

Make Some Noise! Measuring Noise Model Quality in Real-World Quantum Software

Stefan Raimund Maschek, Jürgen Schwitalla

science + computing AG, Eviden

Tübingen, Germany

{stefan-raimund.maschek, juergen.schwitalla}@eviden.com

Maja Franz

Technical University of

Applied Sciences Regensburg

Regensburg, Germany

maja.franz@othr.de

Wolfgang Mauerer

Technical University of

Applied Sciences Regensburg

Siemens AG, Technology

Regensburg/Munich, Germany

wolfgang.mauerer@othr.de

Abstract—Noise and imperfections are among the prevalent challenges in quantum software engineering for current NISQ systems. They will remain important in the post-NISQ area, as logical, error-corrected qubits will be based on software mechanisms. As real quantum hardware is still limited in size and accessibility, noise models for classical simulation—that in some cases can exceed dimensions of actual systems—play a critical role in obtaining insights into quantum algorithm performance, and the properties of mechanisms for error correction and mitigation.

We present, implement and validate a tunable noise model building on the Kraus channel formalism on a large scale quantum simulator system (*Qaptiva*). We use empirical noise measurements from IBM quantum (IBMQ) systems to calibrate the model and create a realistic simulation environment. Experimental evaluation of our approach with Greenberger-Horne-Zeilinger (GHZ) state preparation and QAOA applied to an industrial use-case validate our approach, and demonstrate accurate simulation of hardware behaviour at reasonable computational cost.

We devise and utilise a method that allows for determining the quality of noise models for larger problem instances than is possible with existing metrics in the literature. To identify potentials of future quantum software and algorithms, we extrapolate the noise model to future partially fault-tolerant systems, and give insights into the interplay between hardware-specific noise modelling and hardware-aware algorithm development.

Index Terms—Quantum Computing, Noise Model, Industrial Application

I. INTRODUCTION

Despite recent advancements in constructing error-corrected and fault-tolerant quantum computers [1], [2], noise and imperfections remain dominant challenges in current noisy intermediate-scale quantum (NISQ) systems [3]. Therefore, there persists a need to acknowledge noisy hardware and in particular noisy models of quantum hardware when engineering quantum software and algorithms. For the path toward achieving practical quantum advantage, it is essential to understand which types of noise impact specific computations and to what extent. A deep understanding of hardware limitations can guide the development and deployment of hardware-specific quantum algorithms, fostering a hardware-software co-design paradigm [4], [5]. A critical component in this approach is the ability to accurately simulate the behaviour of current quantum processing units (QPUs) using noise models.

Additional advantages of a noisy simulation over an execution on real QPUs include the ability to analyse quantum algorithms at any intermediate point in a circuit and the ability to directly obtain probabilities (and even amplitudes) instead of relative frequencies. Furthermore, a noise model also represents a fictitious parametrisable QPU that can be used as a test bed for future quantum algorithms.

While the theoretical modelling of various noise sources is well-established [6], simulating the exact behaviour of a specific physical QPU remains challenging. However, it is possible to develop increasingly accurate noise models that capture the nuances of real-world quantum hardware. The fidelity of these models heavily depends on the availability and accuracy of detailed noise data provided by hardware vendors, which is not always fully accessible.

Determining the quality of a noise model is essential for ensuring that the theoretical predictions align with experimental results. As a crucial step in software engineering, it allows developers to identify and correct inaccuracies in their models, ultimately leading to more robust and maintainable quantum software. Without this step, the gap between simulation-based designs and real-world quantum computations could widen, potentially delaying the realisation of practical quantum algorithms. However, typical similarity measures between simulation and experiment, such as the Hellinger distance (HD) [7], quickly reach its limits when increasing the number of qubits due to a lack of sufficient statistics. Therefore, we present a metric that intuitively quantifies the quality of a noise model, based on the overall quality of the corresponding practical quantum algorithms, to which the model is applied to.

In this work, we furthermore evaluate the accuracy of IBM’s publicly available noise data [8] using *Qaptiva*’s noise model implementation [9]. Our evaluation comprises experimental studies using proof-of-concept circuits, such as the Greenberger-Horne-Zeilinger (GHZ) circuit [10]. Additionally, we demonstrate the practical relevance of our approach by applying it to a real-world problem: implementing the quantum approximate optimisation algorithm (QAOA) [11] to address job shop scheduling problems [12].

Furthermore, we extrapolate the performance of quantum algorithms to future QPUs by calibrating our noise models to align with more fault-tolerant systems, which are not cur-

rently available. This allows the development and evaluation of hardware-aware and fault-tolerant algorithms, which, in their current state, cannot be implemented on actual quantum hardware.

Our software stack that integrates the noise model delivers the toolset to build a realistic and hardware-accurate simulation environment. To integrate it with the quantum software ecosystem, we utilise the *QUARK* framework [13].

The remainder of this article is structured as follows: [Sec. II](#) gives an overview of the topic in the domain of quantum software engineering, reviews related work and introduces the theoretical background to noise modelling. In [Sec. III](#), we describe our methodology, including our parametrised noisy simulator, as well as the utilised benchmark circuits and metrics. [Sec. IV](#) explains our experimental results, which are extrapolated to future QPUs in [Sec. V](#) and discussed in [Sec. VI](#). Finally, we conclude in [Sec. VII](#).

II. NOISE MODELLING: SOFTWARE PERSPECTIVE

As long as imperfections in quantum hardware persist, the development of quantum software and algorithms must account for noisy hardware, which is built on diverse paradigms and exhibits distinct properties [14]. These unique characteristics significantly influence the types of noise that can occur on a given platform. Consequently, understanding these noise patterns is essential when designing quantum algorithms or broader quantum software solutions.

However, with limited and costly access to real QPUs, the next best alternative is to develop accurate noise models of these systems. These models serve as critical tools for simulating and predicting the behaviour of quantum hardware in various scenarios, albeit simulations—especially in the noisy case—are only feasible for limited system sizes.

A. Noisy Simulation Challenges

From a quantum software engineering perspective, many existing approaches focus on higher-level abstractions rather than low-level noise modelling. This abstraction, while useful for theoretical advancements, risks oversimplifying the complexities of noise in practical systems. A general noise model may fail to capture the unique characteristics of a specific QPU, necessitating the development of detailed noise models tailored to individual hardware platforms.

The accuracy of these detailed noise models depends on two critical factors: (1) the granularity and detail of the modelling process and (2) the availability and precision of *noise data*, which describes the real-world behaviour of the QPU. Without reliable noise data, even the most sophisticated modelling techniques may fall short of capturing the true dynamics of a quantum system.

Moreover, when treating a noise model as a black box, important insights into the behaviours of near-term quantum algorithms and systems may be lost. This abstraction can hinder the ability to fine-tune algorithms or hardware configurations to mitigate specific types of noise, ultimately limiting the practical advancements in quantum computing.

B. Related Work

The integration of noise models into the quantum software stack, as a drop-in replacement of real QPUs for test and evaluation purposes of quantum algorithms, has been explored in previous studies [15]–[17]. Additionally, a growing body of literature examines the performance of quantum algorithms—such as certain state-preparation tasks [18], or variational algorithms including QAOA [4], [5], [19] and quantum machine learning [20]—under the influence of various noise conditions.

Given that variational algorithms are considered the most promising for near-term quantum systems [3], insights from these studies are particularly valuable for quantum software engineering, as highlighted by Greiwe *et al.* [14], who demonstrate general noise modelling tailored to NISQ algorithms. However, a noticeable research gap exists in directly comparing noise models with real hardware.

While comprehensive benchmarks of actual quantum hardware have been conducted, for example by Li *et al.* [21], most noise model evaluations itself are rather sparse, and focus on conceptual, instead of practical quantum algorithms. For instance, Bravo-Montes *et al.* [22] evaluated various noise models and software simulators against QPUs for small-scale logical, arithmetical, and error-correction circuits, and Grover’s algorithm. Similarly, Georgopoulos *et al.* [7] assessed different noise modelling techniques in quantum walk contexts.

These studies typically measure noise model effectiveness using state fidelities between simulations and experimental results. A notable exception is the work by Weber *et al.* [23], who introduced algorithm-specific quality metrics for variational algorithms. Our work explores a similar direction by introducing refined metrics tailored to specific applications and offering guidance on incorporating insights from noise model performance assessments into the quantum software stack.

C. Theoretical Background

The basis for the noise model is described in Refs. [7], [24] and is implemented using the Qaptiva libraries [9]. The following description makes use of the Kraus formalism with each Kraus operator K acting on the density matrix ρ as

$$\rho \rightarrow \sum_i K_i \rho K_i^\dagger. \quad (1)$$

The noise model is accounting for three main noise channel categories, (1) amplitude damping and dephasing (*i.e.* environmental effects, decoherence) (2) depolarisation (gate errors) and (3) state preparation and measurement (SPAM) errors.

Amplitude Damping describes the natural decay of the excited state $|1\rangle$ to the ground state $|0\rangle$ due to energy exchange with the environment. The corresponding Kraus operators are

$$E_0(t) = |0\rangle\langle 0| + \sqrt{\exp(-t/T_1)} |1\rangle\langle 1|, \quad (2)$$

$$E_1(t) = \sqrt{1 - \exp(-t/T_1)} |0\rangle\langle 1|, \quad (3)$$

parametrised by the relaxation time T_1 .

Pure Dephasing similarly describes the transition of a quantum system towards classical behaviour, defined by the Kraus operators

$$E_0(t) = |0\rangle\langle 0| + \sqrt{1-p(t)}|1\rangle\langle 1|, \quad (4)$$

$$E_1(t) = \sqrt{p(t)}|1\rangle\langle 1|, \quad (5)$$

with the time-dependent probability,

$$p(t) = 1 - \exp\left(-\frac{2t}{T_\varphi}\right) \quad (6)$$

$$\text{where } T_\varphi = \left(\frac{1}{T_2} - \frac{1}{T_1}\right)^{-1}, \quad (7)$$

with the pure dephasing time T_φ .

The combined effect of amplitude damping and dephasing results in the following evolution of the one qubit density matrix, which is the desired phenomenological description:

$$\begin{bmatrix} a & b \\ b^* & 1-a \end{bmatrix} \rightarrow \begin{bmatrix} (a-1)e^{-\frac{t}{T_1}} + 1 & be^{-\frac{t}{T_2}} \\ b^*e^{-\frac{t}{T_2}} & (1-a)e^{-\frac{t}{T_1}} \end{bmatrix} \quad (8)$$

In applying these channels we apply decoherence during idle times, which is a slightly different approach as in Ref. [7]. Our modelling is motivated by the assumption that the qubits interact with the environment especially during idle times of the qubits.

Depolarisation channels are parametrised in this model by the one- and two-qubit gate fidelities $\mathcal{F}_{1\text{-gate}}$ and $\mathcal{F}_{2\text{-gate}}$. The depolarisation noise quantum channel for one-qubit gates is defined by the Kraus operators

$$K_0 = \sqrt{1-p}I, \quad K_i = \sqrt{p/3}\sigma_i, \quad \text{for } i = 1, 2, 3, \quad (9)$$

with σ_i the Pauli matrices, and for two-qubit gates by

$$\{K_i \otimes K_j\}_{i,j=0\dots 3}. \quad (10)$$

The depolarisation noise quantum channel is applied after the (noiseless) quantum gate.

State preparation and measurement (SPAM) errors can be modelled with the following Kraus channels. The measurement error is described by

$$E_1 = \varepsilon_0^{\text{meas}}|0\rangle\langle 0| + (1 - \varepsilon_1^{\text{meas}})|1\rangle\langle 1|, \quad (11)$$

$$E_0 = (1 - \varepsilon_0^{\text{meas}})|0\rangle\langle 0| + \varepsilon_1^{\text{meas}}|1\rangle\langle 1|, \quad (12)$$

which is applied immediately before a measurement, with the error rates $\varepsilon_0^{\text{meas}}/\varepsilon_1^{\text{meas}}$ for mismeasuring $|0\rangle/|1\rangle$ respectively. The initialisation error, applied as the first operation, is described by the Kraus operators

$$E_0^{\text{init}} = (1 - \varepsilon_0^{\text{init}})|0\rangle\langle 0| + \varepsilon_1^{\text{init}}|1\rangle\langle 1|. \quad (13)$$

The noise parameters are summarised in Tab. I. These parameters are experimentally measurable and generally vary among the individual qubits/gates of the physical hardware. The QPU manufacturer provides these parameters in the course of the hardware calibration along with the corresponding uncertainties.

| | Parameter | Ref. Values | Noise Channel |
|-------------------------------|---------------------------|---------------------|-------------------|
| T_1 | Relaxation Time | 10^{-5} s | Amplitude Damping |
| T_2 | Dephasing Time | $\leq 2T_1$ | Dephasing |
| T_φ | Pure Dephasing Time | $\leq 2T_1$ | |
| $t_{1\text{-gate}}$ | 1-Qubit Gate Time | 10^{-8} s | Depolarisation |
| $t_{2\text{-gate}}$ | 2-Qubit Gate Times | 10^{-8} s | |
| $\mathcal{F}_{1\text{-gate}}$ | 1-Qubit Fidelities | ≈ 1 | |
| $\mathcal{F}_{2\text{-gate}}$ | 2-Qubit Fidelities | ≈ 0.97 | |
| $\varepsilon_0^{\text{meas}}$ | Error Rate of $ 0\rangle$ | $\approx 1 \%$ | |
| $\varepsilon_1^{\text{meas}}$ | Error Rate of $ 1\rangle$ | $\approx 1 \%$ | |

TABLE I: Overview of the noise parameters describing an imperfect noisy QPU. Note that the values can significantly vary for different qubits within one QPU. The provided reference values are typical for the IBMQ devices used in this work.

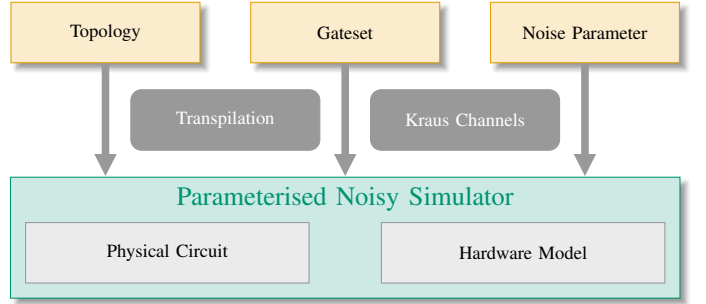


Fig. 1: Summary of the noisy simulation stack, encapsulating topology, gate specifications, environment and transpiler.

III. METHODOLOGY

A. Parametrised simulator

The software components for the noisy simulation are shown in Fig. 1. The three building blocks are the gateset, the topology (*i.e.* the connectivity of the qubits), and the noise parameters of the gates and qubits. The noise parameters, of which typical values are summarised in Tab. I, include the fidelities of the one- and two-qubit gates and the decoherence times of the qubits, as well as the readout error rate of the latter.

The performance of a quantum algorithm also strongly depends on efficient transpilation. In this work we use the Qiskit transpiler [25] for translating circuits to the IBMQ device. It is notable that the Qiskit compiler might introduce ancilla qubits, for instance to abbreviate swap-gates. This may lead to an increased number of qubits in the transpiled circuit, posing potential challenges in the simulation on classical hardware, as well as in the evaluation of the results. We therefore extend the Qiskit transpiler with our software stack, allowing to create fictitious QPUs, which can suit specific algorithms. An ideal QPU architecture derived from this co-design approach could be fed back to hardware developers.

B. Benchmark Circuits

To evaluate the performance of the noise model, we analysed various problems formulated as quantum circuits. The

specific problem formulations are outlined as follows:

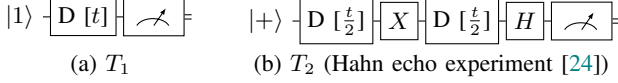


Fig. 2: Quantum circuits for determining the T_1 and T_2 times. The qubits are kept idle for a time t by applying a delay (D).

1) *Idle Circuits*: In order to verify amplitude damping and dephasing, and the corresponding decoherence times T_1 and T_2 in the noise data of IBMQ, we utilise dedicated *idle circuits*. This approach serves two primary objectives: firstly, to provide a practical sense of the magnitude of the individual noise effects; and secondly, to assess the accuracy of the noise parameter data.

a) T_1 : After initialising a qubit in the state $|1\rangle$ by applying a native \hat{X} -gate to the default initial state $|0\rangle$, a delay period t is applied. For this circuit, which is sketched in Fig. 2a, eventually the qubit is measured in the $\{0, 1\}$ basis.

b) T_2 : To determine the effects of dephasing, we replicate the *Hahn echo* experiment [24], in which a qubit is prepared on the equator of the Bloch sphere, for instance by applying a Hadamard gate to the initial $|0\rangle$ state. Subsequently, a delay t is applied, similar to the previous experiment, but with a single \hat{X} -gate applied midway through the delay to account for inhomogeneous broadening mechanisms. Finally, the state is measured in the $\{+, -\}$ basis, which is experimentally realised by applying another Hadamard gate before measurement. The protocol for this experiment is sketched in Fig. 2b.

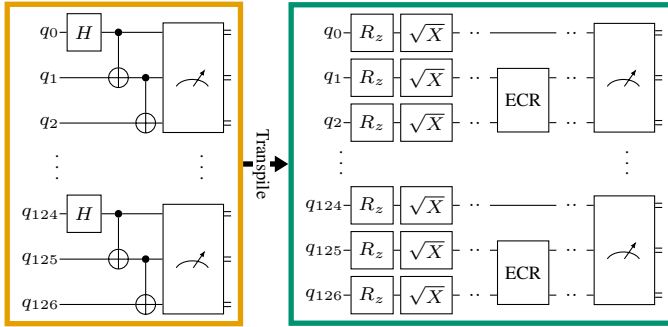


Fig. 3: Multiple three-qubit GHZ protocols, submitted in one circuit execution to an IBMQ system. In this simple case the transpilation step only consists of gate substitutions to the native gateset.

2) *GHZ*: To determine the influence of two-qubit gate errors, we prepare the Greenberger-Horne-Zeilinger (GHZ) [10] state, which is a maximally entangled state. For a general n -qubit system it is defined by

$$|\varphi_{\text{GHZ}}\rangle = \frac{|0\rangle^{\otimes n} + |1\rangle^{\otimes n}}{\sqrt{2}} \quad (14)$$

In particular, we focus on three-qubit GHZ circuits; to still gauge the performance of the full QPU, encompassing all

qubits, we submit multiple GHZ circuits, acting on physically neighbouring qubits, in one execution to the IBMQ system. Using Qiskit's [25] transpilation level 0, only non-native gates are substituted with native gates, without circuit optimisation, which allows to conduct computations on selected qubits. This process is shown in Fig. 3.

3) *Application-oriented benchmark quantum circuits*: To determine the impact of noise on practical quantum algorithms, we evaluate our noise model on the quantum approximate optimisation algorithm (QAOA), which is theoretically flexible in scalability and therefore applicable on current NISQ devices [3].

a) *Quantum Approximate Optimisation Algorithm*: The QAOA employs a quantum circuit with $p \in \mathbb{N}$ layers of unitary operators defined by $2p$ parameters $\vec{\beta}, \vec{\gamma} \in \mathbb{R}^p$. A QAOA layer j consists of two unitaries:

$$U_M(\beta_j) = e^{-i\beta_j \hat{H}_M}, \quad (15)$$

representing mixer Hamiltonian \hat{H}_M , and

$$U_C(\gamma_j) = e^{-i\gamma_j \hat{H}_C}, \quad (16)$$

based on the cost Hamiltonian \hat{H}_C , of which the ground state encodes the optimal solution to a given quadratic unconstrained binary optimisation (QUBO) problem (cf. Sec. III-B3b). The mixer unitary U_M typically consists of \hat{X} -rotations of size β_j on each qubit, while the cost unitary U_C uses single, or multi-qubit \hat{Z} -rotations of size γ_j . The initial state of the QAOA algorithm is usually chosen as the ground state of H_M , in which each qubit is in an equal superposition of $|0\rangle$ and $|1\rangle$, prepared using a layer of Hadamard gates H . A general example for a three-qubit QAOA circuit with $p = 2$ is shown in Fig. 4.

The repeated application of several QAOA layers corresponds to the discretised time evolution governed by the Hamiltonians H_M and H_C . It is known that the quality of the approximation increases with the number of layers [11], albeit the overall solution quality strongly depends on the parameter values $\vec{\beta}$ and $\vec{\gamma}$. To find optimal parameters several approaches exist, which are inspired by adiabatic time-evolution [26], [27], or utilise classical optimisation routines [28], [29]. As in this work, we are mainly interested in the influence of noise and not in QAOA parameter optimisation, we employ the parameter initialisation strategy by Montanez-Barrera et al. [26], where they use fixed linear ramp (LR) schedules. In the LR-QAOA protocol the parameters are given by

$$\beta_j = \left(\frac{j-1}{p} - 1 \right) \Delta_\beta, \quad \gamma_j = \frac{j}{p} \Delta_\gamma \quad (17)$$

for layer $j \in [1, p]$ and ramp sizes $\Delta_\beta, \Delta_\gamma \in [0, 1]$.

b) *Problem Description*: The QAOA is designed to optimise QUBO problems, to which any NP problem can be reduced [30]. The objective function is given by

$$E = \vec{x}^T Q \vec{x}, \quad (18)$$

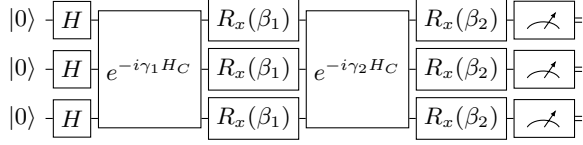


Fig. 4: QAOA circuit before the transpilation step for cost Hamiltonian H_C and $p = 2$.

for a QUBO $Q \in \mathbb{R}^{n \times n}$, and n binary variables $\vec{x}^T = (x_1, x_2, \dots, x_n) \in \{0, 1\}^n$. By switching from binary variables to spin variables ($\in \{-1, 1\}$) [30], a QUBO can equivalently be described by an Ising model \hat{H}_C that can be implemented in the QAOA.

In particular, we consider a problem from an industrially relevant use case, which is Job Shop Scheduling (JSS) [12]. The objective in JSS is to distribute a number of jobs n_j , whose production times $\{t_j\}$ vary, among a number of production machines n_m in such a way that the variance of the total processing time per machine is minimised, and thus the machines are utilised optimally¹. In general, the number of different configurations increases exponentially with $n_m^{n_j}$.

In this work, we consider two encodings for the JSS problem: (1) One-hot encoding, in which each variable (qubit) $x_{j,m}$ corresponds to the truth value of job j being executed on machine m . This may result in invalid schedules such as executing one job multiple times or not at all. Accordingly, penalty terms are added to the objective function to suppress bit-strings corresponding to invalid states. Furthermore, we use a (2) dense encoding, in which a machine on which the job j is executed is represented in an integer register $|m_j\rangle$. While the dense encoding requires less qubits, this comes at the cost of needing terms of higher order than quadratic. Therefore, the problem may no longer be defined as a QUBO, but as a polynomial unconstrained binary optimisation (PUBO) problem. However, higher-order terms are not an obstacle for a QAOA approaches, as several reductions from PUBO to QAOA circuits can be applied (see *e.g.* Refs. [31], [32] for a review). In this work, we use the implementation provided by the Qaptiva libraries [9].

The two resulting QAOA circuit classes have qualitatively different structures, summarised in Tab. II. In this work, we transpile all QAOA circuits to the QPU by using Qiskit’s transpiler with optimisation level 3, which, additionally to native gate substitutions, maps the circuit to suited qubits, and conducts further circuit optimisations [25]. The identical transpiled circuit (*i.e.* with physical qubit mappings and native gates) is then executed on an IBMQ QPU and simulated on the Qaptiva.

¹We apply some simplifications to the original use case, such as regarding all machines equal and not considering job orders

| | | | | | | |
|--------------------------|----------|-----------|----------|-----------|----------|-----------|
| JSS Size (# Jobs) | 4 | 4 | 5 | 5 | 6 | 6 |
| Encoding | Dense | 1-Hot | Dense | 1-Hot | Dense | 1-Hot |
| QAOA Depths | ≤ 8 | ≤ 24 | ≤ 8 | ≤ 16 | ≤ 8 | ≤ 16 |
| # Experiments | 25 | 90 | 25 | 35 | 25 | 35 |
| # Qubits | 4 | 8 | 5 | 10-14 | 6 | 12-14 |
| # 2-Qubit Gates* | 20 | 71 | 34 | 125 | 59 | 178 |
| Circuit Depth* | 62 | 114 | 97 | 168 | 138 | 168 |

TABLE II: Overview of all LR-QAOA benchmark circuits conducted on IBM heron device `ibm_fez`. # Experiments denotes the product of JSS instances and different QAOA depths. The lower part of the table shows the number of qubits, the number of two-qubit gates and circuit depth for problem size (JSS Size). Numbers are averaged over stochastic transpiler passes. Rows marked with * scale with QAOA depth and provide per-layer quantities.

C. Measures of Distance

A metric often used to compare probability distributions is the Hellinger distance (HD)

$$HD(p, q) = \sqrt{\frac{1}{2} \sum_i (\sqrt{p_i} - \sqrt{q_i})^2} \quad (19)$$

which is closely related to the fidelity

$$F(p, q) = \sum_i \sqrt{p_i} \sqrt{q_i} \quad (20)$$

via

$$HD(p, q)^2 = 1 - \sum_i \sqrt{p_i} \sqrt{q_i} = 1 - F(p, q). \quad (21)$$

Following Ref. [7], we use HD to compare probability distributions obtained by running GHZ circuits on the IBMQ device and on the Qaptiva simulator (cf. Sec. IV). Note that to apply this metric, probabilities p_i for every state i of the computational basis must be estimated from relative counts, which gets infeasible with an increasing qubit count as it requires exponentially increasing number of shots. The same problem arises for other commonly used distance measures such as total variation distance, Jensen-Shannon divergence or Kullback–Leibler divergence. We therefore propose an alternative way to quantify the difference between QPU results and simulation: It should be applicable to larger qubit numbers, and should ideally be related to the solution quality when applied to QAOA circuits.

D. Maximum Likelihood Estimation of Best Solution Probability

Based on the experimental observation that probability distributions resulting from an LR-QAOA circuit [26] can be described well by an exponential function in the state energy, we propose to characterise these distributions as

$$p(x) = Z^{-1} \exp(-\zeta E(x)) \quad (22)$$

where x labels the states of the computational basis, ζ is a free parameter, and $Z := Z(\zeta)$ is a normalisation factor to ascertain $\sum_x p(x) = 1$. It holds that

$$Z(\zeta) = \sum_x \exp(-\zeta E(x)). \quad (23)$$

Note the formal similarity with the Boltzmann distribution in statistical physics where ζ signifies an inverse temperature.

We expect that the exponential energy dependency of the probabilities is directly related to the success of QAOA or similar methods, as the density of states at the energy minimum (and also at the energy maximum) is typically exponentially smaller than at medium energies. To find a state near the energy minimum with a probability that is significantly larger than the uniform probability 2^n , the probability distribution must drop exponentially when moving from the energy minimum towards medium energies.

However, our approach is not limited to an exponential form, but can accommodate any piecewise analytical function. Notably, this includes probability distributions that can generally be represented by expanding the exponent (related to their moment). Incorporating higher-order terms in the exponent of Eq. 22 allows us to describe a broad class of distributions. While physical intuition on the fitting parameters might be limited in such cases, it does not reduce applicability of the current method, and is well suited to overcome the problem of distribution suffering from low statistics.

Maximum Likelihood Estimation of ζ and \tilde{p} : Assuming the evaluation of an LR-QAOA circuit with N shots that results in state x_i for n_i times ($\sum_i n_i = N$), we aim to determine the value of ζ such that this result is matched sufficiently accurate. Under the assumption that the probability to find some specific state x is given by Eq. 22, the joint probability to find the given result is

$$P(\zeta) = N! \prod_i \frac{p(x_i; \zeta)^{n_i}}{n_i!}. \quad (24)$$

Now, the maximum likelihood estimator of ζ is the value of ζ that maximises P . It is equivalent but computationally more convenient to maximise $\log(P)$. Ignoring terms that do not depend on ζ results in the condition

$$\begin{aligned} 0 &= \sum_i n_i \frac{d}{d\zeta} \log(p(x_i; \zeta)) = -N \frac{d}{d\zeta} \log(Z) - \sum_i n_i E(x_i) \\ &= N \left(\langle E \rangle_\zeta - \frac{1}{N} \sum_i n_i E(x_i) \right), \end{aligned} \quad (25)$$

where we used

$$\langle E \rangle_\zeta := Z^{-1} \sum_i E(x_i) \exp(-\zeta E(x_i)) = -\frac{d}{d\zeta} \log(Z). \quad (26)$$

Finally, the condition for ζ reads

$$\langle E \rangle_\zeta = \frac{1}{N} \sum_i n_i E(x_i). \quad (27)$$

The calculation of $\langle E \rangle_\zeta$ involves a sum over the whole state space. Nevertheless, and in contrast to the HD, it can still be calculated for large amounts of qubits: Including, in particular, cases in which the required number shots to estimate the whole probability distribution is beyond practical limits.

The statistical uncertainty of the fitting approach is evaluated by varying the fitting parameter from the optimal value such that the resulting likelihood is only 60% of the maximum likelihood. This parameter value corresponds to one standard deviation.

Taking Noise Effects into Account: Noise effects, especially measurement errors, lead to non-vanishing probabilities of high energy states. We account for this by adapting the probability distribution in Eq. 22 by

$$p(x) = Z^{-1} (\exp(-\zeta E(x)) + \delta) \quad (28)$$

with an additional parameter δ . While this expression for $p(x)$ lacks a simple analytic relation similar to Eq. 27, $P(\zeta, \delta)$ can still be maximised numerically. We maximise using the `scipy.optimize` toolkit.

In Sec. IV, we use $\tilde{p} := p(x^*)$ with x^* being a lowest energy state and $p(x)$ given by Eq. 28 with optimised parameters ζ and δ to characterise the probability distributions resulting from QAOA circuits. \tilde{p} is closely related to the probability p_0 of finding an optimal solution, which is an intuitive and practically relevant parameter. Assuming (a) the probability distribution Eq. 28 describes the circuit outcome properly, and (b) the lowest energy state is n -fold degenerated, we have $p_0 = n\tilde{p}$.

IV. NOISE MODEL VALIDATION RESULTS

A. Idle Noise Circuits

To evaluate the relaxation time T_1 and T_2 , provided in the noise model data by IBMQ [8], we submit the idle circuits, as described in Sec. III-B1 on the 127 qubit device `ibm_kyiv`. As the T_1 parameters of the individual qubits in the system are different, the decay of the prepared states can be quantitatively validated by setting the delay parameter t to different selected values (see colour coding in Fig. 5). According to the hypothetical amplitude damping of the excited state (see Eq. 8), the number of measurements with result “1” for qubit i if repeating the experiment N times, is given by $N_i = N e^{-t/T_{1,i}}$ where $T_{1,i}$ is the qubit specific T_1 -relaxation time.

In the top of Fig. 5, this function is shown as solid lines for 4 different values of t . Each dot corresponds to one qubit. Its y-coordinate is the measured 1-count and its x-coordinate is the latest available T_1 calibration data for that qubit as reported by IBMQ. The experimental results for T_1 align with the values provided by IBMQ, when submitting the circuit from Fig. 2a simultaneously on all 127 qubits. The measurement outcomes are accumulated over $N = 4096$ preparations of the same circuit.

When applying the Hahn echo circuit from Fig. 2b for T_2 similarly on all qubits simultaneously, we obtain the results shown at the bottom of Fig. 5. It is evident that results do not follow the hypothetical dephasing curves. Potential explanations for this deviation may include unintended interactions between the qubits, known as *cross-talk* [33], or a perturbation of the echo mechanism caused by temporal shifts of the operations and measurements.

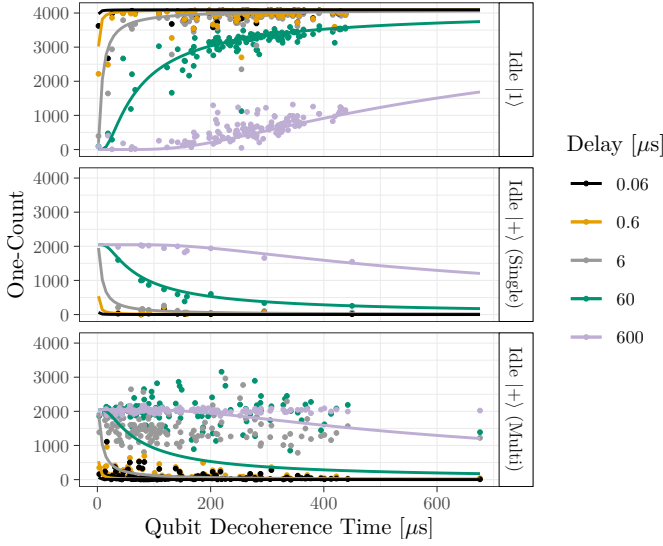


Fig. 5: Experimental results for the decay of single qubit states during the T_1 and T_2 experiments using $N = 4096$ shots and various delay times t . The different delay times are distinguished by colour. Each dot corresponds to one qubit and shows how often this qubit has been found in the 1-state. The x-axis displays the decoherence time (T_1 for the top panel, T_2 for the other two panels) as reported by IBMQ for each qubit. Solid lines show the behaviour as expected from model Eq. 8. The top panel shows the measured decay for decoherence time T_1 , compared with the theoretical exponential decay model $N e^{-t/T_1}$ for each qubit. The middle and bottom panel display the dephasing behaviour from the Hahn echo experiment compared to the theoretical model $\frac{N}{2}(1 - e^{-t/T_2})$. The middle panel refers to individually conducted qubit experiments, the bottom to simultaneously evaluated qubits.

When, instead of evaluating the circuit on all qubits simultaneously, we conduct separate evaluations of 10 individual qubits, with varying reported T_2 times, for a total of $N = 4096$ shots of the circuit the experimentally measured number of “1” results aligns with the theoretical dephasing result of $\frac{N}{2}(1 - e^{-t/T_2})$ as shown in the middle facet of Fig. 5.

B. GHZ

The results presented in Fig. 6 show the differences between the IBMQ device `ibmq_osaka`, and our simulations for the GHZ experiments described in Sec. III-B2. As illustrated through Fig. 6a, there exists a positive correlation between the HD for ideal and noisy simulations, indicating that the simulation is less accurate if the overall errors are large. Notably, the quality of the simulation depends on the accuracy of the noise model as well as on the accuracy of the available noise parameters.

As GHZ circuits are sensitive to single-qubit failures, Fig. 6b shows the dependency of the HD on the maximum occurring errors throughout the three qubits of the GHZ circuit. In that the single-qubit error refers to X and \sqrt{X} gates; the multi-qubit gate error refers to the ECR-gate error. Similarly,

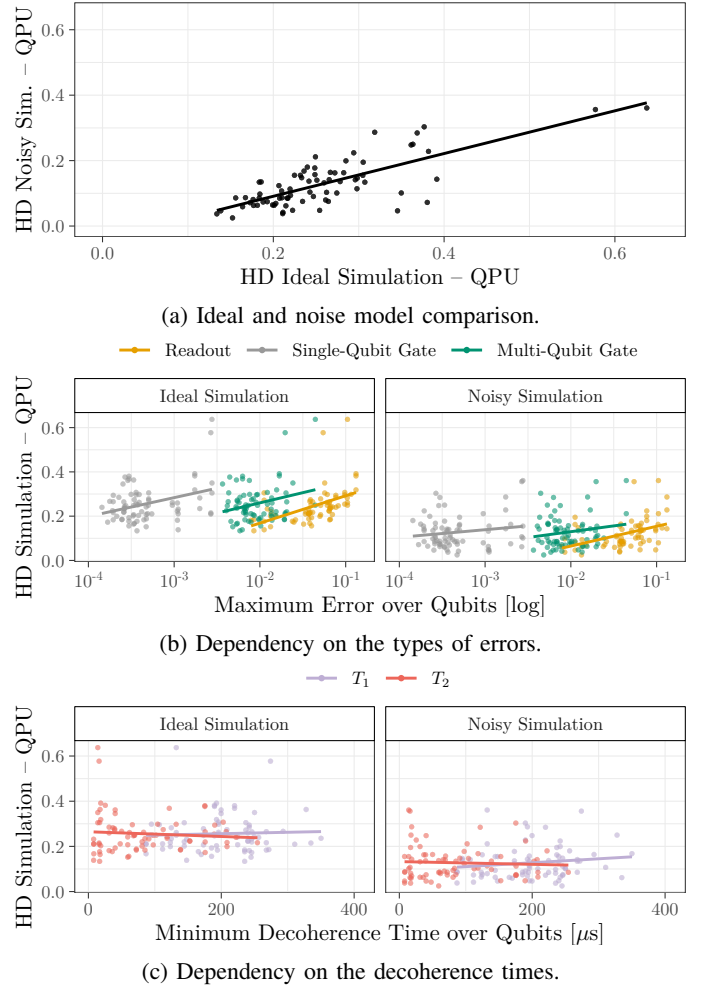


Fig. 6: Hellinger Distance (HD) of the GHZ experiments. Each point refers to the evaluation of one three-qubit GHZ circuit, simultaneously evaluated on the `ibmq_osaka` QPU, as introduced in Sec. III-B2. Each colour comprises the full set of GHZ circuits, but in dependency of different properties of the QPU. Lines represent a linear trend. In (a), the x-axis describes the HD for the evaluation on QPU and an ideal simulation. The noise data of the same set of qubits is used for a simulation with our noise model, of which the HD to the QPU results is shown on the y-axis. Subfigures (b) and (c) show the dependency on the respective maximum error and minimum decoherence time over the corresponding qubits, which is shown on the x-axis.

Fig. 6c shows the dependency on the minimum T_1 and T_2 decoherence times. The left panels show the HD of results from ideal (noiseless) simulation and from `ibmq_osaka` QPU as a function of the strength of the different noise channels as described by the IBMQ noise data. The slope of the linear trend lines in these panels gives a measure of how much the quality of the noisy modelling depends on the different noise errors. We see that the readout and gate errors are the primary sources of errors for the GHZ circuits considered here

while the decoherence times have a smaller impact. The right panels show the HD of our noisy simulations and results from `ibm_osaka` QPU. The smaller slope observed in the right panel of Fig. 6b for the readout and gate error suggests that these can adequately described with our noise model, albeit tend to be modelled less accurate.

The HD between noisy simulation and QPU is on average 0.13 ± 0.07 for a three-qubit circuit, which is comparable to the HD values obtained by the noise models in Ref. [7] for the same circuit size. This indicates that our stack for noise-model performs similar to other state-of-the-art tunable noise models. Two outliers in the execution on the real device are identified (*i.e.*, points where HD between ideal simulation and QPU exceeds 0.5), suggesting that some qubits are not functioning properly. Consequently, these outliers also provide inaccurate noise data.

We iterated the analysis using other well known distance measures (total variation distance, Jensen-Shannon divergence, Kullback–Leibler divergence) instead of HD, arriving at identical conclusions as the overall trend is equal for all tested metrics (distance ranges differ).

C. Noise Model Validation in a use case circuit (QAOA)

In this section we utilise the metric \tilde{p} as introduced in Sec. III-C for characterising the probability distributions obtained from running the circuits on the QPU or simulating them. As for small circuits we observed that the effect of amplitude damping and pure dephasing only contributes insignificantly to the overall error (cf. Fig. 6c), we neglect the impact of these types of noise in the simulations of QAOA.

To quantify the accuracy of the simulation we compare \tilde{p} as derived from the QPU results with those derived from our noisy and noiseless simulations. Fig. 7 shows the values of \tilde{p} averaged over the instances of the JSS problem, summarised in Tab. II, in dependency of the QAOA depth. The dependency of the solution quality on the QAOA depth is an important information for practical considerations of solving a given optimisation problem. As to be expected, the noiseless simulation suggests that an increase in QAOA-depth improves \tilde{p} , our measure for the success rate of the overall circuit. Taking noise into account, either by the noise model or by a real QPU one can conclude a maximal useful QAOA depth. Fig. 7 shows that the trend of the curves can qualitatively be extracted from the simulated data. For circuits with a low number of qubits the deviation of \tilde{p} between noisy simulation and QPU results is lower than 10%. This deviation increases for larger numbers of jobs but the order of magnitude and the above conclusions remains the same.

A problem-independent measure of QAOA performance is the difference between the probability to find the best solution using QAOA and the probability to find the best solution with a uniformly random bit-string, which is $p_0 = 1/2^n$. The gain ratio

$$r = \frac{\tilde{p}_{\text{noisy}} - p_0}{\tilde{p}_{\text{noiseless}} - p_0}$$

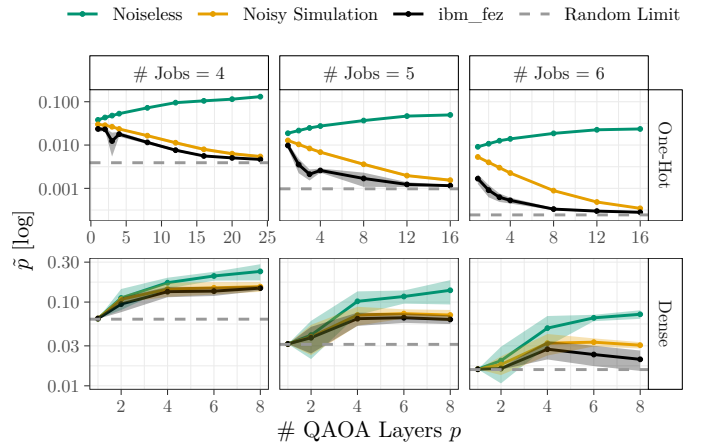


Fig. 7: The behaviour of \tilde{p} as obtained from a maximum likelihood estimation, as a function of the QAOA depth. The black line corresponds to the value obtained from the experiment conducted on the IBM Heron processor `ibm_fez`. The yellow line corresponds to the simulation performed with the Qaptiva and the noise model described above, using the noise parameters that correspond to the calibration data at the time of the experiment. The teal line refers to an ideal simulation without noise.

quantifies the impact of noise on this potential gain. \tilde{p}_{noisy} stands for either \tilde{p} as found from circuit executions on the real QPU or as found from noisy simulations.

A value of 1 for the gain ratio is achieved in the limit of a noiseless execution and becomes 0 if there is no gain of the algorithm with respect to a random choice. In Fig. 8 this quantity is shown for \tilde{p}_{noisy} as determined from QPU and the corresponding noisy simulations. Independent of the benchmark circuit, the gain ratio loosely follows an exponentially deteriorating pattern with increasing 2-qubit gate count. This is observed in both, simulation and experiment. The half-value interval of the gain ratio is ≈ 220 2-qubit gates throughout all simulated results. On real hardware this interval is smaller with ≈ 180 2-qubit gates for healthy QPU executions, indicating and quantifying the systematic underestimation of the effect of noise. The effect of statistical fluctuations of the value r due to the finite number of shots and the fitting procedure has been evaluated and found to be small ($\Delta r \approx 0.02$) compared to the spread between different circuit instances. The cause for this spread is likely the different qubit location on the QPU at which the circuit was executed on, yielding different noise parameter. Systematic uncertainties due to the inaccuracy in the calibration data have not yet been considered and are subject to future studies.

A benchmark-inclusive analysis demonstrates a strong overall agreement between the model and the actual QPU: Disregarding the strongly deviating outliers, the possible reasons for which are discussed below, a linear fit was applied to the data points of the relative deviation (lower panels of Fig. 8). This results in a deviation of less than 20% in the limit of small

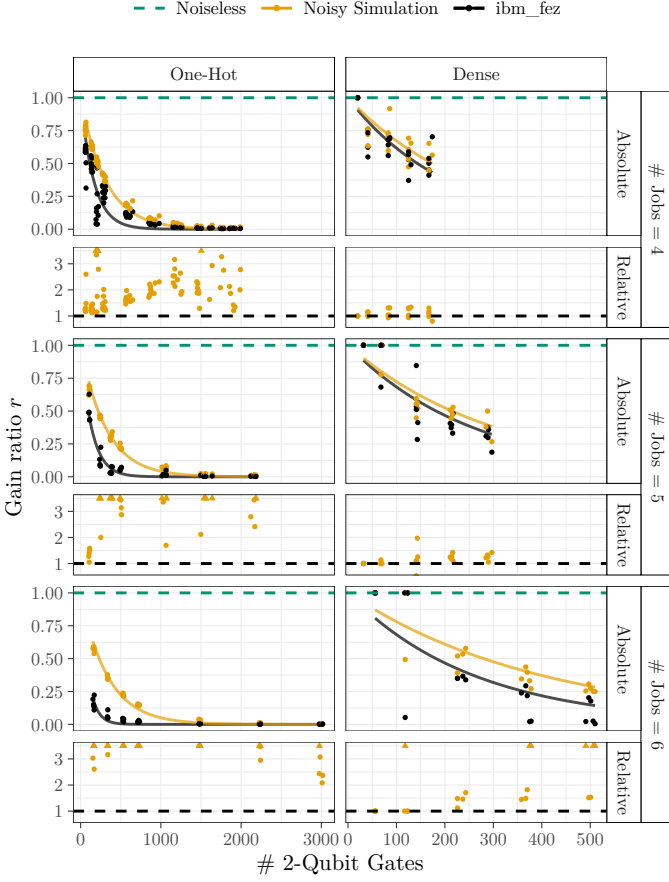


Fig. 8: The gain ratio r (*i.e.* the advantage of the quantum algorithm that is left when conducted on a noisy device) as a function of the number of two-qubit entangling gates (*i.e.* CZ) for real QPU (black) and simulation (yellow). Each point represents one conducted experiment or its simulative prediction respectively. The lines represent an exponential fit of the data points. It starts at 1 for small circuits and decreases to 0 for larger circuits (≈ 1000 CZ gates). The smaller panels show the individual relative error (*i.e.* the ratio of gain ratios) between each prediction with its real QPU counterpart.

circuits. This degree of mis-modelling increases with the size of the circuits in the order of 100% per ≈ 0.1 ms circuit execution time or ≈ 500 applied 2-qubit gates. Nevertheless, this prediction may serve as an upper limit for certain scalability considerations, as discussed in Sec. V.

When examining the individual benchmark scenarios separately, the accuracy of the noise model shows significant variability. While the model accurately describes the 4-job case in one-hot encoding, it is overly optimistic for the 5- and 6-job cases. A detailed analysis of the experimental state distributions for these circuits indicates that the most frequent states contain fewer ones than number of jobs. However, a distinctive feature of the one-hot encoding approach is that the valid (*i.e.*, the lowest-energy) states contain as many ones as there are jobs in the JSS problem. A direct correlation can

be established between each probable state and one of the valid states, which only differ by a Hamming distance of 1. This suggests that the model fails to describe certain aspects that explain the transition from ones to zeros. In fact, since the model was developed in the context of smaller circuits, qubit relaxation was considered negligible and was not simulated. The missing modelling of thermal relaxation into the ground state naturally over-predicts ones and thus states with ones equal to the number of jobs (*i.e.*, valid states), leading to a far higher expectation for \tilde{p} than observed. This explanation aligns with the linearly increasing systematic deviation for the overestimated benchmarks. Another possible explanation is imprecise calibration data. A `prep1_meas0` value (*i.e.* the probability of preparing $|1\rangle$, but measuring zero) of 10%-15% for some of the used qubits may also account for the reduced performance observed on the QPU. Such a value is not implausible, as errors with similar orders of magnitude were determined for other qubits on the same QPU. Furthermore, this explanation is consistent with the following observation made in the 4-job one-hot case: circuits with a QAOA depth of 3 were submitted to IBM on a different date than the circuits with QAOA depth of 2 and 4. The observed performance at this point in time, in terms of \tilde{p} , was significantly worse, despite no fundamental differences in the circuits themselves. A similar observation can be made in the 6-jobs dense-encoded scenario, where there are clearly two distinct clusters.

V. APPLICATION: SCALED FIDELITY

The parametrised noise model offers the possibility of scaling the fidelity to gauge the usability of solution approaches on future, improved QPUs. For this purpose, the error rate is reduced in steps of half an order of magnitude. The simulation is validated against the experimental QPU noise data, which corresponds to the scaling factor 1. In contrast to Sec. IV-C, the utilised noise parameters are constant across all qubits, corresponding to the median value of the respective parameter of the `ibm_kyiv`, to represent a more general contemporary IBM Eagle processor.

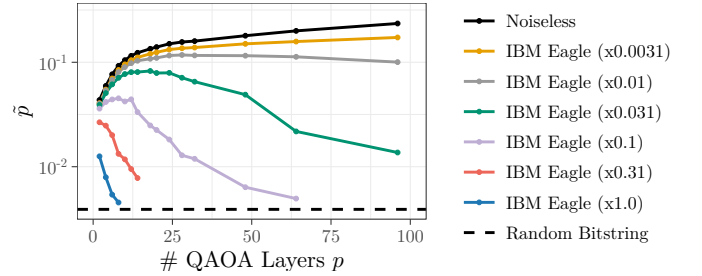


Fig. 9: The behaviour of \tilde{p} as obtained from a maximum likelihood estimation, as a function of the QAOA depth. The different lines correspond to different gate error rates, with the blue line corresponding to the median error rate of a current IBM Eagle processor.

Fig. 9 shows the quality metric \tilde{p} , under the influence of the improved QPUs for an exemplary 4-job JSS instance,

dependent on QAOA circuit depth. It is apparent that the optimal choice of QAOA depth improves progressively with more fault-tolerant systems.

VI. DISCUSSION

We tested the validity of our noise model with conceptual circuits such as the Hahn echo experiment and simultaneous three-qubit GHZ experiments. We found that in most tests the simulated results were in agreement with the experimental results, with small corresponding HDs between 0.03 and 0.12. However, in some cases, the simulation deviates considerably from the experiment, indicating that the noise parameters provided by IBM are not always accurate. Simultaneous Hahn echo experiments on a QPU show a significant difference in the final state compared to the expectation due to dephasing. The dephasing seems to occur faster with simultaneous operations, which we suspect is a consequence of cross-talk [33], that is not captured by our noise model.

Furthermore, practical LR-QAOA circuits were tested, which were generated based on a real-world JSS problem. The benchmark circuits ranged from problem sizes using 4 up to 12 qubits, some of which are even executed on 16 physical qubits after transpilation. We presented a method to assess the quality of probability distributions in large state spaces, incorporating a maximum-likelihood fit. This method was applied to validate the QAOA benchmark circuits. The likelihood fit allows us to characterise probability distributions over large state spaces (2^n) where the full distribution is not accessible due to a lack of statistics (4000 shots). In particular, in the case of the LR-QAOA, the probability of the best solution, \tilde{p} , can be derived from the fit, even if this probability is small. The parameter \tilde{p} allows to validate the noise model by comparing the resulting values of experiment and simulation. The values for \tilde{p} generated by the simulation are in good agreement with what is derived from the experiment, with a deviation of $\approx 10\%$ for short and medium-sized circuits. Still, the order of magnitude is well estimated by the applied noise model for larger circuits. The deviation here is systematic, in that the simulation delivers a \tilde{p} that is slightly too optimistic. Thus, by applying our method we are able to quantify the degree of precision of the examined noise models, and are able to identify the use cases, where the model is not sufficiently accurate. Furthermore, we occasionally experienced IBMQ QPUs delivering unusually bad results: Almost identical logical circuits deliver a significant large probability of success, which only 12 hours later yield significantly worse results. The noisy simulation is not exposed to the fluctuations described here, suggesting that it can offer a more stable environment for the development of algorithms and subroutines.

We attempted validations with different QPU architectures. Unfortunately, many vendors conduct hidden error mitigation and use internally (and intransparently) transpiled circuits, which leads to unsound results, as evaluating identical circuits for the real hardware and the simulation is crucial to validate the noise model. We need to leave this issue to future work

that can be conducted once vendors provide appropriate direct system access.

In addition, the parametrisable noise model was employed to illustrate that the exemplary LR-QAOA approaches necessitate error rates that are orders of magnitude smaller to generate a benefit. Nevertheless, this suggests the existence of promising algorithmic pathways for more fault-tolerant systems.

VII. CONCLUSION AND OUTLOOK

In this work, we presented a modular parametrisable noise model, in which core QPU properties, such as topology, gateset and noise parameters can be individually configured. This makes it a useful and (mostly) reliable tool to evaluate the performance of current and future quantum software. Scaling up simulations to larger problem sizes, and also other types of problems in future work will contribute to the practicality and accuracy of current noise models. Furthermore, our custom application-centred noise model quality metric enables to investigate the performance across different types of QPUs, albeit we need to leave a concrete instantiation to future work.

Generally, noise modelling in quantum computing is a dynamic and evolving field with significant implications for software engineering. By addressing challenges related to noise data precision, and the development of algorithm-specific metrics, we can enhance the accuracy and practicality of noise models. These advancements will not only improve the reliability of quantum algorithms but also pave the way for more efficient and effective quantum software in the future.

APPENDIX

A. Source Code Availability

A package with QUARK [13] modules providing bindings to the Qaptiva [9] libraries and containing our stack for the parametrisable noise model, is currently under active development and planned to be published in QUARK. The experimental data, as well as code for data evaluation and plot generation is available at https://github.com/lfd/make_some_noise.

Acknowledgements The authors acknowledge support by the German Federal Ministry of Education and Research (BMBF), funding program “Quantum Technologies—From Basic Research to Market”, grant numbers 13N16092 and 13N16094. WM acknowledges support by the High-Tech Agenda of the Free State of Bavaria. We thank Christopher Sowinski for his support in accessing the quantum systems and acknowledge the use of IBM Quantum services for this work. The views expressed are those of the authors, and do not reflect the official policy or position of IBM or the IBM Quantum team.

REFERENCES

- [1] R. Acharya *et al.*, “Quantum error correction below the surface code threshold,” *Nature*, Dec. 2024. DOI: [10.1038/s41586-024-08449-y](https://doi.org/10.1038/s41586-024-08449-y).
- [2] M. Aghaee *et al.*, “Interferometric single-shot parity measurement in inas-al hybrid devices,” *Nature*, vol. 638, no. 8051, pp. 651–655, Feb. 2025. DOI: [10.1038/s41586-024-08445-2](https://doi.org/10.1038/s41586-024-08445-2).
- [3] K. Bharti *et al.*, “Noisy intermediate-scale quantum algorithms,” *Rev. Mod. Phys.*, vol. 94, p. 015004, 1 Feb. 2022. DOI: [10.1103/RevModPhys.94.015004](https://doi.org/10.1103/RevModPhys.94.015004).

- [4] H. Safi, K. Wintersperger, and W. Mauerer, "Influence of hw-sw-co-design on quantum computing scalability," in *2023 IEEE International Conference on Quantum Software (QSW)*, 2023, pp. 104–115. DOI: [10.1109/QSW59989.2023.00022](https://doi.org/10.1109/QSW59989.2023.00022).
- [5] K. Wintersperger, H. Safi, and W. Mauerer, "QPU-system co-design for quantum HPC accelerators," in *Architecture of Computing Systems*, M. Schulz *et al.*, Eds., Cham: Springer International Publishing, 2022, pp. 100–114. DOI: [10.1007/978-3-031-21867-5_7](https://doi.org/10.1007/978-3-031-21867-5_7).
- [6] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information - 10th Anniversary Edition*. 2010.
- [7] K. Georgopoulos, C. Emary, and P. Zuliani, "Modeling and simulating the noisy behavior of near-term quantum computers," *Physical Review A*, vol. 104, no. 6, Dec. 2021. DOI: [10.1103/physreva.104.062432](https://doi.org/10.1103/physreva.104.062432).
- [8] IBM. "IBM Quantum." (2025), [Online]. Available: <https://quantum.ibm.com/>.
- [9] Eviden. "Qaptiva." (2025), [Online]. Available: <https://qlm35e.neasqc.eu/doc/>.
- [10] D. M. Greenberger *et al.*, "Bell's theorem without inequalities," *American Journal of Physics*, vol. 58, no. 12, pp. 1131–1143, Dec. 1990. DOI: [10.1119/1.16243](https://doi.org/10.1119/1.16243).
- [11] E. Farhi, J. Goldstone, and S. Gutmann, "A quantum approximate optimization algorithm," 2014. DOI: [10.48550/arXiv.1411.4028](https://doi.org/10.48550/arXiv.1411.4028).
- [12] H. Xiong *et al.*, "A survey of job shop scheduling problem: The types and models," *Computers & Operations Research*, vol. 142, p. 105 731, 2022. DOI: <https://doi.org/10.1016/j.cor.2022.105731>.
- [13] J. R. Finžgar *et al.*, "Quark: A framework for quantum computing application benchmarking," in *2022 IEEE International Conference on Quantum Computing and Engineering (QCE)*, 2022, pp. 226–237. DOI: [10.1109/QCE53715.2022.00042](https://doi.org/10.1109/QCE53715.2022.00042).
- [14] F. Greiwe, T. Krüger, and W. Mauerer, "Effects of imperfections on quantum algorithms: A software engineering perspective," in *2023 IEEE International Conference on Quantum Software (QSW)*, 2023, pp. 31–42. DOI: [10.1109/QSW59989.2023.00014](https://doi.org/10.1109/QSW59989.2023.00014).
- [15] C. Kai-Uwe Becker *et al.*, "Towards a quantum benchmark suite with standardized kpis," in *2022 IEEE 19th International Conference on Software Architecture Companion (ICSA-C)*, 2022, pp. 160–163. DOI: [10.1109/ICSA-C54293.2022.00038](https://doi.org/10.1109/ICSA-C54293.2022.00038).
- [16] S. Resch and U. R. Karpuzcu, "Benchmarking quantum computers and the impact of quantum noise," *ACM Comput. Surv.*, vol. 54, no. 7, Jul. 2021. DOI: [10.1145/3464420](https://doi.org/10.1145/3464420).
- [17] T. Tomesh *et al.*, "Supermarq: A scalable quantum benchmark suite," in *2022 IEEE International Symposium on High-Performance Computer Architecture (HPCA)*, IEEE, Apr. 2022, pp. 587–603. DOI: [10.1109/hpca53966.2022.00050](https://doi.org/10.1109/hpca53966.2022.00050).
- [18] F. Leymann and J. Barzen, "The bitter truth about gate-based quantum algorithms in the nisq era," *Quantum Science and Technology*, vol. 5, no. 4, p. 044 007, Sep. 2020. DOI: [10.1088/2058-9565/abae7d](https://doi.org/10.1088/2058-9565/abae7d).
- [19] S. Thelen, H. Safi, and W. Mauerer, "Approximating under the influence of quantum noise and compute power," in *2024 IEEE International Conference on Quantum Computing and Engineering (QCE)*, vol. 02, 2024, pp. 274–279. DOI: [10.1109/QCE60285.2024.10291](https://doi.org/10.1109/QCE60285.2024.10291).
- [20] M. Franz *et al.*, "Hype or heuristic? quantum reinforcement learning for join order optimisation," in *2024 IEEE International Conference on Quantum Computing and Engineering (QCE)*, vol. 01, 2024, pp. 409–420. DOI: [10.1109/QCE60285.2024.00055](https://doi.org/10.1109/QCE60285.2024.00055).
- [21] A. Li *et al.*, "Qasmbench: A low-level quantum benchmark suite for nisq evaluation and simulation," *ACM Transactions on Quantum Computing*, vol. 4, no. 2, Feb. 2023. DOI: [10.1145/3550488](https://doi.org/10.1145/3550488).
- [22] J. A. Bravo-Montes *et al.*, "A methodology to select and adjust quantum noise models through emulators: Benchmarking against real backends," *EPJ Quantum Technology*, vol. 11, no. 1, Oct. 2024. DOI: [10.1140/epjqt/s40507-024-00284-4](https://doi.org/10.1140/epjqt/s40507-024-00284-4).
- [23] T. Weber *et al.*, "Construction and volumetric benchmarking of quantum computing noise models," *Physica Scripta*, vol. 99, no. 6, p. 065 106, May 2024. DOI: [10.1088/1402-4896/ad406c](https://doi.org/10.1088/1402-4896/ad406c).
- [24] P. Krantz *et al.*, "A quantum engineer's guide to superconducting qubits," *Applied Physics Reviews*, vol. 6, no. 2, Jun. 2019. DOI: [10.1063/1.5089550](https://doi.org/10.1063/1.5089550).
- [25] A. Javadi-Abhari *et al.*, *Quantum computing with Qiskit*, 2024. DOI: [10.48550/arXiv.2405.08810](https://doi.org/10.48550/arXiv.2405.08810).
- [26] J. A. Montanez-Barrera and K. Michielsen, "Towards a universal qaoa protocol: Evidence of a scaling advantage in solving some combinatorial optimization problems," 2024. DOI: [10.48550/arXiv.2405.09169](https://doi.org/10.48550/arXiv.2405.09169).
- [27] L. Zhou *et al.*, "Quantum approximate optimization algorithm: Performance, mechanism, and implementation on near-term devices," *Physical Review X*, vol. 10, no. 2, p. 021 067, Jun. 2020, Publisher: American Physical Society. DOI: [10.1103/PhysRevX.10.021067](https://doi.org/10.1103/PhysRevX.10.021067).
- [28] M. J. D. Powell, "A direct search optimization method that models the objective and constraint functions by linear interpolation," in *Advances in Optimization and Numerical Analysis*, S. Gomez and J.-P. Hennart, Eds. Dordrecht: Springer Netherlands, 1994, pp. 51–67. DOI: [10.1007/978-94-015-8330-5_4](https://doi.org/10.1007/978-94-015-8330-5_4).
- [29] M. Periyasamy *et al.*, "Guided-spsa: Simultaneous perturbation stochastic approximation assisted by the parameter shift rule," in *2024 IEEE International Conference on Quantum Computing and Engineering (QCE)*, vol. 01, 2024, pp. 1504–1515. DOI: [10.1109/QCE60285.2024.00177](https://doi.org/10.1109/QCE60285.2024.00177).
- [30] A. Lucas, "Ising formulations of many NP problems," vol. 2, 2014. DOI: [10.3389/fphy.2014.00005](https://doi.org/10.3389/fphy.2014.00005).
- [31] L. Schmidbauer *et al.*, *It's quick to be square: Fast quadratisation for quantum toolchains*, 2024. DOI: [10.48550/arXiv.2411.19934](https://doi.org/10.48550/arXiv.2411.19934).
- [32] L. Schmidbauer *et al.*, "Polynomial reduction methods and their impact on qaoa circuits," in *2024 IEEE International Conference on Quantum Software (QSW)*, 2024, pp. 35–45. DOI: [10.1109/QSW62656.2024.00018](https://doi.org/10.1109/QSW62656.2024.00018).
- [33] P. Murali *et al.*, "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, Lausanne, Switzerland: Association for Computing Machinery, 2020, pp. 1001–1016. DOI: [10.1145/3373376.3378477](https://doi.org/10.1145/3373376.3378477).