Journal of Combinatorics
Volume 14, Number 4, 445–459, 2023

Counting abelian squares efficiently for a problem in quantum computing*

Ryan Bennink

I describe how the number of abelian squares of given length relates to a certain problem in theoretical quantum computing, and I present a recursive formula for calculating the number of abelian squares of length $t+t$ over an alphabet of size d. The presented formula is similar to a previously known formula but has substantially lower complexity for large d, a key improvement resulting in a practical solution to the original application.

AMS 2000 subject classifications: 05A05, 68Q12.
Keywords and phrases: Abelian squares, quantum computing, quantum circuit expressiveness, variational methods.

1. Introduction

An abelian square is a word whose first half is an anagram of its second half, for example `intestines` = `intes` · `tines` or `bonbon` = `bon` · `bon`. Abelian squares have been a subject of pure math research for many decades [9, 19, 16, 3, 4, 15, 8] but are seemingly not encountered often in scientific applications. Here I describe an application of abelian squares to a problem in the field of quantum computing. This application motivated the development of a recursive formula, presented here, for efficiently calculating the number $f_d(t)$ of abelian squares of length $t+t$ over an alphabet of size d. While the formula derived here is similar to a previously known formula [30],

arXiv: 2203.11886
arXiv: 2208.02360
*This manuscript has been authored by UT-Battelle, LLC, under contract DE-AC05-00OR22725 with the US Department of Energy (DOE). The US government retains and the publisher, by accepting the article for publication, acknowledges that the US government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this manuscript, or allow others to do so, for US government purposes. DOE will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan (http://energy.gov/downloads/doe-public-access-plan).

it is computationally efficient even when d is large, a key improvement which enables a practical solution to the original application. This work highlights the sometimes surprising connections between pure math and applied science, and the value of efficiently computable formulas for practitioners in applied fields.

In the first part of this Letter I review the basics of enumerating abelian squares, derive a new recursive formula for calculating their number, and provide a constructive interpretation for the formula. In the second part I describe the problem of quantifying the expressiveness of parameterized quantum circuits, show how for a particular subclass of circuits it reduces to the problem of counting abelian squares over an exponentially large alphabet, and utilize the new formula to quantify the expressiveness of that subclass of circuits.

2. Counting abelian squares

2.1. Background

Let $f_d(t)$ denote the number of abelian squares of length $t+t$ over an alphabet of d symbols. Trivially, $f_1(t) = 1$ for all t and $f_d(0) = 1$ for all d. It is also not difficult to see that $f_d(1) = d$. To determine $f_d(t)$ for arbitrary d and t, we define the *signature* (sometimes called the Parikh vector) of a word $w \in \{a_1, \ldots, a_d\}^*$ as $(m_1, \ldots, m_d)$ where m_i is the number of times the symbol a_i appears in w. Note that two words are anagrams if and only if they have the same signature. Thus the number of abelian squares is the number of pairs (x, y) such that x and y have the same signature. The number of words with a particular signature $(m_1, \ldots, m_d)$ is given by the multinomial coefficient

$$\binom{m_1 + \cdots + m_d}{m_1, \ldots, m_d} = \frac{(m_1 + \cdots + m_d)!}{m_1! \cdots m_d!}. \tag{1}$$

The number of ways to choose a pair of words, each with signature $(m_1, \ldots, m_d)$, is just the square of this quantity. Therefore the number of abelian squares of length $t + t$ is

$$f_d(t) = \sum_{m_1+\cdots+m_d=t} \binom{t}{m_1, \ldots, m_d}^2 \tag{2}$$

where the sum is implicitly over nonnegative integers. The first few values of $f_d(t)$ are shown in Table 1.

Table 1: Number of abelian squares of length $t + t$ over an alphabet of size d [30]

$d\backslash t$	0	1	2	3	4	5	6	7
1	1	1	1	1	1	1	1	1
2	1	2	6	20	70	252	924	3432
3	1	3	15	93	639	4653	35169	272835
4	1	4	28	256	2716	31504	387136	4951552
5	1	5	45	545	7885	127905	2241225	41467725
6	1	6	66	996	18306	384156	8848236	218040696

Eq. (2) is not easy to evaluate when t is large, as the number of signatures grows combinatorially in d and t. Richmond and Shallit [30] derived a recursive formula using a simple constructive argument: To create a size (t, t) abelian word pair (x, y) over alphabet $\{a_1, \ldots, a_d\}$, first choose the number $i \in \{0, \ldots, t\}$ of occurrences of a_d in each word. There are $\binom{t}{i}$ ways to distribute these occurrences in each word. Then there are $f_{d-1}(t-i)$ ways to create an abelian pair over $\{a_1, \ldots, a_{d-1}\}$ for the remaining $t - i$ symbols in each word. Setting $k = t - i$ and summing over the choice of k yields

$$f_d(t) = \sum_{k=0}^{t} \binom{t}{k}^2 f_{d-1}(k). \tag{3}$$

Using this formula, $f_d(t)$ can be obtained by starting with $f_1(0) = \cdots = f_1(t) = 1$ and computing $f_i(0), \ldots, f_i(t)$ in turn for $i = 2, \ldots, d$ (Fig. 1 left). The cost of computing the values of f_i given the previously computed values of f_{i-1} is $O(1+2+\cdots+t) = O(t^2)$. Thus the complexity of evaluating $f_d(t)$ using (3) is $O(t^2 d)$, a huge improvement over (2) in most cases. However, eq. (3) is still impractical for the quantum computing application to be described later, for which t is typically small but d is exponentially large. This motivates the development of a formula for $f_d(t)$ whose cost scales with t rather than d.

2.2. An alternative recursive formula

In this section I derive an alternative to (3) whose cost of evaluation is only $O(t^2 \min(d, t))$. Let A_d denote an alphabet of d symbols. The number of abelian squares $(x, y) \in A_d^t \times A_d^t$ can be expressed as the sum of the number of anagrams of each word x:

$$f_d(t) = \sum_{x \in A_d^t} \binom{t}{m_1, \dots, m_d}. \tag{4}$$

Here m denotes the signature of $x = (x_1, \dots, x_t)$. We split off the sum over x_t:

$$f_d(t) = \sum_{x' \in A_d^{t-1}} \sum_{x_t \in A_d} \binom{t}{m'_1, \dots, m'_{x_t} + 1, \dots m'_d} \tag{5}$$

where m' is the signature of $x' \equiv (x_1, \dots, x_{t-1})$. We have

$$\binom{t}{m'_1, \dots, m'_{x_t} + 1, \dots m'_d} = \frac{t}{m'_{x_t} + 1} \binom{t-1}{m'_1, \dots, m'_d}. \tag{6}$$

Then

$$f_d(t) = \sum_{x' \in A_d^{t-1}} \sum_{x_t \in A_d} \frac{t}{m'_{x_t} + 1} \binom{t-1}{m'_1, \dots, m'_d}. \tag{7}$$

By symmetry x_t can be replaced by any value; choosing d yields

$$f_d(t) = d \sum_{x' \in A_d^{t-1}} \frac{t}{m'_d + 1} \binom{t-1}{m'_1, \dots, m'_d}. \tag{8}$$

Now, each x' with a given signature contributes the same value to the sum. We may thus replace the sum over x' by a sum over the signatures of x', weighted by the number of occurrences of each signature:

$$f_d(t) = d \sum_{m'_1 + \dots + m'_d = t-1} \frac{t}{m'_d + 1} \binom{t-1}{m'_1, \dots, m'_d}^2. \tag{9}$$

We henceforth suppress the primes on m. The goal now is to move the dependence on m_d out of the sum, leaving something which has the form of (2). We have

$$\binom{t-1}{m_1, \dots, m_d} = \binom{t-1}{m_d} \binom{t-1-m_d}{m_1, \dots, m_{d-1}}. \tag{10}$$

This yields

$$f_d(t) = d \sum_{m_1+\cdots+m_d=t-1} \frac{t}{m_d+1} \binom{t-1}{m_d}^2 \binom{t-1-m_d}{m_1,\ldots,m_{d-1}}^2. \tag{11}$$

$$= d \sum_{m_d=0}^{t-1} \frac{t}{m_d+1} \binom{t-1}{m_d}^2 \sum_{m_1+\cdots+m_{d-1}=t-1-m_d} \binom{t-1-m_d}{m_1,\ldots,m_d}^2. \tag{12}$$

In terms of $k \equiv t-1-m_d$,

$$f_d(t) = d \sum_{k=0}^{t-1} \frac{t}{t-k} \binom{t-1}{t-1-k}^2 \sum_{m_1+\cdots+m_{d-1}=k} \binom{k}{m_1,\ldots,m_d}^2. \tag{13}$$

Comparison of the latter sum to (2) reveals that it is none other than $f_{d-1}(k)$. The remaining quantities can be simplified as follows:

$$\binom{t-1}{t-1-k} = \binom{t-1}{k}, \tag{14}$$

$$\frac{t}{t-k}\binom{t-1}{t-1-k} = \binom{t}{k}. \tag{15}$$

Making these substitutions yields the main result:

$$f_d(t) = d \sum_{k=0}^{t-1} \binom{t}{k}\binom{t-1}{k} f_{d-1}(k). \tag{16}$$

Note the close similarity between (16) and (3). The crucial difference is that in (16) the sum goes up to only $t-1$; that is, each level of recursion decreases *both* t and d (Fig. 1 right). Thus only $\min(t,d)$ levels of recursion are needed. The cost of this algorithm is $O(t^2 \min(d,t))$.

Eq. (16) can be interpreted in terms of the following approach to constructing an abelian pair: There are d choices for the first symbol a of x. Let $k \in \{0,\ldots,t-1\}$ be the number of occurrences in each word of symbols from A_d/a. There are $\binom{t-1}{k}$ choices to place those other symbols in x and $\binom{t}{k}$ places to place those other symbols in y. Then, one creates an abelian pair of size (k,k) over A_d/a, which is an alphabet of size $d-1$.

Fig. 3 (left) shows $f_d(t)$ as a function of t for exponentially increasing values of d. (The lines for $d \geq 64$ are truncated due to the largest results being

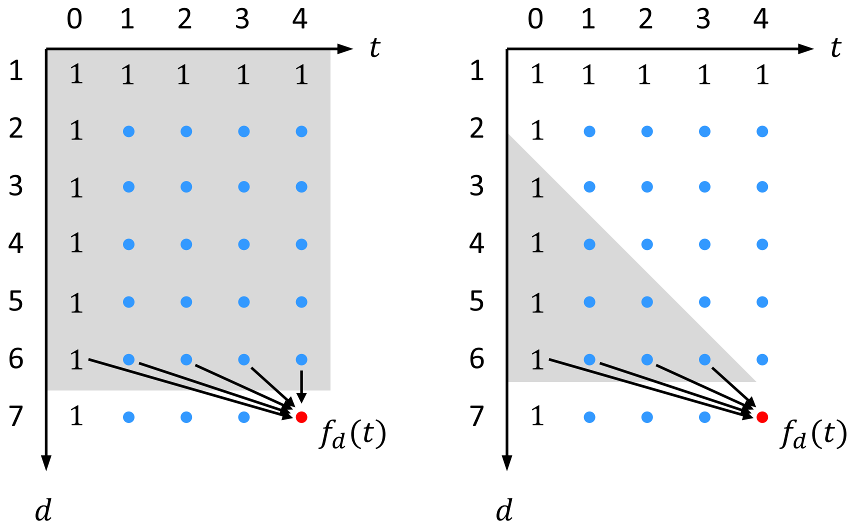

Figure 1: Computational dependencies for two different recursive formulas for $f_d(t)$, the number of abelian squares. (left) Dependency graph for eq. (3), obtained from [30]. (right) Dependency graph for eq. (16). In each case, the desired quantity $f_d(t)$ is shown as a red dot, arrows show the direct dependencies, and the gray shaded region covers all the quantities that must be calculated to determine $f_d(t)$. The pattern on the left leads to a cost of $O(t^2 d)$, while the pattern on the right leads to a cost of $O(t^2 \min(t, d))$.

outside the range of double-precision arithmetic.) The entire plot, comprising 1000 data points, took less than two seconds to compute in MATLAB on a standard personal computer.

3. Application to a problem in quantum computing

3.1. Parameterized quantum circuits and expressiveness

In this section I present an application of formula (16) to a problem in the field of quantum computing. Quantum computing is an emerging approach to computing that leverages the peculiar laws of quantum physics to process information in new, sometimes powerful ways. In the last few years primitive quantum computing devices have become widely available and catapulted quantum computing into a highly active field of research. In the current era of small, noisy devices, the variational approach to quantum computing has become popular [26, 38, 24]. In the variational approach a conventional (digital) computer adjusts the parameters of a parameterized quantum circuit to optimize some function of its output. This approach can be used for a variety of useful tasks such as calculating properties of molecules and materials [28, 21, 17, 34, 25, 22, 20, 12, 6, 11, 37, 5], discrete optimization [10, 27], and machine learning [13, 36, 31, 1, 18], as well as linear algebra [35] and differential equations [23].

A key property of a parameterized quantum circuit is its expressiveness—the range of outputs that can be obtained by varying the parameters. A circuit that is not expressive enough for the problem at hand will produce inferior solutions. On the other hand, a circuit that is overly expressive may be difficult to optimize [14]. For our purposes, the output of a quantum circuit will be the state of an n-qubit register. (A qubit is a quantum bit.) Such a state can be represented by a unit-length complex vector $\psi \in \mathbb{C}^{2^n}$, with the caveat that the overall complex phase of the state is irrelevant.

One way of quantifying the expressiveness of a parameterized circuit is by its fidelity distribution [33, 29]. Fidelity $F(\psi, \psi') = |\langle \psi, \psi' \rangle|^2$, where $\langle \cdot, \cdot \rangle$ denotes inner product, is a measure of the similarity of two quantum states ψ and ψ'. It ranges from 0 (for completely dissimilar states) to 1 (for identical states). Let $\psi(\theta)$ denote the quantum state produced by a quantum circuit as a function of the parameter vector θ. Suppose parameter values are drawn at random. If the circuit is highly expressive, i.e. capable of producing a wide range of states, most of the resulting states will be dissimilar to each other and will have small mutual fidelity. Conversely, if the circuit is inexpressive, i.e. capable of producing only a narrow range of states, most of the produced states will be similar to each other and have large mutual fidelity. Thus the expected value of $F(\psi(\theta), \psi(\theta'))$, where θ, θ' are independent random parameter values, quantifies the circuit's expressiveness: the lower the expected value, the more expressive the circuit. As it turns out, this metric is not very sensitive. A more discerning metric is

$$\mathbb{E}\left[F(\psi(\theta), \psi(\theta'))^t\right] \tag{17}$$

where $t > 1$; typically t is a small positive integer. As t increases, $\mathbb{E}\left[F^t\right]$ becomes less sensitive to the states that are far apart. Thus small values of t measure the expressiveness at a coarse scale in the quantum state space, while large values of t measure the expressiveness at a fine scale.

3.2. Commutative quantum circuits and abelian squares

Commutative quantum circuits (also known as Instantaneous Quantum Polynomial circuits [32]) are a class of relatively simple parameterized quantum

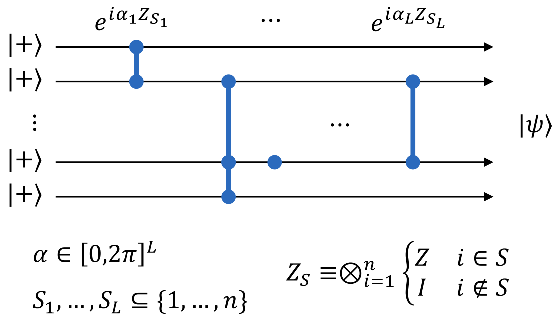

Figure 2: The structure of a commutative quantum circuit (CQC) in the Hadamard basis. Each circuit operation is a multiqubit Z-type rotation on a distinct subset of qubits. A maximal CQC on n qubits consists of all $2^n - 1$ Z-type rotations that act non-trivially on at least one qubit.

circuits whose output distributions are hard to simulate using digital computers [2]. These properties make them an interesting case study in the quest to understand when and why quantum computing is more powerful than classical computing. These properties also suggests that commutative quantum circuits may be a useful ansatz for variational quantum algorithms, for example in the field of machine learning [7].

An n qubit commutative quantum circuit (CQC) of length L can be defined as a sequence of L multiqubit X rotations acting on the state $|0\rangle^{\otimes n}$. (Since these operations all commute, their order does not matter.) For our purposes it will be convenient to treat the circuit and its output in the Hadamard basis; in this basis the circuit consists of L multiqubit Z rotations acting on the state $|+\rangle^{\otimes n}$ where $|+\rangle \equiv (|0\rangle + |1\rangle)/\sqrt{2}$ (Fig. 2). The output state is

$$|\psi\rangle = \left(\prod_{j=1}^{L} \exp(\mathrm{i}\alpha_j Z_{S_j}) \right) \sum_{x \in \{0,1\}^n} \frac{1}{\sqrt{2^n}} |x\rangle \tag{18}$$

Where $S_1, \dots, S_L$ are distinct subsets of $\{1, \dots, n\}$ and $Z_S \equiv \bigotimes_{i=1}^n \begin{cases} Z & i \in S \\ I & i \notin S \end{cases}$, with $Z \equiv |0\rangle\langle 0| - |1\rangle\langle 1|$.

Consider a "maximal" circuit consisting of all 2^n Z-type rotations. Then α may be regarded as a vector over all length-n bitstrings, where each bitstring specifies a particular subset of $\{1, \dots, n\}$. A simple derivation shows that

$$|\psi\rangle = \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} e^{\mathrm{i}\theta_x} |x\rangle \tag{19}$$

where $\theta \in \mathbb{R}^{2^n}$ is the Walsh-Hadamard transform of α. It follows that the ability to prescribe all 2^n components of α implies the ability to prescribe all d components of θ. Since the circuit operation corresponding to α_0 imparts an inconsequential global phase to the quantum state, that circuit operation may be omitted and the global phase may be chosen so that $\theta_0 = 0$. The output state may then be represented by a length-2^n complex vector

$$\psi(\theta) = \left(\frac{1}{\sqrt{d}}, \frac{e^{\mathrm{i}\theta_1}}{\sqrt{d}}, \ldots, \frac{e^{\mathrm{i}\theta_{d-1}}}{\sqrt{d}}\right) \tag{20}$$

where $\theta_1, \ldots, \theta_{d-1}$ can be independently varied. (Here $d = 2^n$ and I have switched indices from bitstrings in $\{0,1\}^n$ to corresponding integers in $\{0, \ldots, 2^n - 1\}$.)

While maximal commutative quantum circuits are not practically realizable for large n (the number of operations is $2^n - 1$), they provide an upper bound on the expressiveness that can be achieved by any commutative quantum circuit with a given number of qubits. As I will now show, the expressiveness of a maximal commutative circuit, as measured by $\mathbb{E}\left[F^t\right]$, is proportional to $f_{2^n}(t)$. The fidelity F is the square of the inner product

$$\psi(\theta)^\dagger \psi(\theta') = \frac{1}{d}\sum_{x=0}^{d-1} e^{\mathrm{i}(\theta'_x - \theta_x)}. \tag{21}$$

In terms of $\phi_x \equiv \theta'_x - \theta_x$ we have

$$F(\psi(\theta), \psi(\theta')) = \left|\psi(\theta)^\dagger \psi(\theta')\right|^2 = \frac{1}{d^2} \sum_{x,y=0}^{d-1} e^{\mathrm{i}(\phi_x - \phi_y)}, \tag{22}$$

$$F(\psi(\theta), \psi(\theta'))^t = \frac{1}{d^{2t}} \sum_{x_1,y_1=0}^{d-1} \cdots \sum_{x_t,y_t=0}^{d-1} e^{\mathrm{i}(\phi_{x_1} + \cdots + \phi_{x_t}) - \mathrm{i}(\phi_{y_1} + \cdots \phi_{y_t})}, \tag{23}$$

and

(24)

$$\mathbb{E}\left[F(\psi(\theta),\psi(\theta'))^t\right] = \frac{1}{d^{2t}}\sum_{x_1,y_1=0}^{d-1}\cdots\sum_{x_t,y_t=0}^{d-1}\mathbb{E}\left[e^{\mathrm{i}(\phi_{x_1}+\cdots+\phi_{x_t})-\mathrm{i}(\phi_{y_1}+\cdots\phi_{y_t})}\right].$$

Let us suppose the rotation angles α_i are drawn uniformly and independently from $[0, 2\pi]$. Then each θ_x and θ'_x are independent and uniform over $[0, 2\pi]$, and ϕ_x is also uniform over $[0, 2\pi]$. For each $i \in \{1, \ldots, d-1\}$, let $m_i(x)$ be the number of occurrences of i in $(x_1, \ldots, x_t)$ and let $m_i(y)$ denote the number of occurrences of i in $(y_1, \ldots, y_t)$. Then the summand may be written as

$$\mathbb{E}\left[e^{\mathrm{i}(\phi_{x_1}+\cdots+\phi_{x_t})-\mathrm{i}(\phi_{y_1}+\cdots\phi_{y_t})}\right] = \mathbb{E}\left[\prod_{i=1}^{d-1} e^{\mathrm{i}(m_i(x)-m_i(y))\phi_i}\right] \tag{25}$$

$$= \prod_{i=1}^{d-1}\mathbb{E}\left[e^{\mathrm{i}(m_i(x)-m_i(y))\phi_i}\right] \tag{26}$$

since the ϕ_i's are independent. Now,

$$\mathbb{E}\left[e^{\mathrm{i}(m_i(x)-m_i(y))\phi_i}\right] = \begin{cases} 1 & m_i(x) = m_i(y) \\ 0 & m_i(x) \neq m_i(y) \end{cases}. \tag{27}$$

Thus the only pairs (x, y) that contribute to $\mathbb{E}\left[F^t\right]$ are those for which $m_i(x) = m_i(y)$ for all $i = 1, \ldots, d-1$. For such pairs it also holds that $m_0(x) = m_0(y)$. That is, a term contributes if and only if $x = (x_1, \ldots, x_t)$ is an anagram of $y = (y_1, \ldots, y_t)$, i.e. xy is an abelian square. It follows that

$$\mathbb{E}\left[F^t\right] = \frac{f_{2^n}(t)}{4^{nt}}. \tag{28}$$

Whereas t is typically small, $d = 2^n$ can be very large, which necessitates use of eq. (16).

It is convenient to compare $\mathbb{E}\left[F^t\right]$ for a given circuit to its minimal value

$$\mathbb{E}\left[F^t\right]_{\min} = \binom{t+d-1}{t}^{-1} \tag{29}$$

which is achieved by a circuit that covers the entire state space uniformly. Fig. 3(right) plots the normalized expressiveness $\mathbb{E}\left[F^t\right]_{\min}/\mathbb{E}\left[F^t\right]$. For all

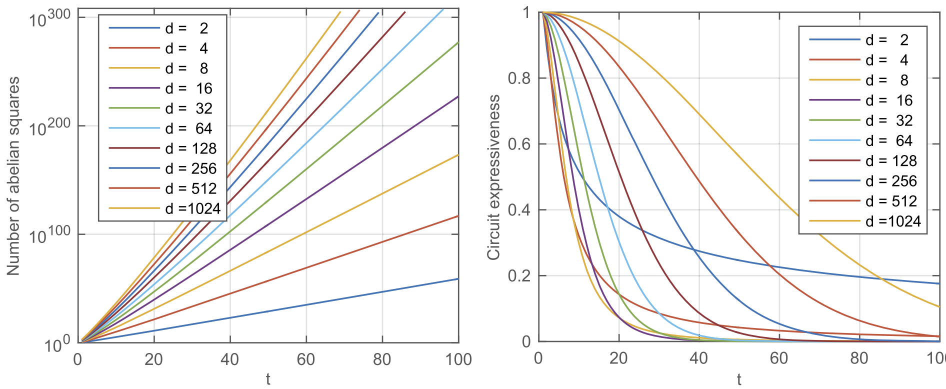

Figure 3: (left) Number $f_d(t)$ of abelian squares of length $t + t$ over an alphabet of size d. (right) Normalized expressiveness $\mathbb{E}\left[F^t\right]_{\min} / \mathbb{E}\left[F^t\right]$ of maximal commutative quantum circuits, as a function of the number of qubits n and the resolving power t.

d, the normalized expressiveness is near 1 at small t and decays to 0 at large t. This indicates that the circuits are highly expressive at coarse scales (small t), but have very low expressiveness at fine scales (large t). That is, the set of states that can achieved by maxmimal commutative quantum circuits span the breadth of the state space, but constitute only a sparse or low-dimensional subset of the state space.

Acknowledgements

This work was performed at Oak Ridge National Laboratory, operated by UT-Battelle, LLC under contract DE-AC05-00OR22725 for the US Department of Energy (DOE). Support for the work came from the DOE Advanced Scientific Computing Research (ASCR) Accelerated Research in Quantum Computing Program under field work proposal ERKJ354.

References

[1] Beckey, J. L., Cerezo, M., Sone, A. and Coles, P. J. (2022). Variational Quantum Algorithm for Estimating the Quantum Fisher Information. *Phys. Rev. Research* **4** 013083. MR4360188

[2] Bremner, M. J., Jozsa, R. and Shepherd, D. J. (2011). Classical Simulation of Commuting Quantum Computations Implies Collapse of the Polynomial Hierarchy. *Proc. R. Soc. A* **467** 459–472. MR2748102

[3] Carpi, A. (1998). On the Number of Abelian Square-Free Words on Four Letters. *Discrete Applied Mathematics* **81** 155–167. MR1492008

[4] Cassaigne, J., Richomme, G., Saari, K. and Zamboni, L. Q. (2011). Avoiding Abelian Powers in Binary Words with Bounded Abelian Complexity. *Int. J. Found. Comput. Sci.* **22** 905–920. MR2806895

[5] Chowdhury, A. N., Low, G. H. and Wiebe, N. (2020). A Variational Quantum Algorithm for Preparing Quantum Gibbs States. *arXiv:2002.00055 [quant-ph].*

[6] Cîrstoiu, C., Holmes, Z., Iosue, J., Cincio, L., Coles, P. J. and Sornborger, A. (2020). Variational Fast Forwarding for Quantum Simulation beyond the Coherence Time. *npj Quantum Inf* **6** 1–10.

[7] Coyle, B., Mills, D., Danos, V. and Kashefi, E. (2020). The Born Supremacy: Quantum Advantage and Training of an Ising Born Machine. *npj Quantum Inf* **6** 1–11.

[8] Crochemore, M., Iliopoulos, C. S., Kociumaka, T., Kubica, M., Pachocki, J., Radoszewski, J., Rytter, W., Tyczyński, W. and Waleń, T. (2013). A Note on Efficient Computation of All Abelian Periods in a String. *Information Processing Letters* **113** 74–77. MR3003705

[9] Erdős, P. (1957). Some Unsolved Problems. *Michigan Math. J.* **4** 291–300. MR0098702 MR0098702

[10] Farhi, E., Goldstone, J. and Gutmann, S. (2014). A Quantum Approximate Optimization Algorithm. *arXiv:1411.4028 [quant-ph].*

[11] Gard, B. T., Zhu, L., Barron, G. S., Mayhall, N. J., Economou, S. E. and Barnes, E. (2020). Efficient Symmetry-Preserving State Preparation Circuits for the Variational Quantum Eigensolver Algorithm. *npj Quantum Inf* **6** 10.

[12] Grimsley, H. R., Economou, S. E., Barnes, E. and Mayhall, N. J. (2019). An Adaptive Variational Algorithm for Exact Molecular Simulations on a Quantum Computer. *Nat Commun* **10** 3007.

[13] Havlíček, V., Córcoles, A. D., Temme, K., Harrow, A. W., Kandala, A., Chow, J. M. and Gambetta, J. M. (2019). Supervised Learning with Quantum-Enhanced Feature Spaces. *Nature* **567** 209–212.

[14] Holmes, Z., Sharma, K., Cerezo, M. and Coles, P. J. (2022). Connecting Ansatz Expressibility to Gradient Magnitudes and Barren Plateaus. *PRX Quantum* **3** 010313.

[15] Huova, M., Karhumäki, J. and Saarela, A. (2012). Problems in between Words and Abelian Words: K-Abelian Avoidability. *Theoretical Computer Science* **454** 172–177. MR2966632

[16] Iliopoulos, C. S., Moore, D. and Smyth, W. F. (1997). A Characterization of the Squares in a Fibonacci String. *Theoretical Computer Science* **172** 281–291. MR1432868

[17] Kandala, A., Mezzacapo, A., Temme, K., Takita, M., Brink, M., Chow, J. M. and Gambetta, J. M. (2017). Hardware-Efficient Variational Quantum Eigensolver for Small Molecules and Quantum Magnets. *Nature* **549** 242–246.

[18] Kardashin, A., Uvarov, A. and Biamonte, J. (2021). Quantum Machine Learning Tensor Network States. *Front. Phys.* **8**.

[19] Keränen, V. (1992). Abelian Squares Are Avoidable on 4 Letters. In *Automata, Languages and Programming* (W. Kuich, ed.). *Lecture Notes in Computer Science* 41–52. Springer, Berlin, Heidelberg. MR1250625

[20] Kokail, C., Maier, C., van Bijnen, R., Brydges, T., Joshi, M. K., Jurcevic, P., Muschik, C. A., Silvi, P., Blatt, R., Roos, C. F. and Zoller, P. (2019). Self-Verifying Variational Quantum Simulation of Lattice Models. *Nature* **569** 355–360.

[21] Li, Y. and Benjamin, S. C. (2017). Efficient Variational Quantum Simulator Incorporating Active Error Minimization. *Phys. Rev. X* **7** 021050.

[22] Liu, J.-G., Mao, L., Zhang, P. and Wang, L. (2021). Solving Quantum Statistical Mechanics with Variational Autoregressive Networks and Quantum Circuits. *Mach. Learn.: Sci. Technol.* **2** 025011.

[23] Lubasch, M., Joo, J., Moinier, P., Kiffner, M. and Jaksch, D. (2020). Variational Quantum Algorithms for Nonlinear Problems. *Phys. Rev. A* **101** 010301.

[24] Magann, A. B., Arenz, C., Grace, M. D., Ho, T.-S., Kosut, R. L., McClean, J. R., Rabitz, H. A. and Sarovar, M. (2021). From Pulses to Circuits and Back Again: A Quantum Optimal Control Perspective on Variational Quantum Algorithms. *PRX Quantum* **2** 010101.

[25] McArdle, S., Jones, T., Endo, S., Li, Y., Benjamin, S. C. and Yuan, X. (2019). Variational Ansatz-Based Quantum Simulation of Imaginary Time Evolution. *npj Quantum Information* **5** 75–81.

[26] McClean, J. R., Romero, J., Babbush, R. and Aspuru-Guzik, A. (2016). The Theory of Variational Hybrid Quantum-Classical Algorithms. *New J. Phys.* **18** 023023.

[27] Moll, N., Barkoutsos, P., Bishop, L. S., Chow, J. M., Cross, A., Egger, D. J., Filipp, S., Fuhrer, A., Gambetta, J. M., Ganzhorn, M., Kandala, A., Mezzacapo, A., Müller, P., Riess, W., Salis, G., Smolin, J., Tavernelli, I. and Temme, K. (2018). Quantum Optimization Using Variational Algorithms on Near-Term Quantum Devices. *Quantum Sci. Technol.* **3** 030503.

[28] Peruzzo, A., McClean, J., Shadbolt, P., Yung, M.-H., Zhou, X.-Q., Love, P. J., Aspuru-Guzik, A. and O'Brien, J. L. (2014). A Variational Eigenvalue Solver on a Photonic Quantum Processor. *Nat Commun* **5** 4213.

[29] Rasmussen, S. E., Loft, N. J. S., Bækkegaard, T., Kues, M. and Zinner, N. T. (2020). Reducing the Amount of Single-Qubit Rotations in VQE and Related Algorithms. *Advanced Quantum Technologies* **3** 2000063.

[30] Richmond, L. B. and Shallit, J. (2009). Counting Abelian Squares. *Electron. J. Combin.* **16** R72. MR2515749

[31] Schuld, M., Sweke, R. and Meyer, J. J. (2021). Effect of Data Encoding on the Expressive Power of Variational Quantum-Machine-Learning Models. *Phys. Rev. A* **103** 032430. MR4243943

[32] Shepherd, D. and Bremner, M. J. (2009). Instantaneous Quantum Computation. *Proc. R. Soc. A* **465** 1413–1439.

[33] Sim, S., Johnson, P. D. and Aspuru-Guzik, A. (2019). Expressibility and Entangling Capability of Parameterized Quantum Circuits for Hybrid Quantum-Classical Algorithms. *Advanced Quantum Technologies* **2** 1900070.

[34] Verdon, G., Marks, J., Nanda, S., Leichenauer, S. and Hidary, J. (2019). Quantum Hamiltonian-Based Models and the Variational Quantum Thermalizer Algorithm. *arXiv:1910.02071 [quant-ph].*

[35] Wang, X., Song, Z. and Wang, Y. (2021). Variational Quantum Singular Value Decomposition. *Quantum* **5** 483.

[36] Wiebe, N. and Wossnig, L. (2019). Generative Training of Quantum Boltzmann Machines with Hidden Units. *arXiv:1905.09902 [quant-ph].*

[37] Xu, L., Lee, J. T. and Freericks, J. K. (2020). Test of the Unitary Coupled-Cluster Variational Quantum Eigensolver for a Simple Strongly Correlated Condensed-Matter System. *Mod. Phys. Lett. B* **34** 2040049. MR4126956

[38] Yuan, X., Endo, S., Zhao, Q., Li, Y. and Benjamin, S. C. (2019). Theory of Variational Quantum Simulation. *Quantum* **3** 191.

Ryan Bennink
Oak Ridge National Laboratory
Oak Ridge, Tennessee 37831
U.S.A.
E-mail address: benninkrs@ornl.gov

Received September 16, 2021