# Variational quantum algorithms on cat qubits

Anne-Solène Bornens<sup>1</sup> and Michel Nowak<sup>1,\*</sup>

<sup>1</sup>Thales Research and Technology, Palaiseau, France

October 26, 2023

### **ABSTRACT**

Variational Quantum Algorithms (VQA) have emerged with a wide variety of applications. One question to ask is either they can efficiently be implemented and executed on existing architectures. Current hardware suffers from uncontrolled noise that can alter the expected results of one calculation. The nature of this noise is different from one technology to another. In this work, we chose to investigate a technology that is intrinsically resilient to bit-flips: cat qubits. To this end, we implement two noise models. The first one is hardware-agnostic – in the sense that it is used in the literature to cover different hardware types. The second one is specific to cat qubits. We perform simulations on two types of problems that can be formulated with VQAs (Quantum Approximate Optimization Algorithm (QAOA) and the Variational Quantum Linear Solver (VQLS)), study the impact of noise on the evolution of the cost function and extract noise level thresholds from which a noise-resilient regime can be considered. By tackling compilation issues, we discuss the need of implementing hardware-specific noise models as hardware-agnostic ones can lead to misleading conclusions regarding the regime of noise that is acceptable for an algorithm to run.

#### 1 Introduction

Quantum computing offers two types of algorithms. The ones that solely rely on the preparation and control of quantum states – referred to as end-to-end quantum algorithms – and those relying on a classical feedback loop – referred to as Variational Quantum Algorithms (VQA). Although end-to-end algorithms come with proven theoretical speedups assuming the availability of enough computational resources, they are hardly implementable on existing hardware. During the current hardware development phase (Noisy Intermediate Scale Quantum phase [1]), operations performed on physical systems will suffer from uncontrolled behavior, which can be considered as noise.

The impact of noise on VQAs has already been studied with a hardware-agnostic noise model, where Pauli gates are introduced before and after the application of unitaries [2]. However, because hardware architectures significantly differ in the way information is encoded, transformed, and finally measured, we can infer that the impact of noise on the performance of an algorithm is also hardware specific. To support this statement, we can cite [3], which already studies the impact of some physical properties of a device on the performance of a VQA. We thus take the same research path, which intends to tighten the gap between the fields of algorithm design and the physics at play in the hardware architectures.

Among the different hardware proposals for quantum computation, cat qubits [4] provide with an interesting

feature: when executing a set of identified gates, the information encoded in the cat states remains protected from bit-flip errors. The goal of the present work is to investigate this specificity by looking at the convergence of VQAs. In order to reach this goal, we first implement noise models that represent as faithfully as possible the physics of cat qubits. We then formulate a set of two problems of interest (optimization and linear system inversion), associated with cost functions. We finally conclude on the impact of noise on their minimization on a large number of configurations. This was made possible by the utilization of the large computing cluster at hand.

We structure the work in the following way: Section 2 first recalls the necessary elements to understand how cat qubits are resilient to a particular type of errors (bit-flips) for a given set of gates. Section 3 regroups the noise models and provides with the characterization of the Hadamard gate which remains unstudied in the literature. By completing the set of noise models, we are finally able to study the impact of uncontrolled operations on VQAs in Section 4. We conclude in Section 5.

## 2 Quantum computing with cat qubits

Quantum computing architectures with the discrete variable formalism (qubits) are usually built on physical systems that have two levels of energy. These two levels represent the ground and the excited state of an artificial atom (in the case of superconducting qubits) or a real one (in the case of trapped cold atoms). However, it has been shown

<sup>\*</sup>michel.nowak@thalesgroup.com

that more complex states can be chosen to encode information. In particular, information can be encoded in the coherent states of a harmonic oscillator [5]. This work exclusively focuses on this technology.

#### 2.1 Encoding information

Let  $|\alpha\rangle$  denote a coherent state of complex amplitude  $\alpha$ . We follow the notations from [6], where the convention for the cat state is taken so that:

$$|0\rangle \approx |\alpha\rangle$$
 (1)

and

$$|1\rangle \approx |-\alpha\rangle \tag{2}$$

The idea of encoding  $|0\rangle$  and  $|1\rangle$  in coherent states is a first step towards designing a bit-flip-resilient qubit.

#### 2.2 Protecting information

The second idea to fully protect information from jumping from  $|0\rangle$  to  $|1\rangle$  spontaneously is to couple the system with an other one stabilizing it. In the following, we will thus adopt the notions from Chapter 4 of [7] where a physical system described by the harmonic oscillator (called the storage) is coupled to a larger system called the reservoir. In the case of cat qubits, the reservoir is designed so as to impose specific conditions to the storage. These conditions are made so that the storage gains or loses photons by pairs. From the expression of coherent states, the information encoded in  $|0\rangle$  will be protected from transforming to  $|1\rangle$  if the rate at which photons are gained or lost by pair dominates the rate at which errors of other type happen.

Let us write the hamiltonians for the entire system. Let S be a storage system. Consider that it is coupled to a reservoir R. The hamiltonian of the total system writes:

$$H = H_S + H_R + H_{SR} \tag{3}$$

where the storage hamiltonian is:

$$H_S = \hbar\omega_s \left( a_s^{\dagger} a_s + \frac{1}{2} \right) \tag{4}$$

and the reservoir hamiltonian follows the same expression:

$$H_R = \hbar \omega_r \left( a_r^{\dagger} a_r + \frac{1}{2} \right) \tag{5}$$

with  $\omega_s$  and  $\omega_r$  respectively the associated frequencies of the storage and the reservoir.  $H_{SR}$  – the interaction hamiltonian – can be designed such as it takes the form derived in Chapter 4 of [8]:

$$H_{SR} = g_2^* (a_s^{\dagger})^2 a_r + g_2 a_s^2 a_r^{\dagger} \tag{6}$$

where  $g_2$  is a coupling constant.  $H_{SR}$  involves the square of the annihilation and the creation operators of the storage, which guarantees the desired two photon gain or loss condition. We illustrate the interactions in Figure 1 by following the same notations of reference [9], which introduces a method to engineer these conditions in practice.

The reservoir is imposed with a radiation of frequency  $\omega_p$  such that:

$$\omega_p = 2\omega_s - \omega_r \tag{7}$$

Figure 1(a) illustrates how photons are gained in pairs by the storage via the coupling with the reservoir, while Figure 1(b) shows how photons are emitted by the storage. Mathematically explaining how this process leads to an intrinsic protection against bit-flips is extensively covered in [4], whereas experimental data can be found in [10].

#### 2.3 Applying gates

This intrinsic noise bias of cat qubits needs to be conserved even during the execution of gates. Such gates that maintain this noise bias are called bias-preserving gates [4]. The idea of cat qubits is thus to exclusively use this kind of gates so as to maintain a protection against the bit-flips along the computation. Although the set of bias-preserving gates is large, it does not contain gates such that the Hadamard gate, or rotations of an arbitrary angle around the X axis (except for the X-rotation of angle  $\pi$ , which is bias-preserving). In such cases, indeed, bit-flip errors will be introduced via conversion of the phase-flip type errors into bit-flip errors through the depolarizing gates.

The idea behind cat qubits is to design logical gates [6] from these bias preserving gates, by introducing error correction codes designed to reduce the phase-flip errors. In this work, we make the assumption that gates are only applied at the physical level. We thus do not resort to error correction techniques, so as to study the possibility to compute solutions in an intermediate regime, where the number of available qubits is not large enough to allow for error correction.

In order to perform some operations on variational circuits, we need the Hadamard gate. We are aware that introducing a depolarizing gate will remove one part (if not all) of the advantage of cat qubits, but we need it to compile a VQA. Consider the circuit depicted in Figure 2. It shows how to perform a Hadamard gate on cat qubits. Although the original paper in which this circuit is proposed stands for logical gates, one can still use it at the physical level.

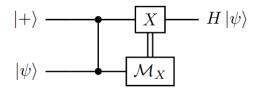


Figure 2: Hadamard circuit on cat qubits architectures adapted from [6]. The initial state is  $|\psi\rangle$ .

An ancilla register is prepared in the  $|+\rangle$  state. Then, a CZ gate follows between the ancilla register and the register on which we wish to apply the Hadamrd gate. An X

(a) Absorption of two photons by the storage

(b) Emission of two photons by the storage

Figure 1: Conversion process between the storage and the reservoir [9]

gate is finally applied on the ancilla register conditioned on the measurement in the X basis of the initial state register. The ancilla register finally contains the desired output state.

#### 3 Noise models

We now tackle the problem of implementing a relevant noise model for our study of VQAs. Section 3.1 regroups the main contributions of this work, where the implementation of a noise model specific to cat qubits is provided. Section 3.2 then covers a theoretical and hardware-agnostic noise model proposed in [2] along with some adaptations to better compare with the cat qubits model.

## 3.1 Cat qubits model

Let us now introduce the noise model for the cat qubits technology. As mentioned in Section 2, the encoding of cat qubits relies on the fine manipulation of coherent states constrained to gain or lose photons by pairs. One first source of error originates from non respecting this constraint, thus losing photons one by one. A second source of error comes from the fact that the gates are applied within a certain time-frame, leading to so-called non adiabatic effects. The gate duration is optimized to maximise the gate fidelity taking into account these two phenomena [6]. Note that errors will happen asymmetrically on the control or target qubits, and this asymmetry is taken into account in the probabilistic noise models of [11].

For the Hadamard gate, we consider the circuit in Figure 2 proposed in [6] at the logical level. We wish to use this strategy to implement the Hadamard gate at the physical level. We want a simple realistic error model of this Hadamard gate for our simulations. This model could

be directly derived from the circuit. However, this would require to add at least one ancilla to each circuit we simulate. To overcome this burden, we choose to seek for a single qubit noise model, as presented in Figure 3.

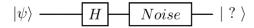


Figure 3: Equivalence of noisy hadamard circuit on cat qubits for simulation

As in [11] we only consider the errors at the first order in the sense that we neglect the contributions of multiple errors accumulating in the small Hadamard circuit. This gives a first approximation of the level of noise that one can expect from this circuit. The corresponding model is presented in bold in Table 1.

#### 3.2 Hardware-agnostic model

Recent work makes the assumption that noise can be modelled by adding Pauli gates between each unitary in the circuit [2]. They assume that each unitary is taken as being one layer of the circuit. Even if the nature of the layer depends on the algorithm (QAOA [13], VQLS [14], VQC [15], ...), we can say that it encapsulates one application of a chosen ansatz. We will call this model the hardware-agnostic layer-wise model. One modification that we apply compared to [2] is that we restrict the errors to be of X and Z type only. This is to be able to compare the noise models more appropriately.

This model allowed to derive potential predictions on the performance of the gradient descent in variational algorithms. We here recall the main result of this work. Consider the probabilities of adding a X, Y of Z on each qubit

Physical level	I		$\mathcal{P}_{ +\rangle}$		$Z(\theta)$		X		H		CZ		CNOT		Toffoli	
	Error	Probability	Error	Probability	Error	Probability	Error	Probability	Error	Probability	Error	Probability	Error	Probability	Error	Probability
	I	1-p	I	1-p	I	1-p	I	1-p	I	1 - 5p	I	1-2p	I	1-4p	I	1 - 6p
	Z	p	Z	p	Z	p	Z	p	$\mathbf{z}$	3p	$Z_1$	p	$Z_1$	3p	$Z_1$	p
									X	$^{2}\mathrm{p}$	$Z_2$	p	$Z_2$	p/2	$Z_2$	p
												-	$Z_1Z_2$	p/2	$Z_3$	p/2
														- /	$CZ_{12}$	3p
															$CZ_{12}Z_{3}$	p/2

Table 1: Final error model. The bold probabilities correspond to what one needs for VQAs. The explicit expression of most of the probabilities can be found in [11] and [12].

after the application of a layer of the VQA:  $q_X, q_Y, q_Z$ . Define p as being the maximum of this set of probabilities

$$p = \sqrt{\max\{q_X, q_Y, q_Z\}} \tag{8}$$

Given these assumptions, an upper bound on the partial derivative of the cost function has been derived and can indicate at which regime the algorithm will suffer from poor convergence.

**Theorem 1** (Upper bound on the partial derivative [2]). Consider an L-layered ansatz. Suppose that local Pauli noise with parameter p acts before and after each layer with parameter  $\theta$ . Then the following bound holds for the partial derivative of the noisy cost function.

$$\left| \frac{\partial \mathcal{C}}{\partial \theta} \right| \le A \, n^{1/2} p^{L+1} \tag{9}$$

where A is a constant of the problem that does not depend on the number of qubits n and on the noise level p.

The cat qubits specific model adds noise at the application of each gate. This makes the agnostic model a priori more optimistic than the specific one. We thus decide to apply the noisy gates in the hardware-agnostic model at each gate so that to be able to compare to the cat qubits specific model. We will refer to this noise model as the hardware-agnostic gate-based noise model.

Finally, because hardware providers do not propose the same set of native gates, we chose to constrain the hardware-agnostic model to be compiled with the same gates as the cat qubits one. This model will be adapted if the Toffoli gate is present in the algorithm, and separate simulations with the gate based model will be carried out with and without allowing the Toffoli gate to be present in the set of basis gates for the compilation step since this gate is native with cat qubits but not with most of other existing technologies.

All operations that can be performed in parallel are done in parallel and not sequentially.

## 3.3 Comparing the models

With these models in hand, comparing the cost function convergence will allow to extract the impact of the asymetric noise of cat qubits compared to a uniformly depolarizing channel.

We need to set the definition of the noise level p which will vary from  $10^{-9}$  to  $10^{-1}$ . We chose to set it to the phase flip probability of the Z gate, which is what is

taken in [11] to introduce the noise model. We justify this choice by stating that this unit of error is the unique thing that is common between the different noise models.

One last consideration that needs to be tackled is that the cat qubits model is designed to take into account errors on idle qubits while other experience an application of a gate. In the cat qubits model, a phase flip probability is applied with probability p. We chose to apply the same scheme to the hardware-agnostic model, but with a depolarizing probability of p for the phase flip error and p for the bit-flip error.

We will conclude on the relevance of this choice at the moment of the results' analysis.

### 4 Numerical Experiments

The idea of VQAs is to encode the problem of interest into the minimization of a cost function with some part of the circuit being dependent on varying parameters [16]. For each application, the structure of the circuit is chosen so that the expectation of some measurement is proportional to the cost function. Depending on the outcome of the experiment, a classical optimizer can tune the parameters according to the gradient of the cost function.

Without being comprehensive, but with the hope to cover the widest range of applications, we choose to focus on two particularly challenging setups. First, optimization problems are frequently being taken to analyse VQAs [2, 3], we thus include the Quantum Approximate Optimization Algorithm (QAOA) [13] into our study. Then, a more recent VQA has emerged to solve linear systems [14]: the Variational Quantum Linear Solver (VQLS) which can be helpful to solve computational physics problems.

For each parameterized algorithm, we select setups where we can increase the size of the problem so that to exhibit the limits of convergence.

#### 4.1 Noise models implementation

In order to perform the simulations with the two noise models, we need to implement them. On a quantum computer, the noise would arise at each sample of the circuit. The correct strategy to simulate the noise according to the failure rate probability p would thus be to randomly introduce gates with respect to this probability at each shot. This can be performed with the custom noise models proposed by Qiskit.

In order not to miss certain gates that do not have a noise model implemented, for example  $R_y$  gates, we transpile the circuit with the following gates:  $R_z$ , CX, CCX, X, Z, H. When the CCX gate will be present, we will also remove it from the set of available gates for transpilation to test for the potential appearance of discrepancies compared to the native toffoli gate in the cat qubits technology.

#### 4.2 Quantum Approximate Optimization Algorithm

We first focus on the MaxCut problem, that can be solved with QAOA [17]. Let G be an unweighted graph on n vertices. A cut is a partition of the graph's vertices into two complementary sets S and T. The MaxCut problem consists in finding a cut such that the number of edges between S and T is as large as possible.

#### 4.2.1 Algorithm implementation

We wish to formulate this problem so that it can be solved with QAOA. Figure 4 recalls the structure of the circuit used to solve an optimization problem. Alternating layers of two successive unitaries are applied to a register initially prepared in the  $|+\rangle^{\otimes n}$  state.

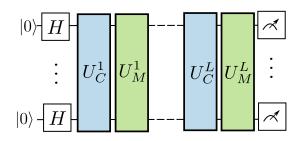


Figure 4: Circuit for QAOA

The first unitary is the mixing one and is built on the following hamiltonian by introducing degrees of freedom  $\beta$ :

$$U_M(\beta) = e^{-i\beta H_M} \quad \text{with} \quad \beta \in [0, 2\pi]$$
 (10)

and

$$H_M = \sum_{i=1}^n X^i \tag{11}$$

The second unitary is the problem one and is built so that to encode the problem of interest. It will also be parameterized by another set of degrees of freedom  $\gamma$ :

$$U_C(\gamma) = e^{-i\gamma H_C} \quad \text{with} \quad \gamma \in [0, 2\pi]$$
 (12)

In order to find an expression for  $H_C$ , we need to derive a cost function that will encode the MaxCut problem when minimizing it. To do so, we define a classical cost function C(x)

$$C(x) = 1 - \frac{1}{2n_E(G)} \sum_{(i,j) \in E(G)} (1 - x_i x_j)$$
 (13)

where  $x=\{x_i\}_{i\in [\![1,n]\!]}$  is the ensemble of the vertices, such as  $x_i=1$  if  $x_i\in\mathcal{S}$  and  $x_i=-1$  if  $x_i\in\mathcal{T}$ . E(G) is the set of edges of the graph and  $n_E(G)$  is the number of edges of G. The normalization by  $n_E(G)$  is added to allow comparisons between different graphs of the same number of nodes.

In order to map this problem to QAOA, we define the problem hamiltonian as:

$$H_C = 1 - \frac{1}{2n_E(G)} \sum_{(i,j) \in E(G)} (1 - Z^i Z^j)$$
 (14)

with  $Z^i$  the Z operator that acts on the ith qubit. Alternately applying the two unitaries with L repetitions, we get the parameterised final state of the problem:

$$|\psi_L(\gamma,\beta)\rangle = U_M^L(\beta_L)U_C^L(\gamma_L)\cdots U_M^1(\beta_1)U_C^1(\gamma_1)|+\rangle^{\otimes n}$$
(15)

We finally measure all qubits and compute the cost func-

$$C_{OAOA}(\gamma, \beta) = \langle \psi_L(\gamma, \beta) | H_C | \psi_L(\gamma, \beta) \rangle$$
 (16)

#### 4.2.2 Results analysis

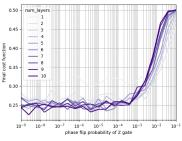
The idea is to observe how the introduction of noise into the quantum circuit of the algorithm has an impact on its optimized solution and to analyse in theses conditions the influence of the size of the graph  $(n \in [4;9])$ , the number of layers  $(L \in [1;10])$  and the noise strength  $(p \in [10^{-9};10^{-1}])$ . For the sake of simplicity, we fix the number of qubits to 5 in the main text.

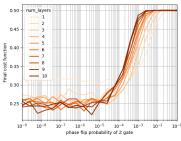
We execute different series of simulations by varying these parameters. In order to make sure we do not work on particular graphs, each configuration is averaged on 100 randomly generated graphs of the same size, connecting each pair of vertices with probability 0.6. We can then observe the final cost function value after convergence in different setups.

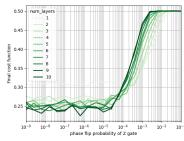
We start by looking at the layer-wise hardware agnostic model which is the current reference in the literature [2].

In Figure 5(a), we fix the number of qubits to 5 and vary the number of layers starting from 1 to 10. We observe the final value of the cost function obtained after convergence against the level of noise introduced in the simulation. For large levels of noise, we identify a regime in which the noise is strong enough so that the cost function converges to 0.5 which corresponds to a random choice of edges cut. Depending on the number of layers, the final value of the cost function will start diverging from 0.5. It then enters a transient regime where the cost function exponentially decreases towards a reference level - which we will refer to as the noise resilient regime.

Figure 5(b) shows the same results for the hardware-agnostic gate-based model. The trend of the curves is the same, however, the thresholds are shifted towards lower values. This is due to the fact that noise is introduced







- (a) Layer-wise hardware-agnostic
- (b) Parallel gate-based hardware-agnostic
- (c) Parallel gate-based cat qubits

Figure 5: MaxCut QAOA: Evolution of the final cost function reached after convergence depending on the number of layers ranging from 1 to 10 for different noise models 5(a) layer-wise hardware-agnostic. 5(b) parallel hardware-agnostic. 5(c) parallel cat qubits model.

with the same probability at each gate instead of each layer with the same noise model (uniformly depolarizing). We draw a first conclusion on this result: the noise model of [2] is relevant to obtain the trend of the results even in an approximated regime where noise is introduced layerwise.

Figure 5(c) shows the results for the cat qubits model. We observe that the trend is the same as the ones for the hardware-agnostic models.

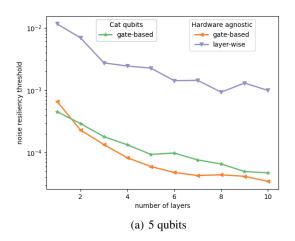


Figure 6: MaxCut QAOA noise resiliency thresholds evolution against the number of layers for 5 qubits

We can then extract noise resiliency thresholds from which we can consider the algorithm to be unaffected by the presence of noise. We thus implement a threshold detection routine that extracts the value from which the convergence enters the transient regime.

Figure 6 shows the evolution of the threshold against the number of layers for the two hardware agnostic models and for the cat qubits model. Appendix A regroups all the configurations simulated. We observe that the thresholds

shifts to lower values as the number of layers increases. This could be explained by the Theorem 1, which stands that the performance of the algorithm might degrade exponentially with the number of layers considering noise and  $|\partial C/\partial\theta| \leq Dp^{L+1}$  with  $D \in \mathbb{R}$ . We note that the hardware-agnostic model compiled in parallel (gate-based) and the cat qubits model overlap at low values of number of layers but starts diverging as the number of layers increases. The thresholds for the cat qubits model are slightly higher that the thresholds of the hardware-agnostic one. Appendix A confirms this observation when varying the number of qubits up to 10. The layer-wise model reaches the noise resilient regime earlier than the others, and this was expected given that noise is introduced at each layer and not at each time step of the circuit.

#### 4.3 Variational Quantum Linear Solver

We wish to study an algorithm that outputs an estimation  $\hat{x}$  of the solution to the problem

$$Ax = b (17)$$

where A is a matrix and b a target vector.

#### 4.3.1 Algorithm implementation

The goal of the VQLS algorithm [14] is to produce a quantum state  $|\psi\rangle=A\,|\hat{x}\rangle$  that is proportional to  $|b\rangle$ , the representation of b in amplitudes. The encoding of  $|\hat{x}\rangle$  is performed thanks to variational circuit  $V(\theta)$  acting on a register that is initially prepared in the  $|0\rangle$  state. We thus have  $|\hat{x}\rangle=V(\theta)\,|0\rangle$ . A classical optimization loop is then used to tune the parameters  $\theta$  with respect to the minimization of a cost function. In order to encode the problem in this cost function, we define it in the following way [14]:

$$C_{\text{SWAP test}}(\theta) = 1 - \left| \langle b | \psi(\theta) \rangle \right|^2 \tag{18}$$

Minimizing  $C_{\text{SWAP test}}(\theta)$  will tune the parameters  $\theta$  such that  $|\psi(\theta)\rangle$  is close to  $|b\rangle$ . The distance contribution is estimated with the circuit depicted in Figure 7.

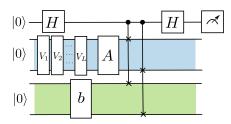


Figure 7: Circuit for VQLS [14].  $V_i$  represent the i-th layer of the variational circuit.

To study the robustness of the algorithm, we select a simple problem and set A to the identity, and b to a flat vector. Simulations with different level of noise ranging from  $10^{-9}$  to  $10^{-1}$  were performed by varying the number of layers and the size of the problem (number of qubits). In order to have a statistically relevant estimation of the cost function, we run the simulations 100 times and average the final cost function obtained after convergence.

Note that the two Hadamard gates acting on the ancilla register are bias-preserving. Indeed, the first one prepares the  $|+\rangle$  state and the last one measures in the + basis[11].

#### 4.3.2 Results analysis

We fix the problem to 7 qubits in the main text, thus inverting matrices of size  $8\times 8$  (3 qubits for the target register, 3 qubits for matrix register, 1 ancilla) and only look at gatebased noise models. Appendix B regroups configurations with matrices of size  $4\times 4$  (5 qubits)  $8\times 8$  (7 qubits)  $16\times 16$  (9 qubits) and  $32\times 2$  (11 qubits). In Figure 8, we select the hardware-agnostic noise model with 8(a) and without 8(b) a native Toffoli. Figure 8(c) shows the results for the cat qubits model. For each configuration, we identify the same 3 phases of convergence depending on the level of noise introduced in the circuit.

We first observe that the noise resilient regimes reach a lower level of final cost function while increasing the number of layers. Then, we observe that increasing the number of layers diminishes the noise resiliency threshold, which was already observed for QAOA.

In Figure 9, we compare the behavior of the noise models. The thresholds for the hardware-agnostic model compiled in parallel and considering that the Toffoli is a native gate are lower than the ones with the cat qubits model.

If the Toffoli gate is not considered as native and is instead compiled with other one and two qubit gates, the depth of the circuit increases and the threshold is altered towards a lower value. For a low number of layers, the main contributions to the errors originates from the Toffoli layers. As the number of layers increases, the gap between the native and non native Toffoli gates decreases.

This indicates that the fact that the Toffoli gate is native for the cat qubits model might offer an advantage over a technology that needs to compile it with other gates. In the NISQ regime, the depth introduced by the compilation of the Toffoli gate is large enough compared to the total depth of the circuit, which constrains the averaged error rate introduced in the circuit.

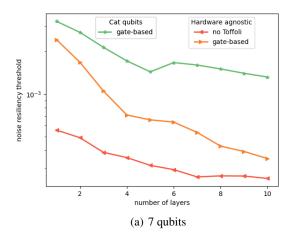


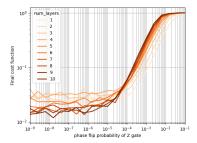
Figure 9: VQLS noise resiliency thresholds against the number of layers for 7 qubits.

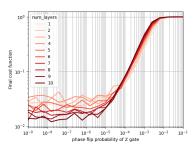
#### 5 Conclusion

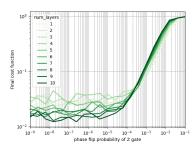
Cat qubits are thought to be used in a fault tolerant architecture for which there exist demonstrations of reduced overhead when introducing error correcting codes compared to other technologies. However, the intrinsic property of protection against bit-flips lead us to think that some gain could be obtained at the physical level, before introducing error correction codes. The goal of this work was thus to quantify the convergence of Variational Quantum Algorithms on cat qubits. While finalizing our work, we became aware of a study carried out in parallel with a different approach that focuses on the performance of the QAOA algorithm with cat qubits [18].

To reach this goal, we first started by extracting the relevant information on the noise models in the cat qubits formalism from the literature. We proposed a way to integrate the Hadamard gate, thus introducing bit-flips in the computation.

We then applied the resulting noise model to two problems of interest that can be tackled with VQAs, optimization and linear system inversion. For optimization, we chose a problem that is frequently used for benchmarking, the MaxCut problem. VQLS was used for the system inversion on the identity with a simple target vector. We studied the final value of the cost function after convergence for different levels of noise, for different problem sizes, and for different degrees of freedom of the algorithm - namely the number of layers at which each ansatz is repeated. This lead us to determine thresholds for the







(a) Hardware-agnostic gate-based with (b) Parallel hardware-agnostic gate-based Toffoli without Toffoli

(c) Parallel gate-based Cat qubits

Figure 8: VQLS: Evolution of the final cost function reached after convergence depending on the number of layers ranging from 1 to 10 for different noise models 8(a) parallel hardware-agnostic with a native Toffoli. 8(b) parallel hardware-agnostic without a native Toffoli. 8(c) parallel cat qubits model.

noise levels from which the convergence can be considered in a noise-resilient regime.

A hardware-agnostic noise model allowed us to compare our hardware-specific results to a generic model, with a symmetric depolarization. We observed that the convergence behaves similarly for the two types of models, which confirms that the hardware-agnostic model can be used to have an idea of the behavior of the algorithm in the presence of noise. However, we observe that the acceptable regime of noise varies depending on the chosen noise model for the two algorithms studied. For QAOA, the discrepancy is low, but still increases with the number of layers. For the swap test the discrepancy significantly increases with the number of layers. By introducing compilation constraints as the presence of a native Toffoli gate, we show that the additional depth required decreases the noise resiliency threshold. Concluding on the potential advantage of the cat qubits physical model over a symmetric depolarising technology is not as straightforward as expected. One has to take into account compilation and choosing a relevant common reference between the noise models (the Z failure rate for our study).

## Acknowledgement

The authors thank Jérémie Guillaud for his support in understanding the cat qubits formalism and for giving feedback.

### References

- [1] John Preskill. Quantum computing in the nisq era and beyond. *Quantum*, 2:79, 2018.
- [2] Samson Wang, Enrico Fontana, Marco Cerezo, Kunal Sharma, Akira Sone, Lukasz Cincio, and Patrick J Coles. Noise-induced barren plateaus in variational quantum algorithms. *Nature communications*, 12(1):1–11, 2021.

- [3] Loïc Henriet. Robustness to spontaneous emission of a variational quantum algorithm. *Physical Review A*, 101(1):012335, 2020.
- [4] Jérémie Guillaud, Joachim Cohen, and Mazyar Mirrahimi. Quantum computation with cat qubits. *arXiv* preprint arXiv:2203.03222, 2022.
- [5] Mazyar Mirrahimi, Zaki Leghtas, Victor V Albert, Steven Touzard, Robert J Schoelkopf, Liang Jiang, and Michel H Devoret. Dynamically protected catqubits: a new paradigm for universal quantum computation. *New Journal of Physics*, 16(4):045014, 2014.
- [6] Jérémie Guillaud and Mazyar Mirrahimi. Repetition cat qubits for fault-tolerant quantum computation. *Physical Review X*, 9(4):041053, 2019.
- [7] Claude Cohen-Tannoudji, Jacques Dupont-Roc, and Gilbert Grynberg. *Processus d'interaction entre photons et atomes*. Edp Sciences, 2012.
- [8] Raphaël Lescanne. Engineering multi-photon dissipation in superconducting circuits for quantum error correction. PhD thesis, Université Paris sciences et lettres, 2020.
- [9] Zaki Leghtas, Steven Touzard, Ioan M Pop, Angela Kou, Brian Vlastakis, Andrei Petrenko, Katrina M Sliwa, Anirudh Narla, Shyam Shankar, Michael J Hatridge, et al. Confining the state of light to a quantum manifold by engineered two-photon loss. *Science*, 347(6224):853–857, 2015.
- [10] Raphaël Lescanne, Marius Villiers, Théau Peronnin, Alain Sarlette, Matthieu Delbecq, Benjamin Huard, Takis Kontos, Mazyar Mirrahimi, and Zaki Leghtas. Exponential suppression of bit-flips in a qubit encoded in an oscillator. *Nature Physics*, 16(5):509–513, 2020.
- [11] Jérémie Guillaud and Mazyar Mirrahimi. Error rates and resource overheads of repetition cat qubits. *Physical Review A*, 103(4):042413, 2021.

- [12] Christopher Chamberland, Kyungjoo Noh, Patricio Arrangoiz-Arriola, Earl T. Campbell, Connor T. Hann, Joseph Iverson, Harald Putterman, Thomas C. Bohdanowicz, Steven T. Flammia, Andrew Keller, Gil Refael, John Preskill, Liang Jiang, Amir H. Safavi-Naeini, Oskar Painter, and Fernando G.S.L. Brandão. Building a fault-tolerant quantum computer using concatenated cat codes. *PRX Quantum*, 3:010329, Feb 2022.
- [13] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. A quantum approximate optimization algorithm. *arXiv preprint arXiv:1411.4028*, 2014.
- [14] Carlos Bravo-Prieto, Ryan LaRose, Marco Cerezo, Yigit Subasi, Lukasz Cincio, and Patrick J Coles. Variational quantum linear solver. *arXiv preprint arXiv:1909.05820*, 2019.

- [15] Maria Schuld, Alex Bocharov, Krysta M Svore, and Nathan Wiebe. Circuit-centric quantum classifiers. *Physical Review A*, 101(3):032308, 2020.
- [16] Marco Cerezo, Andrew Arrasmith, Ryan Babbush, Simon C Benjamin, Suguru Endo, Keisuke Fujii, Jarrod R McClean, Kosuke Mitarai, Xiao Yuan, Lukasz Cincio, et al. Variational quantum algorithms. *Nature Reviews Physics*, 3(9):625–644, 2021.
- [17] https://qiskit.org/textbook/chapplications/qaoa.html.
- [18] Pontus Vikstål, Laura García-Álvarez, Shruti Puri, and Giulia Ferrini. Quantum approximate optimization algorithm with cat qubits, 2023.

# A QAOA noise resiliency thresholds

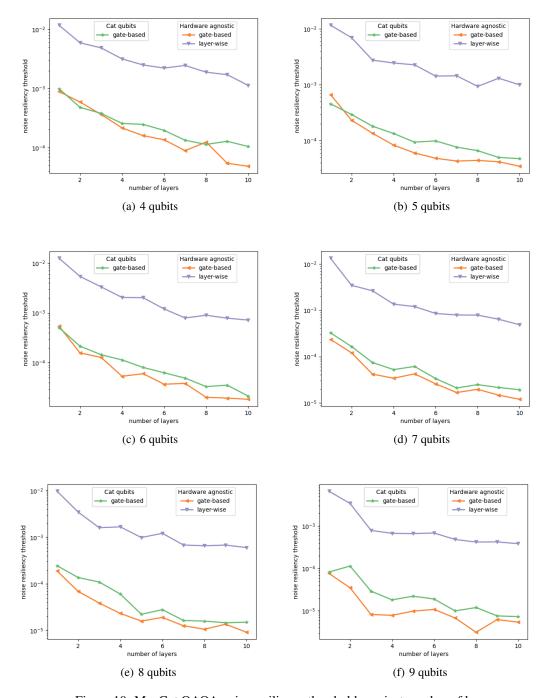


Figure 10: MaxCut QAOA noise resiliency thresholds against number of layers

# B VQLS noise resiliency thresholds

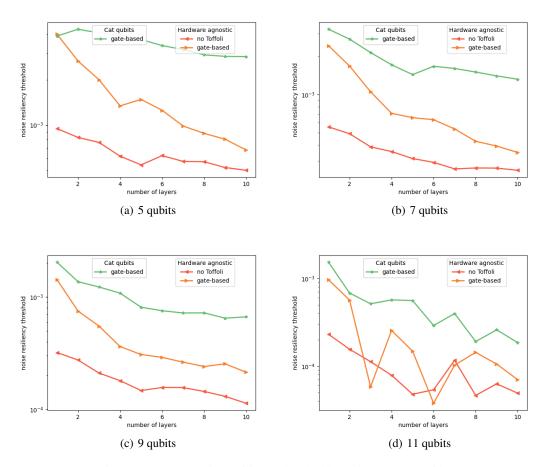


Figure 11: VQLS noise resiliency thresholds against number of layers