26]. We implement this algorithm on a 4-qubit circuit (three input qubits and one ancilla), where an ideal measurement outcome is deterministic. This algorithm is particularly sensitive to interference effects, providing a rigorous test for noise mitigation.

*3) Grover's Search:* Grover's algorithm offers a quadratic speedup for searching unstructured databases [27]. In our 4-qubit implementation, the algorithm amplifies the probability of a marked state. Due to its iterative structure, it is especially prone to error accumulation, making it a robust benchmark for our GA.

*4) Quantum Fourier Transform (QFT) and Inverse QFT (IQFT):* The QFT and its inverse are central to phase estimation and other transform operations [1, 28]. Their sensitivity to phase errors makes them effective in assessing the benefits of pulse-level error mitigation.

For all algorithms, performance is quantified using the fidelity $F(\rho, \sigma)$ between the noisy output state and the ideal target state:

$$F(\rho, \sigma) \equiv \left( \mathrm{Tr} \sqrt{\rho^{1/2} \sigma \rho^{1/2}} \right)^2,$$

with higher values indicating that the output state more closely approximates the desired state [1].

### B. Simulation Parameters and Configurations

Our simulations employ a combined noise regime – including amplitude damping, dephasing, and independent Pauli errors- with parameters chosen to emulate typical NISQ conditions [29, 30, 31, 32]. For all experiments, the genetic algorithm (GA) is configured with a population size of 100 individuals and is run for 200 generations. Table II summarizes the specific parameters for each experimental scenario, which include variations in relaxation ($T_1$) and dephasing ($T_2$) times, as well as bit-flip and phase-flip error probabilities. The benchmarks span several quantum circuits implemented on 4-qubit systems, namely Grover's Search, Deutsch-Jozsa, Bernstein-Vazirani, Quantum Fourier Transform (QFT), and its inverse (IQFT).

This setup, along with our adaptive GA framework and realistic noise modeling, enables a comprehensive evaluation of our error mitigation technique across various algorithms and operating conditions.

| Scenario | Circuit | Num Qubits | $T_1$ | $T_2$ | Bit-Flip Prob | Phase-Flip Prob |
|---|---|---|---|---|---|---|
| 1 | Grover | 4 | 30 | 15 | 0.01 | 0.01 |
| 2 | Grover | 4 | 50 | 30 | 0.02 | 0.02 |
| 3 | Grover | 4 | 100 | 60 | 0.05 | 0.05 |
| 4 | Deutsch-Jozsa | 4 | 30 | 15 | 0.01 | 0.01 |
| 5 | Deutsch-Jozsa | 4 | 50 | 30 | 0.02 | 0.02 |
| 6 | Deutsch-Jozsa | 4 | 100 | 60 | 0.05 | 0.05 |
| 7 | Bernstein-Vazirani | 4 | 30 | 15 | 0.01 | 0.01 |
| 8 | Bernstein-Vazirani | 4 | 50 | 30 | 0.02 | 0.02 |
| 9 | Bernstein-Vazirani | 4 | 100 | 60 | 0.05 | 0.05 |
| 10 | QFT | 4 | 30 | 15 | 0.01 | 0.01 |
| 11 | QFT | 4 | 50 | 30 | 0.02 | 0.02 |
| 12 | QFT | 4 | 100 | 60 | 0.05 | 0.05 |
| 13 | IQFT | 4 | 30 | 15 | 0.01 | 0.01 |
| 14 | IQFT | 4 | 50 | 30 | 0.02 | 0.02 |
| 15 | IQFT | 4 | 100 | 60 | 0.05 | 0.05 |

TABLE II: Sampling of Experimental Scenarios and Simulation Parameters. Each scenario corresponds to a 4-qubit circuit with specific decoherence times ($T_1$ and $T_2$) and error probabilities for bit-flip and phase-flip channels. All simulations were performed using a genetic algorithm configured with 200 generations and a population of 100 individuals.

## V. Results

We evaluated our adaptive pulse-level error mitigation GA over 15 experimental scenarios (see Table II) on five benchmark quantum algorithms implemented on 4-qubit [1] systems. The algorithms under study were:

- **Grover's Search** (Scenarios 1–3),
- **Deutsch–Jozsa** (Scenarios 4–6),
- **Bernstein–Vazirani** (Scenarios 7–9),
- **Quantum Fourier Transform (QFT)** (Scenarios 10–12), and
- **Inverse QFT (IQFT)** (Scenarios 13–15).

For each algorithm, experiments were performed under three noise regimes (see Table III):

- **Low Noise:** $T_1 = 30$, $T_2 = 15$, with bit-flip and phase-flip probabilities of 0.01.
- **Medium Noise:** $T_1 = 50$, $T_2 = 30$, with bit-flip and phase-flip probabilities of 0.02.
- **High Noise:** $T_1 = 100$, $T_2 = 60$, with bit-flip and phase-flip probabilities of 0.05.

In all scenarios, the genetic algorithm (GA) is configured with a population size of 100 and is run for 200 generations. Performance is quantified using the fidelity $F(\rho, \sigma)$ between the noisy final state and the ideal target state [1].

We computed the evolution of average fidelity over multiple generations for all five algorithms under the three noise conditions (Figure 2). Each subplot corresponds to one algorithm, with curves representing Low, Medium, and High noise levels. As illustrated, the adaptive GA consistently enhances fidelity over successive generations. Although the magnitude of improvement is generally greater under low noise conditions, significant gains are observed even in high noise regimes when compared to the unoptimized baseline.

The final outcomes for each algorithm are summarized in Tables IV–VIII. For each noise level, the following metrics are reported:

- **Original Fidelity:** The fidelity before any optimization.
- **Avg. Optimized Fidelity:** The average fidelity achieved by the population after optimization.
- **Max Optimized Fidelity:** The highest fidelity achieved during the run.
- **Gen Max Fidelity:** The generation at which the maximum fidelity was reached.
- **Total Execution Time:** The overall wall-clock time required for the experiment.

Mean and standard deviation (SD) values across the noise regimes are provided across the optimization process.

TABLE III: Definition of Noise Regimes

| Noise Level | $T_1$ | $T_2$ | Bit-Flip Prob | Phase-Flip Prob |
|---|---|---|---|---|
| Low | 30 | 15 | 0.01 | 0.01 |
| Medium | 50 | 30 | 0.02 | 0.02 |
| High | 100 | 60 | 0.05 | 0.05 |

The final outcomes for each algorithm are detailed in Tables IV–VIII. These tables list, for each noise regime, the original fidelity (obtained without optimization), the average optimized fidelity, the maximum optimized fidelity achieved, the generation at which the maximum fidelity was observed, and the total execution time of the experiment. Mean and standard deviation (SD) values across the noise levels are provided where applicable.

Together, Tables IV–VIII provide a comprehensive overview of our experimental results. For each algorithm, these tables report the fidelity before optimization, the average fidelity after running the GA, the best (maximum) fidelity achieved during the run, the generation at which this maximum was reached, and the total execution time. The mean and standard deviation values further quantify the variability of the outcomes across different noise conditions.

Overall, the results demonstrate that our adaptive pulse-level error mitigation methodology consistently improves the fidelity of quantum circuits across a range of algorithms and noise regimes. Although the absolute fidelity values and convergence speeds differ with the noise level, our method

[1]Detailed results: `IEEE Quantum Week Conference Data for Review/instructions.md`. See *Code and Data Availability* below for data access and replication.

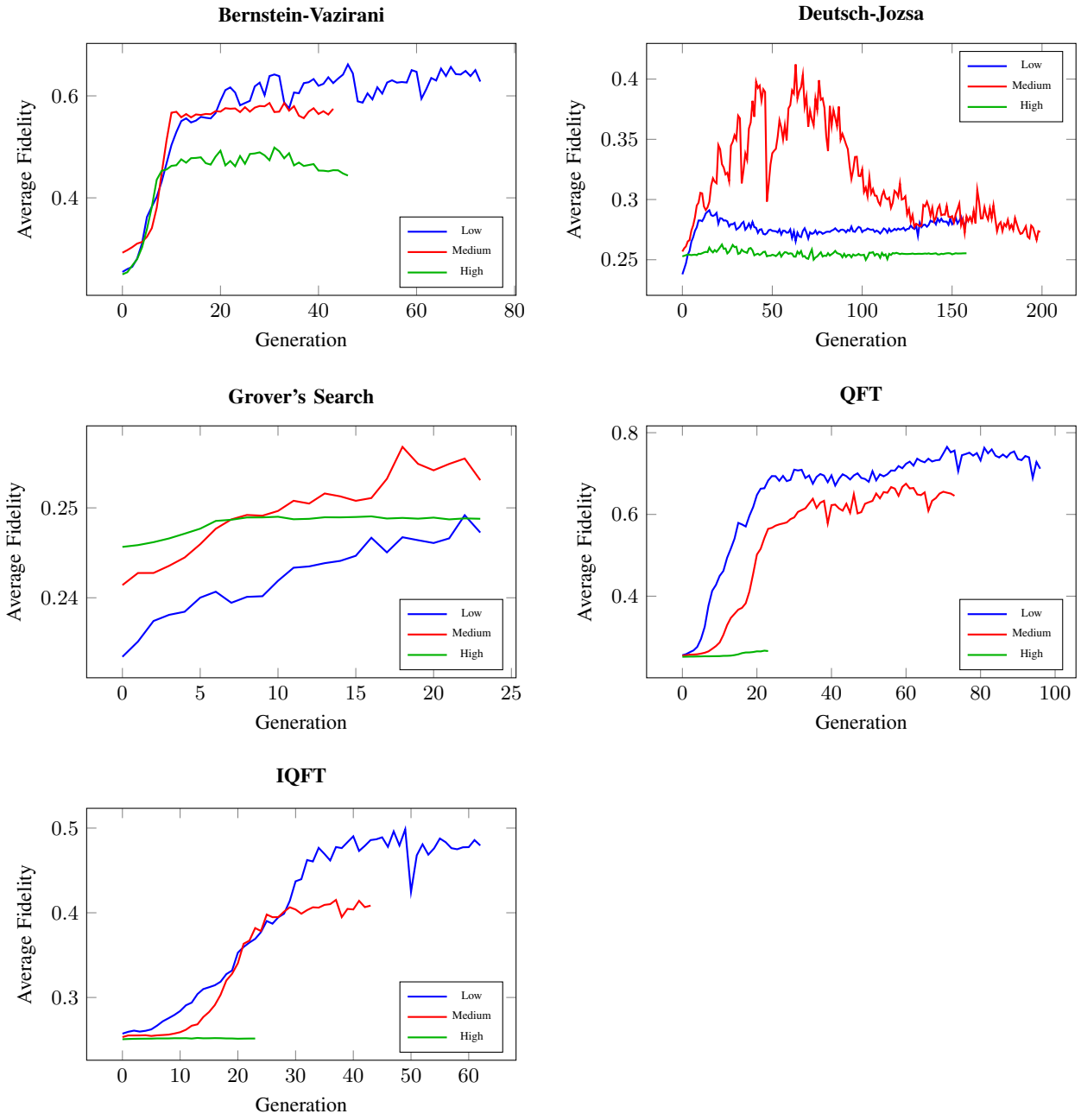


Fig. 2: Evolution of average fidelity over generations for five benchmark quantum algorithms under three noise regimes (see Table III for details). Each subplot corresponds to one algorithm, with curves representing the Low, Medium, and High noise regimes.

TABLE IV: Final Outcomes for Bernstein–Vazirani (4 Qubits)

| Noise Level | Qubits | Original Fidelity | Avg. Optimized Fidelity | Max Optimized Fidelity | Gen Max Fidelity | Total Execution Time |
|---|---|---|---|---|---|---|
| Low | 4 | 0.2650 | 0.6617 | 0.7229 | 46 | 1203.84 s |
| Medium | 4 | 0.3007 | 0.5859 | 0.6326 | 30 | 1002.85 s |
| High | 4 | 0.2491 | 0.4986 | 0.5676 | 31 | 861.35 s |
| *Mean* | | 0.2716 | 0.5821 | 0.6410 | – | 1022.68 s |
| *SD* | | 0.0264 | 0.0816 | 0.0780 | – | 172.10 s |

yields significant enhancements over unoptimized baseline implementations. These findings underscore the potential of our technique to extend the practical capabilities of Noisy Intermediate-Scale Quantum (NISQ) devices.

TABLE V: Final Outcomes for Deutsch–Jozsa (4 Qubits)

| Noise Level | Qubits | Original Fidelity | Avg. Optimized Fidelity | Max Optimized Fidelity | Gen Max Fidelity | Total Execution Time |
|---|---|---|---|---|---|---|
| Low | 4 | 0.2425 | 0.2911 | 0.4197 | 15 | 17994.25 s |
| Medium | 4 | 0.2549 | 0.4122 | 0.5467 | 63 | 5646.18 s |
| High | 4 | 0.2519 | 0.2626 | 0.4058 | 22 | 8685.75 s |
| *Mean* | | 0.2498 | 0.3220 | 0.4574 | – | 10775.39 s |
| *SD* | | 0.0065 | 0.0794 | 0.0776 | – | 6433.79 s |

TABLE VI: Final Outcomes for Grover's Search (4 Qubits)

| Noise Level | Qubits | Original Fidelity | Avg. Optimized Fidelity | Max Optimized Fidelity | Gen Max Fidelity | Total Execution Time |
|---|---|---|---|---|---|---|
| Low | 4 | 0.2423 | 0.2491 | 0.2741 | 22 | 4554.00 s |
| Medium | 4 | 0.2381 | 0.2568 | 0.3048 | 18 | 2219.36 s |
| High | 4 | 0.2460 | 0.2491 | 0.2506 | 16 | 4447.70 s |
| *Mean* | | 0.2421 | 0.2517 | 0.2765 | – | 3740.35 s |
| *SD* | | 0.0040 | 0.0044 | 0.0272 | – | 1318.29 s |

TABLE VII: Final Outcomes for IQFT (4 Qubits)

| Noise Level | Qubits | Original Fidelity | Avg. Optimized Fidelity | Max Optimized Fidelity | Gen Max Fidelity | Total Execution Time |
|---|---|---|---|---|---|---|
| Low | 4 | 0.2543 | 0.4985 | 0.5313 | 49 | 8012.64 s |
| Medium | 4 | 0.2500 | 0.4152 | 0.4410 | 37 | 8661.14 s |
| High | 4 | 0.2488 | 0.2522 | 0.2546 | 13 | 3713.86 s |
| *Mean* | | 0.2510 | 0.3886 | 0.4090 | – | 6795.88 s |
| *SD* | | 0.0029 | 0.1253 | 0.1411 | – | 2688.73 s |

TABLE VIII: Final Outcomes for QFT (4 Qubits)

| Noise Level | Qubits | Original Fidelity | Avg. Optimized Fidelity | Max Optimized Fidelity | Gen Max Fidelity | Total Execution Time |
|---|---|---|---|---|---|---|
| Low | 4 | 0.2562 | 0.7653 | 0.7813 | 71 | 14234.71 s |
| Medium | 4 | 0.2544 | 0.6752 | 0.7005 | 60 | 7460.93 s |
| High | 4 | 0.2522 | 0.2667 | 0.2705 | 22 | 2636.15 s |
| *Mean* | | 0.2543 | 0.5691 | 0.5841 | – | 8110.60 s |
| *SD* | | 0.0020 | 0.2657 | 0.2746 | – | 5826.51 s |

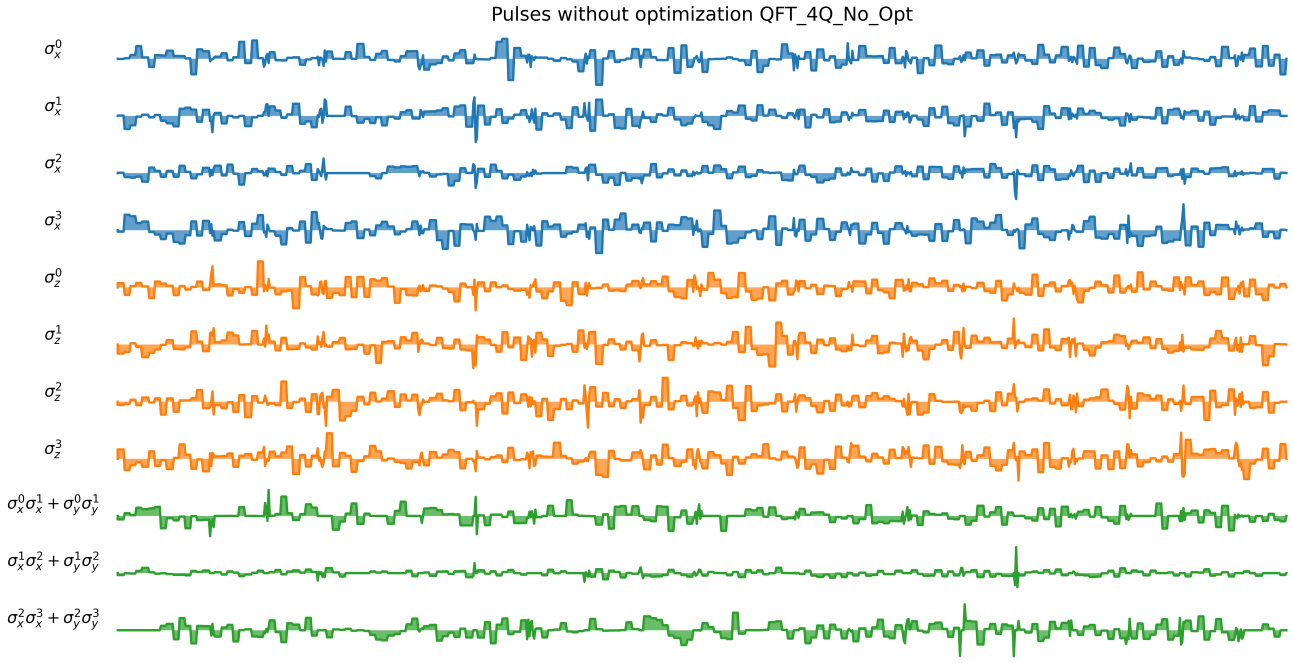


Fig. 3: Baseline pulse waveforms for the Quantum Fourier Transform (QFT) obtained for a representative individual across all simulations.

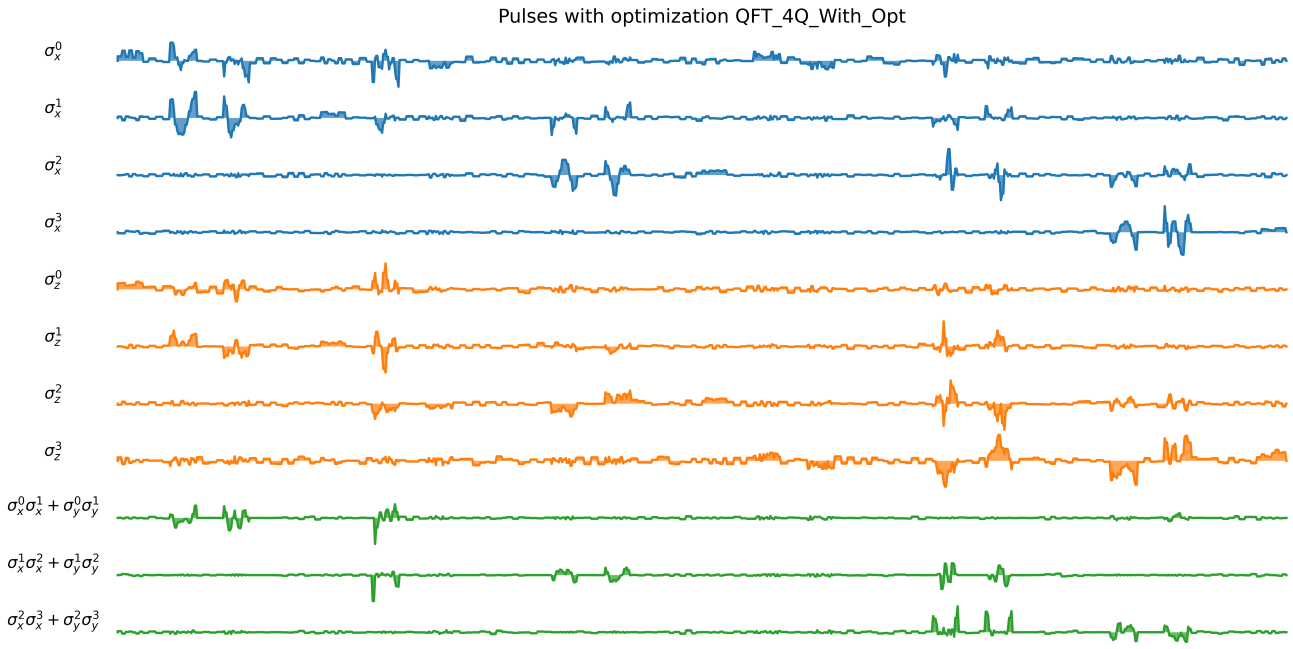


Fig. 4: Optimized pulse waveforms for the QFT obtained for the same representative individual after applying our adaptive genetic algorithm.

As shown in Figures 3 and 4, there is a clear difference between the baseline and optimized pulse waveforms for the QFT algorithm. The optimized pulses exhibit significant modifications in both amplitude and phase profiles, which contribute to a marked improvement in fidelity under noisy conditions. This visual evidence corroborates our quantitative results and highlights the efficacy of our adaptive pulse-level error mitigation technique in enhancing the performance of QFT on NISQ devices.

## VI. Discussion

In this section, we contextualize and interpret the results presented in Tables IV–VIII and Figs. 2–4. By examining each benchmark algorithm in turn, we highlight key trends and attempt to explain how (and possibly why) our pulse-level optimization via an adaptive genetic algorithm (AGA) consistently improves circuit fidelity in the Noisy Intermediate-Scale Quantum (NISQ) setting.

### A. Bernstein–Vazirani

For Bernstein–Vazirani (Table IV), the baseline fidelities range from roughly $0.25$ to $0.30$ across the three noise scenarios. Upon optimization, the average fidelity climbs into the $0.50$–$0.66$ range, reaching a maximum of $0.72$ under low noise. In Fig. 2 (top-left), both the low- and medium-noise curves ascend steadily for the first 40–50 generations, eventually plateauing near $0.60$–$0.70$. Because Bernstein–Vazirani is relatively shallow, the AGA can effectively counteract decoherence and bit/phase-flip errors by re-tuning pulse amplitudes and phases. However, when noise is high, the circuit still benefits from a notable jump in fidelity (up to about $0.57$), albeit at a slower and more constrained pace.

### B. Deutsch–Jozsa

The Deutsch–Jozsa algorithm (Table V) starts with baseline fidelities close to $0.25$, where it is challenging to distinguish balanced from constant functions under strong noise. As the genetic algorithm optimizes pulses, medium noise (red curve in Fig. 2 top-right) exhibits the most significant gains, briefly exceeding $0.40$–$0.45$. Interestingly, this surpasses the fidelity achieved under low noise in some runs. One likely explanation is that the initial, unoptimized pulse schedule for medium noise left more "room" for improvement than its low-noise counterpart. By generation 50–60, the average fidelity in medium noise conditions its maximum of around $0.54$–$0.55$, underscoring that pulse-level adjustments can be especially effective if the underlying gate implementations are initially suboptimal. Under high noise, average fidelity remains comparatively low (about $0.26$), underscoring the limited power of local pulse corrections when decoherence is dominant.

### C. Grover's Search

Grover's Search (Table VI and Fig. 2 middle-left) displays more modest overall improvements. While there is a small but discernible fidelity boost –rising from about $0.24$–$0.25$ to $0.27$–$0.30$ in the best cases- this is less pronounced than in Bernstein–Vazirani or QFT. Grover's iterative nature amplifies small phase or amplitude errors over multiple rounds of oracle and diffusion operations, making them harder to correct

entirely at the pulse level. The results suggest that algorithms with repeated gate layers may require additional error mitigation beyond local pulse parameter refinements. Nevertheless, the fact that fidelity increases in each noise scenario confirms that AGA-driven pulse adaptation still confers a measurable advantage over unoptimized schedules.

### D. Quantum Fourier Transform (QFT)

In contrast, QFT (Table VIII) achieves some of the largest fidelity gains of all benchmarks. Under low noise, fidelity escalates from around $0.25$ to $0.78$ at its peak –more than tripling the baseline. As seen in Fig. 2 (middle-right), the average fidelity rises steeply within the first 40 generations. Medium noise also performs well, achieving maximum fidelities of about $0.70$. This strongly indicates that phase-sensitive algorithms, which rely on controlled-phase gates, reap substantial rewards from pulse-level refinement. Even minor corrections to pulse phases and amplitudes appear to mitigate errors that would otherwise compromise the final interference pattern. However, under high noise, the maximum fidelity hovers around $0.27$, reflecting the saturating impact of amplitude damping and random phase flips that cannot be fully compensated by local pulse adjustments.

### E. Inverse QFT (IQFT)

IQFT shares several features with its forward counterpart but tops out at moderately lower fidelities (Table VII and Fig. 2 bottom). In the low-noise regime, AGA increases fidelity from around $0.25$ to about $0.53$, with a similar pattern in medium noise. Relative to QFT, this difference may stem from measurement basis interactions or subtle differences in how the circuit's controlled gates arrange phase relationships during the final uncomputation. Under high noise, fidelity remains close to baseline, again suggesting that once noise dominates, the marginal gains from pulse-level optimization diminish.

### F. Cross-Cutting Observations

*Saturation Effects.:* In all algorithms, high noise limits final fidelity: once $T_1/T_2$ effects and Pauli errors dominate, pulse-level control alone cannot eliminate decoherence. This plateauing behavior is evident in the green curves of Fig. 2, which rarely exceed the baseline by more than a few percentage points.

*Circuit Depth and Gate Composition.:* Algorithms with fewer layers (e.g., Bernstein–Vazirani) and those centered on phase gates (QFT, IQFT) generally obtain greater fidelity gains. In iterative algorithms such as Grover's, even small residual errors multiply, limiting overall improvements. This observation is consistent with fundamental limits on error-mitigation performance reported by Takagi *et al.* [8].

*Sensitivity to Pulse Initial Conditions.:* For Deutsch–Jozsa in medium noise, the larger gap between baseline and optimized fidelity suggests that an initially less-refined pulse schedule can yield high relative gains once the AGA explores phase/amplitude corrections. Conversely, a near-optimal baseline in low noise can reduce the total improvement, despite being closer to the hardware's fundamental limits.

### G. Implications for NISQ Computing

These findings affirm that refined pulse-level control offers a practical means to curb errors in near-term quantum devices without altering circuit or adding qubits. Our methodology is particularly well-suited to moderately noisy conditions, where adaptive adjustments can substantially restore phase coherence. While not a panacea –especially at extreme noise levels- this strategy can be combined with other techniques (e.g., dynamical decoupling, zero-noise extrapolation) [33, 9] to further extend algorithmic performance. Moreover, as Figs. 3 and 4 illustrate, the optimized pulses often deviate substantially from baseline waveforms, underscoring the flexibility and need for data-driven, dynamic tuning in quantum hardware implementations.

Evidence provided here suggests that employing an adaptive genetic algorithm to shape control pulses on the fly is a powerful tool for quantum error mitigation in NISQ-era machines. By addressing errors at their physical origin, this method enables significant fidelity gains, thereby broadening the practical scope of existing quantum algorithms before large-scale error-corrected quantum hardware becomes available.

## VII. Conclusion

We benchmarked an adaptive genetic algorithm specifically designed for pulse-level quantum error mitigation. Our methodology underwent evaluation on a suite of standard quantum algorithms –Bernstein–Vazirani, Deutsch–Jozsa, Grover's Search, Quantum Fourier Transform (QFT), and its inverse (IQFT)- demonstrating its potential to significantly enhance the resilience and fidelity of quantum circuits operating in noisy environments.

Our findings underscore the significance of assisted pulse engineering in bridging the performance and fidelity gaps between theoretical circuit abstractions and practical hardware implementations. By optimizing control pulses in real time, our method effectively compensates for various noise sources without altering the logical structure of the original circuit. This not only simplifies the error mitigation process but also preserves the integrity of quantum algorithms, making them more robust against the inherent imperfections of Noisy Intermediate-Scale Quantum (NISQ) devices while preventing an increase in circuit complexity for human interpretation purposes. Moreover, due to compute intensity, this optimization technique requires HPC infrastructure at scale.

The improvements in fidelity we observed across diverse algorithms highlight the versatility of our genetic algorithm framework, even under varying noise conditions. Although these were more pronounced in some circuits than in others, our results demonstrate that tailored pulse-level adjustments can substantially mitigate errors, especially for phase-sensitive operations such as those in the QFT and IQFT.

**Limitations.** Although our results demonstrate substantial improvements in circuit fidelity through adaptive pulse-level error mitigation, several limitations remain. First, our experiments were limited to 4-qubit circuits; the scalability of this method to larger, more complex setups remains to be validated.

Second, the noise models used in our simulations capture key error sources but may not fully represent the complex error dynamics encountered in real quantum hardware, especially under severe decoherence. Finally, the performance of real-time pulse optimization is influenced by the starting pulse settings, and it might need additional fine-tuning to adapt to different hardware architectures or to cope with rapidly changing noise environments. Future studies should explore these aspects to improve the adaptability and overall reliability of pulse-level error mitigation methods in practical quantum computing applications.

**Future Work.** We identify three primary avenues of interest:

1) **Method refinement and integration.** As computational power increases, our algorithm can be executed with larger populations and over more generations, enabling a more thorough evaluation across diverse noise regimes. We also anticipate that improvements to the GA engine –such as incorporating adaptive mutation schedules- will accelerate convergence. Moreover, integrating this method into established libraries like QuTiP would streamline the development of pulse-level optimization workflows, encouraging wider adoption in the quantum computing community.
2) **QPU and algorithms testing.** Our next step involves testing the pulse-level optimization strategy on actual quantum hardware to assess its ability to perform adaptive error mitigation under real-world noise conditions. This will require pulse-level access to quantum processing units, fostering potential collaborations with hardware testbeds and quantum technology providers. Additionally, expanding our evaluation to include a broader array of quantum algorithms –particularly those that challenge the performance of NISQ devices- will help further validate the robustness and practical impact of our methods.
3) **Oracle-based heuristics discovery.** It seems reasonable to hypothesize that the structure of an oracle will impact its pulse-level optimization potential. Their rigorous exploration would help classify specific cases into circuits whose optimization requires more compute power and those where approximation still renders good results. We anticipate pulse-level compilation heuristics analogous to those in quantum circuit compilation [34].

## Acknowledgments

'W.A-C. wishes to thank CENFOTEC University for providing direct academic support and resources essential for the completion of this research. S.N-C. thanks the National Center for Supercomputing Applications and the Illinois Quantum Information Science and Technology Center for continued support, and CENFOTEC University for the opportunity to establish a fruitful research collaboration. Francini Corrales and Geisel Hernández provided insights across the design and execution of the project. Lastly, we acknowledge the QuTiP and qutip-qip teams for their outstanding tools and assistance, which were instrumental in the successful implementation of the quantum simulations reported in this work.

## Code and Data Availability

All code and scripts required to replicate experiments and results described above are available on our GitHub repository. We provide detailed instructions to facilitate the reproduction of our simulations and analyses on various computational platforms, including environment setup and example scripts.

## References

[1] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge: Cambridge University Press, 2010.

[2] J. Preskill, "Quantum computing in the nisq era and beyond," *Quantum*, vol. 2, p. 79, Aug. 2018.

[3] J. Roffe, "Quantum error correction: an introductory guide," *Contemporary Physics*, vol. 60, no. 3, pp. 226–245, Jul. 2019.

[4] G. Ithier, E. Collin, P. Joyez, P. J. Meeson, D. Vion, D. Esteve, F. Chiarello, A. Shnirman, Y. Makhlin, J. Schriefl, and G. Schön, "Decoherence in a superconducting quantum bit circuit," *Physical Review B*, vol. 72, no. 13, Oct. 2005. [Online]. Available: http://dx.doi.org/10.1103/PhysRevB.72.134519

[5] P. Murali, D. C. Mckay, M. Martonosi, and A. Javadi-Abhari, "Software mitigation of crosstalk on noisy intermediate-scale quantum computers," in *Proceedings of the Twenty-Fifth International Conference on Architectural Support for Programming Languages and Operating Systems*, ser. ASPLOS '20. ACM, Mar. 2020, p. 1001–1016. [Online]. Available: http://dx.doi.org/10.1145/3373376.3378477

[6] W. Aguilar-Calvo and S. Núñez-Corrales, "Adaptive genetic algorithms for pulse-level quantum error mitigation," 2025. [Online]. Available: https://arxiv.org/abs/2501.14007

[7] P. W. Shor, "Scheme for reducing decoherence in quantum computer memory," *Phys. Rev. A*, vol. 52, no. 4, pp. R2493–R2496, Oct. 1995.

[8] R. Takagi, S. Endo, S. Minagawa, and M. Gu, "Fundamental limits of quantum error mitigation," *npj Quantum Information*, vol. 8, no. 1, Sep. 2022. [Online]. Available: http://dx.doi.org/10.1038/s41534-022-00618-z

[9] T. Giurgica-Tiron, Y. Hindy, R. LaRose, A. Mari, and W. J. Zeng, "Digital zero noise extrapolation for quantum error mitigation," in *2020 IEEE International Conference on Quantum Computing and Engineering (QCE)*, 2020, pp. 306–316.

[10] E. van den Berg, Z. K. Minev, A. Kandala, and K. Temme, "Probabilistic error cancellation with sparse pauli–lindblad models on noisy quantum processors," *Nature Physics*, vol. 19, no. 8, p. 1116–1121, May 2023. [Online]. Available: http://dx.doi.org/10.1038/s41567-023-02042-2

[11] K. N. Smith *et al.*, "Programming physical quantum systems with pulse-level control," *Front. Phys.*, vol. 10, p. 2022, 2022.

[12] R. Porotti, V. Peano, and F. Marquardt, "Gradient-ascent pulse engineering with feedback," *PRX Quantum*, vol. 4, no. 3, p. 030305, Jul. 2023.

[13] T. Caneva, T. Calarco, and S. Montangero, "Chopped random-basis quantum optimization," *Phys. Rev. A*, vol. 84, no. 2, p. 022326, Aug. 2011.

[14] N. Shammah, A. Saha Roy, C. G. Almudever, S. Bourdeauducq, A. Butko, G. Cancelo, S. M. Clark, J. Heinsoo, L. Henriet, G. Huang, C. Jurczak, J. Kotilahti, A. Landra, R. LaRose, A. Mari, K. Nowrouzi, C. Ockeloen-Korppi, G. Prawiroatmodjo, I. Siddiqi, and W. J. Zeng, "Open hardware solutions in quantum technology," *APL Quantum*, vol. 1, no. 1, Feb. 2024. [Online]. Available: http://dx.doi.org/10.1063/5.0180987

[15] A. E. Eiben, Z. Michalewicz, M. Schoenauer, and J. E. Smith, "Parameter control in evolutionary algorithms," in *Parameter Setting in Evolutionary Algorithms*, ser. Studies in Computational Intelligence, F. G. Lobo, C. F. Lima, and Z. Michalewicz, Eds. Berlin, Heidelberg: Springer, 2007, vol. 54.

[16] O. Di Matteo, S. Núñez-Corrales, M. Stechły, S. P. Reinhardt, and T. Mattson, "An abstraction hierarchy toward productive quantum programming," *arXiv preprint arXiv:2405.13918*, 2024.

[17] F. Arute, K. Arya, R. Babbush *et al.*, "Quantum supremacy using a programmable superconducting processor," *Nature*, vol. 574, pp. 505–510, 2019.

[18] B. Li, S. Ahmed, S. Saraogi, N. Lambert, F. Nori, A. Pitchford, and N. Shammah, "Pulse-level noisy quantum circuits with QuTiP," *Quantum*, vol. 6, p. 630, Jan. 2022. [Online]. Available: https://doi.org/10.22331/q-2022-01-24-630

[19] G. Di Bartolomeo, M. Vischi, F. Cesa, R. Wixinger, M. Grossi, S. Donadi, and A. Bassi, "Noisy gates for simulating quantum computers," *Physical Review Research*, vol. 5, no. 4, Dec. 2023. [Online]. Available: http://dx.doi.org/10.1103/PhysRevResearch.5.043210

[20] J. R. Johansson, P. D. Nation, and F. Nori, "Qutip 2: A python framework for the dynamics of open quantum systems," *Computer Physics Communications*, vol. 184, no. 4, pp. 1234–1240, Apr. 2013.

[21] T. Alexander, N. Kanazawa, D. J. Egger, L. Capelluto, C. J. Wood, A. Javadi-Abhari, and D. C. McKay, "Qiskit pulse: programming quantum computers through the cloud with pulses," *Quantum Science and Technology*, vol. 5, no. 4, p. 044006, Aug. 2020.

[22] C. Jassadapakorn and P. Chongstitvatana, "Self-adaptation mechanism to control the diversity of the population in genetic algorithm," *International Journal of Computer Science and Information Technology*, vol. 3, no. 4, p. 111–127, Aug. 2011. [Online]. Available: http://dx.doi.org/10.5121/ijcsit.2011.3409

[23] Y. Hold-Geoffroy, O. Gagnon, and M. Parizeau, "Once you scoop, no need to fork," in *Proceedings of the 2014 Annual Conference on Extreme Science and Engineering Discovery Environment*, ser. XSEDE '14. New York, NY, USA: Association for Computing Machinery, 2014. [Online]. Available: https://doi.org/10.1145/2616498.2616565

[24] F.-A. Fortin, F.-M. D. Rainville, M.-A. Gardner, M. Parizeau, and C. Gagné, "Deap: Evolutionary algorithms made easy," *Journal of Machine Learning Research*, vol. 13, pp. 2171–2175, Jul. 2012.

[25] E. Bernstein and U. Vazirani, "Quantum complexity theory," in *Proceedings of the Twenty-Fifth Annual ACM Symposium on Theory of Computing*, ser. STOC '93. New York, NY, USA: Association for Computing Machinery, 1993, p. 11–20. [Online]. Available: https://doi.org/10.1145/167088.167097

[26] D. Collins, K. W. Kim, and W. C. Holton, "Deutsch-jozsa algorithm as a test of quantum computation," *Physical Review A*, vol. 58, no. 3, pp. R1633–R1636, Sep. 1998.

[27] L. K. Grover, "A fast quantum mechanical algorithm for database search," in *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing (STOC '96)*, Philadelphia, Pennsylvania, USA, 1996, pp. 212–219.

[28] D. Coppersmith, "An approximate fourier transform useful in quantum factoring," 2002.

[29] D. Greenbaum and Z. Dutton, "Modeling coherent errors in quantum error correction," *Quantum Science and Technology*, vol. 3, no. 1, p. 015007, Dec. 2017. [Online]. Available: http://dx.doi.org/10.1088/2058-9565/aa9a06

[30] Y. Ma, M. Hanks, E. Gneusheva, and M. S. Kim, "Reshaping quantum device noise via quantum error correction," 2024. [Online]. Available: https://arxiv.org/abs/2411.00751

[31] Y. Ma, M. Hanks, and M. S. Kim, "Non–pauli errors can be efficiently sampled in qudit surface codes," *Phys. Rev. Lett.*, vol. 131, p. 200602, 11 2023. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevLett.131.200602

[32] J. J. Wallman and J. Emerson, "Noise tailoring for scalable quantum computation via randomized compiling," *Phys. Rev. A*, vol. 94, p. 052325, 11 2016. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevA.94.052325

[33] A. Rahman, D. J. Egger, and C. Arenz, "Learning how to dynamically decouple," 2024. [Online]. Available: https://arxiv.org/abs/2405.08689

[34] K. Karuppasamy, V. Puram, S. Johnson, and J. P. Thomas, "A comprehensive review of quantum circuit optimization: Current trends and future directions," *Quantum Reports*, vol. 7, no. 1, p. 2, 2025.