Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Angle Domain Gaussian Perturbations via Ancilla Hamming Weight

Ismail Habib

ismail.habib@gmail.com

INTI International University and Colleges

Tin Tin Ting

INTI International University and Colleges

Research Article

Keywords: Quantum circuits, Gaussian perturbations, Angle encoding, Ancilla qubits, Hamming weight, Quantum sampling, Explainable AI, perturbation

Posted Date: April 26th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-9487446/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Angle Domain Gaussian Perturbations via Ancilla Hamming Weight

Ismail Habib[1,2*] and Tin Tin Ting[2,3†]

[1*]Information Systems, INTI International University and Colleges Persiaran Perdana BBN, Putra Nilai, 71800 Nilai, Negeri Sembilan, Malaysia.

*Corresponding author(s). E-mail(s): ismail.habib@gmail.com;
Contributing authors: tintin.ting@newinti.edu.my;
†These authors contributed equally to this work.

Abstract

We introduce a quantum circuit that works in the angle domain to generate local, approximately Gaussian perturbations around a given vector. Instead of encoding a full distribution into quantum amplitudes, our method operates directly on rotation angles, making it better suited for localized perturbations rather than global distribution loading.

For each feature qubit, we add $\boldsymbol{m}$ ancilla qubits prepared in the $|+\rangle$ state and apply the same controlled $\boldsymbol{R_y(\alpha)}$ rotation from each of them. Together, these rotations create a shift that depends on the number of ancillas measured in the state $|\mathbf{1}\rangle$ (the Hamming weight $\boldsymbol{W}$). After applying a centering correction, the resulting perturbation follows a discrete distribution:

$$\boldsymbol{x' = x + \frac{\alpha}{a}\left(W - \frac{m}{2}\right), \quad W \sim \text{Binomial}(m, \tfrac{1}{2}).}$$

We derive closed-form expressions for the mean and variance of the perturbation and show that selecting

$$\boldsymbol{\alpha = \frac{2a\sigma}{\sqrt{m}}}$$

yields a target variance $\boldsymbol{\sigma^2}$. While the finite-$\boldsymbol{m}$ distribution is discrete, it converges to a Gaussian distribution under standardization via the Central Limit Theorem, with approximation error governed by Berry–Esseen bounds.

We implement the method using [1] and illustrate how it behaves in low-dimensional settings. Overall, the approach can be seen as a simple and

lightweight way to generate perturbations for angle-encoded data, making it useful for applications like explainability (e.g., LIME), robustness analysis, and data augmentation in hybrid quantum-classical workflows.

Keywords: Quantum circuits, Gaussian perturbations, Angle encoding, Ancilla qubits, Hamming weight, Quantum sampling, Explainable AI, perturbation

1 Introduction

Gaussian perturbations are widely used in tasks such as model explainability, sensitivity analysis, and data augmentation. For example, methods like Local Interpretable Model-Agnostic Explanations (LIME) generate samples around a target instance to understand how a model behaves locally. This means that the quality and structure of the perturbation distribution are crucial for producing accurate and stable explanations.

In the quantum setting, most research on probability distributions has focused on preparing full distributions encoded in quantum amplitudes. Early work, such as Grover–Rudolph state preparation [2], shows how to build quantum states that represent efficiently integrable probability density functions. This idea has been further developed in applications like quantum risk analysis [3] and amplitude estimation pipelines [4–6], which combine distribution loading with estimation techniques. More recently, tensor-network approaches, such as matrix product states, have been used to efficiently represent smooth distributions, including Gaussian states [7–9].

Another direction of research looks at variational and generative methods, such as quantum circuit Born machines and quantum generative adversarial networks (qGANs), which learn distributions from data by tuning parameterized quantum circuits [10, 11]. Although these approaches are quite powerful, they are designed to model entire distributions, rather than generate small perturbations around a specific input.

In contrast, generating *Gaussian perturbations around a given vector* has received relatively little attention in the quantum literature. Unlike approaches that aim to prepare full distributions, this problem starts from a known vector and focuses on generating samples in its vicinity with controllable statistical properties. This type of capability is particularly useful in hybrid quantum-classical workflows that require local exploration of the input space.

In this work, we introduce *Angle Domain Gaussian Perturbations via Ancilla Hamming Weight (ADGPAH)*, a quantum circuit that works directly in the angle domain. Instead of encoding a full distribution, the idea is to perturb an already encoded vector by applying controlled rotations based on the Hamming weight of ancilla qubits prepared in the $|+\rangle$ state. Each ancilla adds a small rotation, and together they create a centered random shift in the feature qubit.

The perturbations follow a discrete distribution based on a binomial random variable. As the number of ancillas grows, this distribution becomes increasingly Gaussian

after standardization, as expected from the Central Limit Theorem. Unlike amplitude-preparation methods, our approach is lightweight and avoids the need to compute cumulative distribution functions or build deep circuits.

The main contributions of this paper are summarized as follows:

- We propose a novel angle-domain perturbation mechanism based on ancilla Hamming weight, designed for sampling around an encoded vector.
- We derive the exact finite-sample distribution of the perturbations and provide closed-form expressions for mean and variance.
- We show how to calibrate circuit parameters to match a desired variance and analyze the asymptotic convergence to a Gaussian distribution.
- We provide a proof-of-concept implementation using [1] and demonstrate the method in low-dimensional settings.

It is important to note that this method is not designed to replace general-purpose distribution loading. Instead, it complements existing approaches by offering a simple and controllable way to generate perturbations.

2 Literature Review

There has been significant research on using quantum systems to generate and sample probability distributions, usually by encoding complete distributions into quantum states. Here, we review the main approaches and explain how our work relates to them.

2.1 Amplitude Preparation Methods

One of the most common ways to represent probability distributions in quantum systems is through amplitude encoding. Grover and Rudolph [2] showed that if a probability density function (PDF) is easy to integrate, it can be discretized and translated into the amplitudes of a quantum state using controlled rotations. The idea is to repeatedly split the cumulative distribution function (CDF), building the distribution step by step in a tree-like sequence of rotations.

Building on this idea, later work has explored practical use cases. One example is quantum risk analysis [3], which combines amplitude loading with amplitude estimation [4, 5] to estimate financial risk. Although effective, these approaches depend on accurately computing cumulative probabilities and tend to produce deep circuits, which are difficult to run on today's quantum devices.

2.2 Repeat-Until-Success and Incremental Construction Methods

To reduce the depth and precision challenges of amplitude preparation, researchers have explored repeat-until-success strategies. For example, the method proposed by Rattew et al. [12], building on earlier work in repeat-until-success circuit synthesis [13], constructs the desired distribution step by step using probabilistic updates and intermediate measurements.

Intuitively, these approaches build the desired distribution step by step by adding many small, nearly independent contributions, much like the idea behind the Central Limit Theorem. They reduce circuit depth and qubit usage by reusing ancilla qubits, but this introduces randomness into the execution and requires classical feedback to guide the process.

2.3 Tensor-Network and Structured State Preparation

More recently, researchers have turned to tensor-network methods to represent structured probability distributions more efficiently. Matrix Product State (MPS) approaches, for example, allow smooth distributions such as Gaussians to be captured in a compact form with fewer resources [7–9]. The key idea is to take advantage of the low entanglement typically present in these distributions, which makes state preparation more efficient.

Another line of work uses quantum Fourier transforms to shape distributions in the amplitude domain. While this idea is elegant in theory, it is quite sensitive to noise and depends on highly accurate operations, which makes it difficult to apply on current quantum hardware.

2.4 Variational and Generative Models

Variational quantum circuits are often used for generative modeling. For example, quantum circuit Born machines (QCBMs) and quantum generative adversarial networks (qGANs) learn probability distributions from data by tuning the parameters of quantum circuits [10, 11]. These methods are quite flexible and can model complex distributions, but they depend on a training process and the availability of suitable data.

An important point is that these approaches are inherently *global*: they aim to learn the full distribution, rather than generate small perturbations around a particular input.

2.5 Other Approaches

There are also other approaches, such as amplitude estimation pipelines [6, 14], linear-combination-of-unitaries (LCU) methods [15, 16], and sampling techniques based on quantum random access memory (QRAM) [17]. While these methods are powerful, they usually assume that the distribution is already encoded or can be accessed through structured data.

2.6 Positioning of This Work

Most of the methods described above aim to model or prepare full probability distributions. In contrast, our focus is on a different task: generating small perturbations around a specific vector that has already been encoded in a quantum circuit.

Our method works directly in the angle domain and uses the Hamming weight of ancilla qubits to create controlled perturbations. Unlike amplitude-based methods, it does not need to encode a full distribution, and unlike variational approaches, it does

not require training or data. Instead, it provides a simple and analytically clear way to generate perturbations with a desired mean and variance.

This distinction matters because our approach is not meant to replace existing distribution-loading methods, but to complement them by providing a simple way to perform localized sampling in hybrid quantum-classical workflows.

3 Proposed Methodology

In this section, we formalize the proposed angle-domain perturbation mechanism and derive its statistical properties.

3.1 Problem Definition

Let $x = (x_1, \ldots, x_d) \in \mathbb{R}^d$ be a target vector. Each feature is encoded into a quantum state via an affine angle encoding:

$$\theta_i = a_i x_i + b_i, \quad a_i \neq 0.$$

Our objective is to generate perturbed samples x' such that:

$$\mathbb{E}[x'] = x, \quad \mathrm{Cov}(x') = \mathrm{diag}(\sigma_1^2, \ldots, \sigma_d^2),$$

for user-specified variances σ_i^2.

We focus on the *independent (diagonal covariance) case* in this work.

3.2 Circuit Construction

For each feature i, the circuit is constructed as follows:

1. **Feature Encoding:** Prepare a feature qubit in the state
$$R_y(\theta_i)\,|0\rangle\,.$$
2. **Ancilla Initialization:** Allocate m ancilla qubits initialized in the superposition state:
$$|+\rangle^{\otimes m}\,.$$
3. **Controlled Rotations:** Apply m identical controlled rotations:
$$CR_y(\alpha_i),$$
where each ancilla qubit acts as a control for a rotation on the feature qubit.
4. **Centering Rotation:** Apply a single unconditional rotation:
$$R_y\left(-\frac{m}{2}\alpha_i\right).$$

Since all rotations are about the y-axis, they commute and can be combined additively.

3.3 Perturbation Mechanism

Let $z \in \{0,1\}^m$ denote the measurement outcome of the ancilla register and define its Hamming weight:

$$W = \sum_{k=1}^{m} z_k.$$

Because the ancillas are prepared in $|+\rangle^{\otimes m}$, we have:

$$W \sim \text{Binomial}(m, \tfrac{1}{2}).$$

Conditioned on W, the total rotation applied to the feature qubit becomes:

$$\theta_i' = \theta_i + \alpha_i \left(W - \frac{m}{2}\right).$$

Decoding back to the data domain gives:

$$x_i' = \frac{\theta_i' - b_i}{a_i} = x_i + \frac{\alpha_i}{a_i}\left(W - \frac{m}{2}\right).$$

3.4 Statistical Properties

We now state the main result governing the perturbation.

Proposition 1. For each feature i, the perturbed variable x_i' satisfies:

$$\mathbb{E}[x_i'] = x_i, \quad \text{Var}(x_i') = \frac{\alpha_i^2}{a_i^2} \cdot \frac{m}{4}.$$

Proof. Since $W \sim \text{Binomial}(m, \frac{1}{2})$, we have:

$$\mathbb{E}[W] = \frac{m}{2}, \quad \text{Var}(W) = \frac{m}{4}.$$

Substituting into the expression for x_i':

$$\mathbb{E}[x_i'] = x_i + \frac{\alpha_i}{a_i}\left(\mathbb{E}[W] - \frac{m}{2}\right) = x_i,$$

$$\text{Var}(x_i') = \left(\frac{\alpha_i}{a_i}\right)^2 \text{Var}(W) = \frac{\alpha_i^2}{a_i^2} \cdot \frac{m}{4}.$$

□

3.5 Variance Calibration

To achieve a target variance σ_i^2, we choose:

$$\alpha_i = \frac{2a_i\sigma_i}{\sqrt{m}}.$$

Substituting into the variance expression yields:

$$\mathrm{Var}(x_i') = \sigma_i^2.$$

3.6 Discrete Nature and Resolution

The perturbation is discrete, since W takes integer values. The step size between adjacent values is:

$$\Delta_i = \frac{\alpha_i}{a_i} = \frac{2\sigma_i}{\sqrt{m}}.$$

Thus, the resolution improves as m increases, with step-to-spread ratio:

$$\frac{\Delta_i}{\sigma_i} = \frac{2}{\sqrt{m}}.$$

3.7 Asymptotic Gaussian Behavior

Although the finite-m distribution is discrete, it converges to a Gaussian distribution under standardization:

$$\frac{W - m/2}{\sqrt{m/4}} \xrightarrow{d} \mathcal{N}(0,1),$$

by the Central Limit Theorem.

The convergence rate is governed by the Berry–Esseen theorem [18], which bounds the approximation error by:

$$O\left(\frac{1}{\sqrt{m}}\right).$$

Therefore, the perturbation x_i' approaches a Gaussian distribution as m increases, with a trade-off between smoothness and circuit resources.

3.8 Multivariate Case

For d features, each feature uses an independent ancilla block, yielding:

$$\mathbb{E}[x'] = x, \quad \mathrm{Cov}(x') = \mathrm{diag}(\sigma_1^2, \ldots, \sigma_d^2).$$

Extension to correlated covariance structures is not implemented in this work and is left for future research.

4 Implementation and Experimental Evaluation

4.1 Implementation Details

We implement the proposed perturbation circuit using [1], following the approach described in Section 4. In practice, each feature qubit is coupled with its own set of m ancilla qubits, which operate independently.

We run the experiments using the `AerSimulator` backend with the `matrix_product_state` simulation method. This method uses tensor-network techniques to efficiently simulate circuits with structured entanglement, which makes it a good fit for the repeated controlled-rotation patterns in our design.

Before execution, all circuits are transpiled, and measurements are collected from both the feature and ancilla registers. The results are then processed classically to recover the samples x', by calculating the Hamming weight of each ancilla block and applying the decoding formula:

$$x'_i = x_i + \frac{\alpha_i}{a_i}\left(W_i - \frac{m}{2}\right).$$

4.2 Experimental Setup

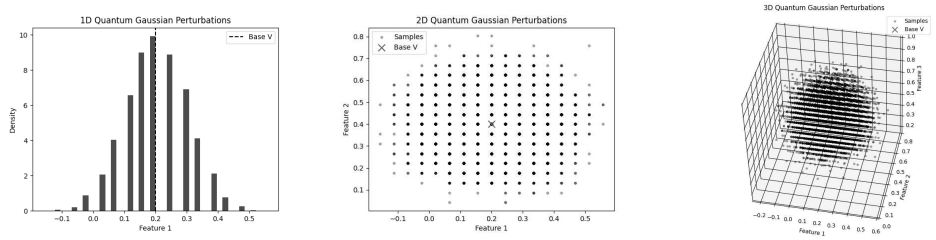
We evaluate the method on low-dimensional vectors to illustrate its behavior:

For all experiments, we use:

$$m = 20, \quad \text{shots} = 20{,}000.$$

The perturbation variance is controlled through the parameter α_i as defined in Section 4.

Figure 1 illustrates the generated samples for the 1D, 2D, and 3D cases. The samples are centered around the target vector and exhibit symmetric dispersion consistent with the intended perturbation mechanism. In higher dimensions, the sample clouds



(a) 1D (b) 2D (c) 3D

Fig. 1: Generated samples in (a) 1D, (b) 2D, and (c) 3D for $m = 20$ and 20,000 shots. The samples are centered around the target vector and illustrate the approximate Gaussian perturbation behavior.

form approximately isotropic clusters (under equal variance settings), reflecting the independence of features in the current construction.

4.3 Qualitative Results

4.4 Discussion of Distribution Behavior

The observed distributions reflect the underlying binomial structure of the perturbation mechanism. For $m = 20$, the number of distinct values per feature is limited to $m + 1 = 21$, resulting in a discrete set of possible perturbation levels.

However, due to the Central Limit Theorem, the distribution of samples exhibits an approximately Gaussian shape when visualized, particularly in aggregate. This effect becomes more pronounced as m increases, at the cost of additional circuit resources.

The trade-off can be summarized as follows:

- Increasing m improves smoothness and Gaussian approximation
- Increasing m increases qubit count and circuit depth

4.5 Limitations of Current Experiments

The current experimental results are qualitative and intended as a proof of concept. A more comprehensive evaluation would include:

- **Moment validation:** Empirical mean and variance compared to theoretical values
- **Distributional metrics:** Kolmogorov–Smirnov (KS) distance or Wasserstein distance to a target Gaussian
- **Convergence analysis:** Study of approximation quality as a function of m
- **Baseline comparison:** Comparison with classical Gaussian sampling and quantum amplitude-preparation methods

These evaluations are left for future work.

4.6 Interpretation

At a high level, the sampling process is determined by the measurement outcomes of the ancilla register. Since the ancillas start in $|+\rangle$ states, their outcomes are uniformly distributed across all bitstrings, which naturally leads to a binomial distribution in terms of the Hamming weight.

In other words, the perturbations are produced by systematically transforming the measurement outcomes, rather than by directly encoding a Gaussian distribution in the amplitudes. This makes it clear that the method is designed as an angle-domain perturbation mechanism, not a general-purpose distribution generator.

4.7 Code Availability

The implementation of the proposed method, including circuit construction and post-processing routines, is publicly available at: https://github.com/ihabib1983/AngleDomainGaussianPerturbationsviaAncillaHammingWeight

The repository contains reproducible scripts for generating the experimental results presented in this paper.

5 Complexity Analysis

In this section, we analyze the resource requirements of the proposed perturbation method in terms of qubits, gate counts, circuit depth, and classical post-processing.

5.1 Qubit Requirements

For a d-dimensional input vector, the circuit uses:

- d feature qubits (one per dimension)
- dm ancilla qubits (each feature uses m ancillas)

Thus, the total number of qubits is:

$$n = d(m + 1).$$

5.2 Classical Bit Requirements

The measurement results include:

- d bits for feature qubits (optional for decoding)
- dm bits for ancilla qubits

Thus, the total classical output size per shot is:

$$d(m + 1).$$

If decoding is performed solely from the ancilla register, only dm bits are required.

5.3 Gate Complexity

For each feature i, the circuit consists of:

- **Encoding:** 1 single-qubit rotation $R_y(\theta_i)$
- **Ancilla Preparation:** m Hadamard gates
- **Controlled Rotations:** m controlled-$R_y(\alpha_i)$ gates
- **Centering:** 1 single-qubit rotation

Aggregating across all d features:

- **Single-qubit gates:**

$$d \text{ (encoding)} + dm \text{ (Hadamards)} + d \text{ (centering)} = dm + 2d$$

- **Two-qubit gates:**

$$dm \text{ controlled rotations}$$

5.4 Circuit Depth

The depth depends on the execution model:

Sequential execution (per feature):

All m controlled rotations act on the same feature qubit and must be applied sequentially:

$$\text{Depth per feature} = O(m).$$

Parallel execution (across features):

Since features are independent, all d feature blocks can be executed in parallel:

$$\text{Total depth} = O(m).$$

Fully serialized execution:

If hardware constraints prevent parallelism:

$$\text{Total depth} = O(dm).$$

5.5 Alternative Depth Optimization

The dominant depth arises from applying m controlled rotations per feature. An alternative approach is to explicitly compute the Hamming weight using a quantum adder circuit [19–21] and apply a single rotation conditioned on the result.

This approach can reduce depth to $O(\log m)$ at the cost of additional qubits and circuit complexity. We do not implement this variant in the current work.

5.6 Runtime Considerations

Each shot of the circuit produces one sample. For S shots:

- **Quantum execution cost:** $O(S \cdot m)$ (depth-dominated)
- **Classical post-processing:**

  The decoding step computes the Hamming weight for each feature:

  $$O(d \cdot m) \quad \text{per sample.}$$

  Thus, total classical processing cost is:

  $$O(S \cdot d \cdot m).$$

5.7 Scalability Discussion

The method exhibits the following trade-offs:

- Increasing m improves Gaussian approximation but increases qubit count and depth
- Increasing d scales linearly in both qubits and gates
- The method is naturally parallel across features

Compared to amplitude-preparation methods, which often require deep recursive circuits, the proposed approach uses a simpler and more modular structure,

making it potentially more suitable for near-term quantum devices, particularly in low-dimensional settings.

6 Conclusion

In this work, we introduced a simple angle-domain quantum circuit for generating perturbations around an encoded vector using the Hamming weight of ancilla qubits. The idea is to apply controlled rotations from ancillas prepared in the $|+\rangle$ state, which produces perturbations whose size follows a binomial distribution.

We derived the exact distribution of the perturbation for a finite number of samples and showed that its mean and variance can be controlled analytically. By choosing the rotation parameter carefully, the method can produce perturbations with a desired variance. Although the distribution is discrete when only a limited number of ancillas are used, it becomes increasingly Gaussian after standardization as the number of ancillas grows, following the Central Limit Theorem and Berry–Esseen bounds.

This approach is fundamentally different from existing methods for preparing quantum distributions. Instead of encoding a full distribution in the amplitudes, it focuses on a simple perturbation mechanism in the angle domain. As a result, it is especially useful for hybrid quantum-classical workflows that need localized sampling, such as explainability, robustness analysis, and data augmentation.

That said, the method also has some important limitations. For finite circuit sizes, the perturbations are still discrete, and improving the Gaussian approximation requires more ancillas, which increases the circuit resources. The current design also only supports independent perturbations across features, and extending it to handle correlations is still an open challenge. Finally, this work does not claim any quantum advantage over classical methods and should be seen mainly as a methodological contribution rather than a performance improvement.

There are several directions for future work. One is to extend the method to handle full covariance structures, for example by introducing correlations between ancillas or using coherent transformations. Another is to evaluate the method more thoroughly by measuring its accuracy and comparing it with classical and quantum baselines. Finally, we plan to apply this perturbation primitive in complete workflows, such as quantum-enhanced explainability and sensitivity analysis.

Overall, this work provides a simple and analytically tractable building block for perturbation in quantum circuits, complementing existing approaches to quantum distribution preparation.

Funding

The authors did not receive any specific funding for this work.

References

[1] Qiskit Development Team: Qiskit: An open-source SDK for quantum computing. https://github.com/Qiskit/qiskit. Accessed: 2026-02-01 (2024)

[2] Grover, L., Rudolph, T.: Creating superpositions that correspond to efficiently integrable probability distributions. arXiv preprint quant-ph/0208112 (2002)

[3] Woerner, S., Egger, D.J.: Quantum risk analysis. npj Quantum Information **5**(1), 15 (2019)

[4] Brassard, G., Hoyer, P., Mosca, M., Tapp, A.: Quantum amplitude amplification and estimation. arXiv preprint quant-ph/0005055 (2000)

[5] Montanaro, A.: Quantum speedup of monte carlo methods. Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences **471**(2181), 20150301 (2015)

[6] Grinko, D., Gacon, J., Zoufal, C., Woerner, S.: Iterative quantum amplitude estimation. npj Quantum Information **7**(1), 52 (2021)

[7] Iaconis, J., Johri, S., Zhu, E.Y.: Quantum state preparation of normal distributions using matrix product states. npj Quantum Information **10**(1), 15 (2024)

[8] Smith, K.C., Khan, A., Clark, B.K., Girvin, S.M., Wei, T.-C.: Constant-depth preparation of matrix product states with adaptive quantum circuits. PRX Quantum **5**(3), 030344 (2024)

[9] Sano, Y., Hamamura, I.: Quantum state preparation for probability distributions with reflection symmetry using matrix-product states. Physical Review Applied **24**(3), 034062 (2025)

[10] Benedetti, M., Garcia-Pintos, D., Perdomo, O., Leyton-Ortega, V., Nam, Y., Perdomo-Ortiz, A.: A generative modeling approach for benchmarking and training shallow quantum circuits. npj Quantum information **5**(1), 45 (2019)

[11] Zoufal, C., Lucchi, A., Woerner, S.: Quantum generative adversarial networks for learning and loading random distributions. npj Quantum Information **5**(1), 103 (2019)

[12] Rattew, A.G., Sun, Y., Minssen, P., Pistoia, M.: The efficient preparation of normal distributions in quantum registers. Quantum **5**, 609 (2021)

[13] Bocharov, A., Roetteler, M., Svore, K.M.: Efficient synthesis of universal repeat-until-success quantum circuits. Physical review letters **114**(8), 080502 (2015)

[14] Suzuki, Y., Uno, S., Raymond, R., Tanaka, T., Onodera, T., Yamamoto, N.: Amplitude estimation without phase estimation: Y. suzuki et al. Quantum Information Processing **19**(2), 75 (2020)

[15] Childs, A.M., Wiebe, N.: Hamiltonian simulation using linear combinations of unitary operations. arXiv preprint arXiv:1202.5822 (2012)

[16] Low, G.H., Chuang, I.L.: Hamiltonian Simulation by Qubitization. Quantum 3 (July 2019), 163 (2019)

[17] Giovannetti, V., Lloyd, S., Maccone, L.: Quantum random access memory. Physical review letters **100**(16), 160501 (2008)

[18] Shevtsova, I.G.: On the absolute constants in the berry–esseen inequality and its structural and nonuniform improvements. Informatika i Ee Primeneniya [Informatics and its Applications] **7**(1), 124–125 (2013)

[19] Draper, T.G., Kutin, S.A., Rains, E.M., Svore, K.M.: A logarithmic-depth quantum carry-lookahead adder. arXiv preprint quant-ph/0406142 (2004)

[20] Cuccaro, S.A., Draper, T.G., Kutin, S.A., Moulton, D.P.: A new quantum ripple-carry addition circuit. arXiv preprint quant-ph/0410184 (2004)

[21] Gidney, C.: Halving the cost of quantum addition. Quantum **2**, 74 (2018)