Received: Added at production | Revised: Added at production | Accepted: Added at production

DOI: xxx/xxxx

ARTICLE TYPE

Experiments Testing the Commutativity of Finite-Dimensional Algebras with a Quantum Adiabatic Algorithm

Summary

Determining whether a given algebra is commutative or not is important in the study of these algebraic objects in general and in the classification of semifields in particular. The best classical (i.e. non-quantum) algorithm for this task has a running time which is or order $O(n^3)$, where n is the dimension of the algebra. To reduce this cost, in this paper we study an approach to test the commutativity of a finite dimensional algebra using quantum adiabatic computing. Previous quantum algorithms solving the same problem were based on Grover's quantum search. The algorithm is built from a quantum oracle for the multiplication constants of the algebra. Results of the experiments carried out on a quantum computer simulator, based on two different annealing schedules, are presented, showing that a quantum adiabatic algorithm for the problem can determine the commutativity of finite-dimensional algebras with one-side bounded error with a running time of order $O(\sqrt{n^3})$, achieving a quadratic speed-up over the classical case.

KEYWORDS:

Quantum adiabatic algorithms, Quantum computing, Commutativity, Finite dimensional algebras, Quantum oracles

1 | INTRODUCTION

The paradigm of quantum computing [1,2,3,4], which is based on a direct manipulation of atomic-scaled states of matter, has been theoretically shown to outperform classic computations in some cases. For instance, Grover [5] and Shor [6] algorithms, just to name the best well-known quantum algorithms, perform respectively quadratically and exponentially better than any other classic algorithm. Apart from them, other quantum algorithms exist [7], but it is seems like there are not so many new alternatives that outperform classical techniques [8]. That is the reason why it has been suggested that broading the scope of problems in which the known quantum algorithms can be applied is an important research direction [9].

In this sense, quantum algorithms for the study of problems in algebraic structures have been considered [10,11]. One of them is our recent study of commutativity of finite-dimensional algebras over an arbitrary field [12]. It is based on the model of quantum gates, that resembles the microdesign of classical algorithms from logical gates [13]. Our algorithm was theoretically proved to be quadratically better than any classical algorithm and optimal among the quantum ones (when using the query model). In practice, however, no large scale experiments have been made, as the current state of technology of general purpose quantum computers is rather limited [14], especially in the number of qubits of the available computers.

Apart from the standard design of quantum algorithms from quantum gates, other approaches exist. Among them, one of the most popular is that of adiabatic algorithms, which are based on an evolution of the quantum states by a controlled series of Hamiltonians [15,16]. So far, This model has practically scaled better, with some quantum machines coping with up to some hundreds of qubits at the same time [17]. In this context, we have developed a quantum algorithm for the study of commutativity of finite dimensional algebras based on adiabatic techniques. The construction of the required Hamiltonian is based on the multiplication constant oracle introduced in our previous work and we are able to recover all the good properties of that previous algorithm in this new setting.

The outline of this paper is as follows. Preliminaries on finite-dimensional algebras and the adiabatic quantum model are collected in sections 2 and 3. Section 4 is devoted to present a general method to construct a Hamiltonian from a quantum oracle, and to introduce our adiabatic algorithm based on this methodology. Finally, experiments carried out on a adiabatic quantum simulator and conclusions can be found in section 5.

2 | FINITE-DIMENSIONAL ALGEBRAS OVER A FIELD

In this paper K will be a field (it can be either finite, i.e., a Galois field $\mathbb{F}_q$ [18], or infinite such as the real or complex number fields), and A will be a $K-$algebra, i.e., a $K-$vector space equipped with a bilinear product $\cdot$ [19]. The algebra is commutative or associative if the multiplication satisfies the commutative or associative properties:

$$a \cdot b = b \cdot a \ , \ \forall a, b \in A \qquad \text{or} \qquad (a \cdot b) \cdot c = a \cdot (b \cdot c) \ , \ \forall a, b, c \in A$$

If the underlying $K-$vector space is finite dimensional the $K-$algebra is called finite-dimensional, and it is known as unital when the product has an identity element. Examples of unital algebras include division algebras (when any nonzero element has left and right multiplicative inverses), matrix rings over the field K, Lie and Jordan algebras (i.e., $K-$algebras satisfying Lie or Jordan identities [20]) and finite semifields (i.e., $\mathbb{F}_q-$finite dimensional division algebras [21]).

If A is a $n-$dimensional $K-$algebra ($n \in \mathbb{N}$), and $B = \{x_1, \ldots, x_n\}$ is a K-basis of a A (i.e., $A = K < \{x_1, \ldots, x_n\} >$ and B is K-linearly independent), then there exists a unique set of constants $\{M_{ijk}\}_{i,j,k=1}^{n} \subseteq K$ such that

$$x_i \cdot x_j = \sum_{k=1}^{n} M_{ijk} x_k \ , \ \forall i, j \in \{1, \ldots, n\}$$

This set of multiplication constants that completely determines the product in A is called multiplication table corresponding to A with respect to the basis B. It can be straightforwardly shown that A is a commutative algebra if and only if $M_{ijk} = M_{jik}$, for all $1 \leq i, j, k \leq n$.

A quantum algorithm for deciding the commutativity of a finite dimensional $K-$algebra A with basis $B = \{x_1, \ldots, x_n\}$ was introduced in [12]. In that paper, the multiplication constants of the algebra A with respect to B are query-modeled input data. These multiplication constants (being elements of the field K) were given by an $l-$bit representation (when $K = \mathbb{F}_q$ is a finite field with q elements, then l can be taken as $\lceil \log_2 q \rceil$, whereas when K is an infinite field an standard numerical representation must be used).

For technical reasons, the algebra A was embedded in an algebra $\hat{A}$ with holds the same commutative character as A but whose dimension is slightly larger. In the present paper we proceed in a similar way, but we benefit from the need of more conservative dimensions than in our previous work. As in the previous paper, an oracle made from the oracle $O^{\hat{A}}$ will be used in the quantum computations. Then, here is an analogue of [12, Lemma 1] that allows us to consider dimensions that are always a power of 2.

Lemma 1. For any $n \in \mathbb{N}$, take $m \in \mathbb{N}$ such that $n \leq 2^m$. Then, $\hat{n} = 2^m$, and the $\hat{n}-$dimensional algebra $K-$algebra $\hat{A} = A \times K^{\hat{n}-n}$ with the product given by the rule $(a, \lambda) \cdot (b, \mu) = (ab, 0)$, is commutative if and only A is. In particular, $\hat{n} = \Theta(n)$, and a query oracle $O^{\hat{A}}$ for the multiplication constants of $\hat{A}$ can be made from a query to the oracle O^A for the multiplication constants of A, at most.

Proof. Obviously, the only way $\hat{A}$ can be non-commutative is if there exist a and b in A such that $a \cdot b \neq b \cdot a$ in the original A, for on the new, added elements the multiplication is always 0. What is more, to recover a multiplication constant we only need to check whether it corresponds to elements from A (in which case we consult O^A exactly once) or not (and then we return 0). What is more, it is clear that this operation can be implemented in a quantum circuit (i.e. with a unitary transformation) provided we have a quantum circuit for O^A.

□

In this context we model the oracle $O^{\hat{A}}$ in such a way that $3m$ index register qubits provide the encoding of the triple ijk while the multiplication constant $\hat{M}_{ijk}$ is added to the l oracle qubits. From it, we build the problem oracle, that models the binary function $f(i, j, k) = 1 - \delta_{\hat{M}_{ijk}, \hat{M}_{jik}}$ that marks with a one those multiplication constants M_{ijk} which are different from the corresponding M_{jik}, i.e., those constants witnessing the noncommutativity of A. As such an oracle is different when K has only two elements or it has more than two, as we will show in detail in Section 4.

The main theorem of[12] is the following:

Theorem 1. There is a quantum decision algorithm on the gate model requiring $\Theta(\sqrt{n^3})$ queries to the oracle O^A such that, on output NO certifies the noncommutativity of A with certainty, while on output YES the probability of A being noncommutative is a constant strictly smaller than 1. Moreover, it is query-optimal among the quantum algorithms, in the sense that any other algorithm with bounded error probability for testing the commutativity of the finite dimensional $K-$algebra A uses $\Omega(\sqrt{n^3})$ queries.

Our objective in this paper is to construct a quantum adiabatic algorithm that recovers this quadratic speed-up over the classical case while retaining the one-sided bounded error.

3 | ADIABATIC QUANTUM ALGORITHMS

Adiabatic quantum computation is a polynomially equivalent model to the standard gate model of quantum computation[15,22] that has been applied, for instance, to the simulation of quantum systems[23], to solving NP-complete problems[24] and to deep learning[25]. Its theoretical fundation is the *Quantum Adiabatic Theorem*[26] that states, roughly speaking, that if a quantum system is prepared in an initial ground state of a Hamiltonian (i.e. a state of minimal energy or, equivalently, an eigenvector associated to a minimal eigenvalue of the Hamiltonian) the system is driven by a sequence of slightly changing Hamiltonians then, with high probability, the final state will be also in the ground state of the last Hamiltonian[27].

The idea is, then, to code the information of the problem to solve by means of a Hamiltonian such that its ground states are the solutions to the problem. Usually, getting a system in a ground state of that Hamiltonian is not a simple task, so instead we prepare a quantum system in a easy to get initial state and let it slowly evolve according to a time-dependent Hamiltonian that eventually reaches the desired, final Hamiltonian whose ground states are the solutions to the problem.

Technically, let us assume a quantum system evolves according to the Schrödinger equation

$$i\hbar\frac{d}{dt}|\psi_t\rangle = H(t)|\psi_t\rangle$$

in a time range $0 \leq t \leq T$. If $t(s)$ is a strictly increasing function from 0 to T as $0 \leq s \leq 1$, then let us denote $\bar{H}(s) = H(t(s))$ and $|\bar{\psi}_s\rangle = |\psi_{t(s)}\rangle$. The Schrödinger equation can now be written as $\frac{d}{ds}|\bar{\psi}_s\rangle = -i\tau(s)\bar{H}(s)|\bar{\psi}_s\rangle$, for a certain function $\tau(s)$. Finally, if $E_0(s)$ and $E_1(s)$ are the first and second eigenvalues of the Hamiltonian $\bar{H}(s)$, then let us denote the minimum spectral gap as $\Delta = \min_{0\leq s\leq 1}[E_1(s) - E_0(s)]$.

Theorem 2 (Adiabatic). [28] Let $|\psi_0\rangle$ be the ground state of the Hamiltonian $H(0)$. If $\Delta > 0$ and $\tau(s) >> \frac{\|\frac{d}{ds}\bar{H}(s)\|}{\Delta^2}$, where $\|\cdot\|$ is the matrix norm induced by the L_2 metric, then with high probability $|\psi_T\rangle$ is the ground state of $H(T)$.

In practice, the computation procedure is implemented as follows:

1. An objective real valued function f on m variables which is to be minimized is considered
2. An initial Hamiltonian H_I is chosen, so that its ground state is easy to simulate in a register Q_I of $m' \geq m$ qubits (some extra ancillary qubits might be necessary).
3. A final Hamiltonian H_F is built from f so that the ground state of Q_F is an eigenstate of H_F having minimum eigenvalue, i.e., it encodes a solution to the minimization problem of f.
4. A changing Hamiltonian performing a gradual transition from H_I to H_F is created by the rule:
$$H(t) = (1 - s(t))H_I + s(t)H_F$$
for all $t \leq T$, and a certain function $s(t)$ that increases from 0 to 1.
5. By choosing T sufficiently large the evolution of the initial state Q_I by that series of Hamiltonians can be brought arbitrarily close to Q_F, so that a final appropriate measurement provides the ground state of Q_F with high probability.

It is clear that the computational complexity of an adiabatic quantum algorithm is largely determined by the minimum spectral gap that separates the ground state and the next excited state. The problem of determining the complexity of a quantum adiabatic process is, in general, hard, as stated in [29]:

> Determining this gap in the limit of large problem size is currently an important open problem in adiabatic quantum computing [...]. It is thus not possible to determine the computational complexity of adiabatic quantum algorithms in general, nor, consequently, of the specific adiabatic quantum algorithms presented in this paper.

Fortunately, for our particular problem we will be able to use some existing results about this gap, as we will mention in Section 5.

Thus, in this paper, we apply the general framework of adiabatic quantum computation to our specific problem. The objective function to be minimized is f and for that purpose we create an appropriate final Hamiltonian from the oracle O^f introduced above. How to achieve this in general will be explained in the next section.

4 | FROM QUANTUM ORACLES TO HAMILTONIANS

In this section we state, for completeness, a general procedure to derive, from a given quantum oracle, a Hamiltonian that encodes, as its ground state, the solution to an optimization problem.

Lemma 2. Let O be a quantum oracle that evaluates a binary function g on m qubits, i.e.,

$$O(|x\rangle|y\rangle) = |x\rangle|y \oplus g(x)\rangle \ , \ \forall |x\rangle \in (\mathbb{Z}/2\mathbb{Z})^m, \forall |y\rangle \in \mathbb{Z}/2\mathbb{Z}$$

If M is the set of $x \in (\mathbb{Z}/2\mathbb{Z})^m$ such that $g(x) = 1$, then both O and $\frac{Id+O}{2}$ are Hamiltonians whose ground state is generated by

$$\left\{ |x\rangle \left(\frac{|0\rangle - |1\rangle}{2} \right) \mid x \in M \right\}$$

The corresponding eigenvalues are -1 and 0, respectively.

Proof. It is enough to prove the lemma for O, as the second case can be straightforwardly deduced from it. First, notice that $O^2 = Id$ and that $O^\dagger = O^{-1}$ because O is a unitary operator. Then, $O^\dagger = O$, the operator O is Hermitian and therefore it is a Hamiltonian.

Also, for all $x \in (\mathbb{Z}/2\mathbb{Z})^m$, we have

$$O\left(|x\rangle \left(\frac{|0\rangle + |1\rangle}{2}\right)\right) = |x\rangle \left(\frac{|0 \oplus g(x)\rangle + |1 \oplus g(x)\rangle}{2}\right) = |x\rangle \left(\frac{|0\rangle + |1\rangle}{2}\right)$$

and

$$O\left(|x\rangle \left(\frac{|0\rangle - |1\rangle}{2}\right)\right) = |x\rangle \left(\frac{|0 \oplus g(x)\rangle - |1 \oplus g(x)\rangle}{2}\right) = (-1)^{g(x)}|x\rangle \left(\frac{|0\rangle - |1\rangle}{2}\right)$$

This means that $\mathbb{C}^{\otimes(m+1)} = E(1) \oplus E(-1)$, where $E(\lambda)$ are subspaces of eigenvectors with eigenvalue $\lambda \in \{-1, 1\}$:

$$E(1) = \left\langle \left\{ |x\rangle \left(\frac{|0\rangle + |1\rangle}{2}\right) \ , \ x \in (\mathbb{Z}/2\mathbb{Z})^m \right\} \cup \left\{ |x\rangle \left(\frac{|0\rangle - |1\rangle}{2}\right) \ , \ x \in (\mathbb{Z}/2\mathbb{Z})^m \text{ s.t. } g(x) = 0 \right\} \right\rangle$$

$$E(-1) = \left\langle \left\{ |x\rangle \left(\frac{|0\rangle - |1\rangle}{2}\right) \ , \ x \in (\mathbb{Z}/2\mathbb{Z})^m \text{ s.t. } g(x) = 1 \right\} \right\rangle$$

□

Notice that this lemma is useful for it allows to transform the problem of searching states marked by an oracle from the quantum circuit model to the quantum adiabatic computational paradigm.

In our case, for the problem of determining the commutativity of finite-dimensional algebras, we suppose we are given an oracle to access the multiplication constantes of the algebra. This oracle allows us to obtain (a quantum state representing) M_{ijk} given (quantum states representing) i, j and k as presented in Figure 1. Notice that we use l qubits to represent each constant, with l a number that will only depend on the size of the underlying field K (if K is infinite, an approximate representation must be adopted).

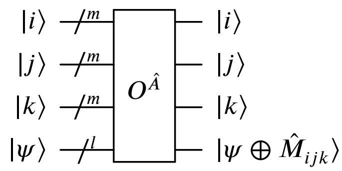

FIGURE 1 Oracle for the multiplication constants of the algebra

From this circuit, we can then construct an oracle that will output 0 on those triples (i, j, j) for which $M_{ijk} = M_{jik}$ and 1 on those triples (i, j, j) for which $M_{ijk} \neq M_{jik}$; that is, it marks those indices that are witnesses of the non-commutativity of the algebra. If the underlying field is $\mathbb{Z}/2\mathbb{Z}$, the construction of this new oracle is straightforward: we just need to consult the constants M_{ijk} and M_{jik} and compute their sum modulo 2. This can be accomplished with two consults to the oracle for the multiplicative constants and two swap operations (see Figure 2).

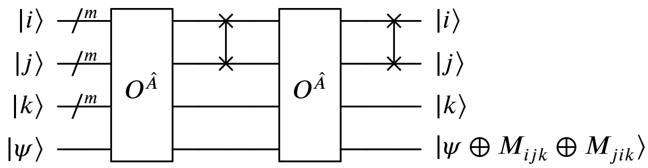

FIGURE 2 Problem oracle for fields with characteristic 2

For the general case, when K is not $\mathbb{Z}/2\mathbb{Z}$ and more than one qubit is necessary to represent the underlying field, the construction is a bit more complex (see Figure 3) for we have to use ancilla qubits and we need to uncompute the values on those qubits to return their state to $|0\rangle$.

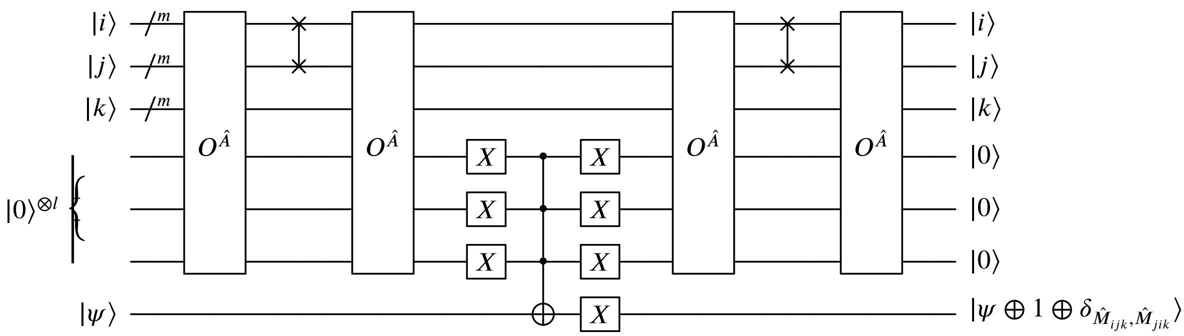

FIGURE 3 Problem oracle for fields with characteristic different from 2

Notice, however, that the number of consults to the original multiplication oracle is, in both cases, a constant that does not depend on the dimension n of the algebra under study (namely, it is 2 if $l = 1$ and 4 if $l > 1$).

We will use this construction in the next section in order to transform our original search problem into a quantum adiabatic optimisation problem.

5 | EXPERIMENTS AND RESULTS

We have carried out experiments based on our previous theoretical considerations to determine if concrete $\mathbb{Z}/2\mathbb{Z}-$finite dimensional algebras are commutative or not. Assuming that we have been given a multiplication constant oracle O^A, we construct the corresponding final Hamiltonian $H_F = O^f$ on $3m$ qubits using the results of sections 2 and 4 ($m = \log_2 n$). Let us notice that, just like in [12], a standard phase-oracle technique allows to remove the last ancillary qubit in the construction of the oracle O^f. Our choice for the initial Hamiltonian was

$$H_I = Id - |\psi_0\rangle\langle\psi_0|$$

where

$$|\psi_0\rangle = \frac{1}{\sqrt{2^{3m}}} \sum_{i=0}^{2^{3m}-1} |i\rangle$$

Notice that its ground state contains the initial state that was taken: $|\psi_0\rangle$ with eigenvalue equal to 0.

In all the cases, the quantum adiabatic algorithm is used to search for a triple (i, j, k) such that $M_{ijk} \neq M_{jik}$. If such a triple is found, then we declare that the algebra is not commutative. In other case, we state that algebra is commutative. Explicitly, the algorithm is as follows:

Algorithm 1: Quantum adiabatic algorithm to test the commutativity

---

Set the state of the computer to $|\psi_0\rangle$, the ground state of the initial Hamilton H_I
Apply the Hamiltonian $H(t) = (1 - s(t))H_I + s(t)H_F$ for times $t \in [0, T]$
Measure the system to obtain (i, j, k)
If $M_{ijk} \neq M_{jik}$, return NO
Else, return YES

Notice that the only source of error of the algorithm comes from an answer in which we declare that the algebra is commutative, for we only say that is non-commutative we have found an explicit violation of the commutativity constrains.

The practical details of the experiment are as follows. We have chosen dimensions $n = 2, 4, 8$ and 16. Also, we have considered both global and local annealing schedules for the evolution of the Hamiltonian. In the first case, the linear function $s(t) = t$ was chosen, where as in the second case the function $s(t)$ was chosen according to [30], that is, s verifying:

$$t = \frac{1}{2\epsilon} \frac{N}{\sqrt{N-1}} \Big(\arctan\left(\sqrt{N-1}(2s-1)\right) + \arctan\left(\sqrt{N-1}\right) \Big)$$

where

$$N = 2^{3m}$$

and ϵ verifies

$$T = \frac{\pi}{2\epsilon}\sqrt{N}$$

The experiments were carried out in a standard computer using the Qutip python package [31,32].

In the first experiment, we compare the behavior of global vs. local evolution for algebras of dimensions $n = 2, 4, 8$ and 16. In these cases we assume that only one pair of multiplication constants (out of the n^3 possible multiplication constants) witnesses the noncommutativity of the algebra. The probability of the adiabatic quantum algorithm finding one of those witnesses against the number of iterations made is plotted in Figures 4 through 7.

As stated in [30], global schedule yields a computation time of linear order, so there is no advantage of the method compared to a classical search. However, when the local schedule is considered, we have a quadratic speed-up compared to classical search (so the algorithm can be viewed as the adiabatic evolution version of the Grover's algorithm presented in [12]). Notice also that in this case the success probability exhibits an oscillatory pattern, as noticed in [33,34].

In the second experiment, we take a noncommutative algebra of dimension $n = 8$ with different numbers of multiplication constants witnessing its noncommutativity. Namely, we consider sets of 56, 112, 168 and 224 nonmatching pairs of multiplication constants out of the $\frac{(2^8)^3-(2^8)^2}{2} = 224$ possible noncommutative witness pairs. The probability of finding one of those constants,

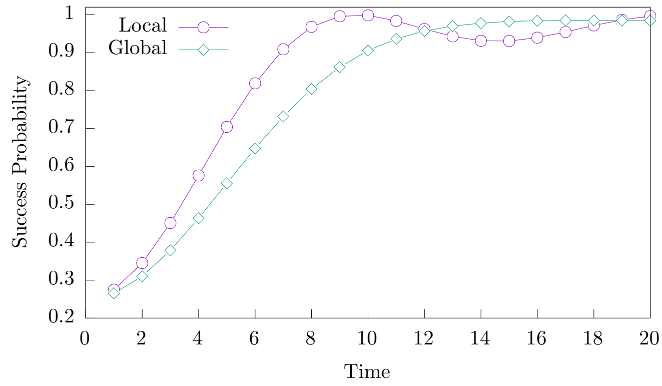

FIGURE 4 Probability of detecting the unique pair of noncommutative witnesses for $\mathbb{Z}/2\mathbb{Z}-$algebras of dimension $n = 2$

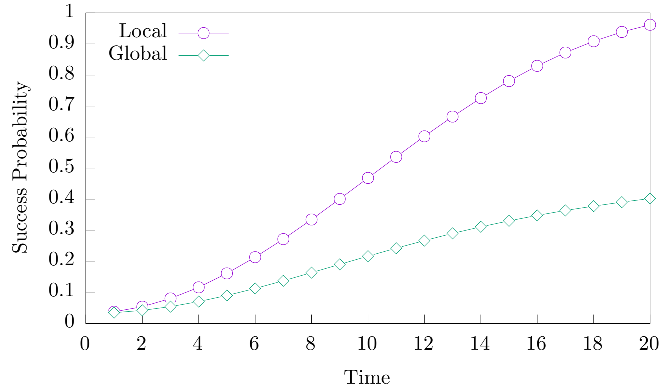

FIGURE 5 Probability of detecting the unique pair of noncommutative witnesses for $\mathbb{Z}/2\mathbb{Z}-$algebras of dimension $n = 4$

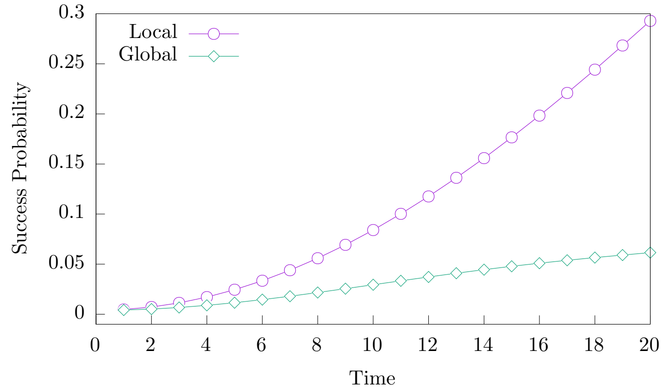

FIGURE 6 Probability of detecting the unique pair of noncommutative witnesses for $\mathbb{Z}/2\mathbb{Z}-$algebras of dimension $n = 8$

when local search is carried out, is presented in Figure 8. Observe, once again, the oscillatory pattern in the success probability (even though it is predicted to converge to one as the number of iterations increases).

Finally, we have studied the number of iterations required to find a unique noncommutative witness pair of multiplication constants with probabilities at least $\frac{1}{4}$, $\frac{1}{2}$, $\frac{3}{4}$, $\frac{9}{10}$ and $\frac{99}{100}$ (for algebras of dimensions $n = 2, 4$ and 8 with global and local schedules and also $n = 16$ with local schedule). As shown in Figures 9 and 10, the local annealing schedule yields better results than

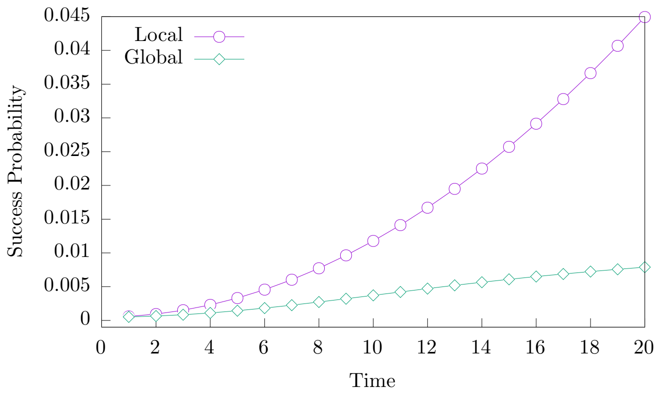

FIGURE 7 Probability of detecting the unique pair of noncommutative witnesses for $\mathbb{Z}/2\mathbb{Z}-$algebras of dimension $n = 16$

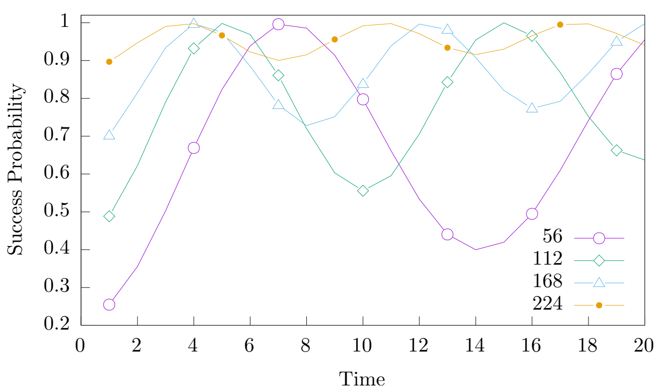

FIGURE 8 Probability of detecting a noncommutative witness pair (out of the $56, 112, 168$ or 224) for a $\mathbb{Z}/2\mathbb{Z}-$algebra of dimension $n = 8$

the global schedule: in the first case the number of iterations increases with a factor of $\sqrt{2^3}$, whereas in the second case the increasing factor is 2^3.

6 | CONCLUSIONS AND FUTURE WORK

In this paper we have presented experiments testing the commutativity of finite-dimensional algebras with a Quantum Adiabatic Algorithm. This type of quantum model is a (polynomially equivalent) alternative to the standard quantum gate model that was considered for the same problem in [12]. Experiments were carried out with both global and local annealing schedules and it has been shown that in the later case Grover's order of complexity is recovered. This is technically relevant because the adiabatic quantum model has developed larger real models in practice so performance on a quantum computer with more number of qubits (and so, dealing with algebras of bigger dimensions) without sacrificing the theoretical complexity seems feasible. Also, the new algorithm retains the property of having one side error: we always determine with certainty when the algebra under study is non-commutative and the only possible error is when we declare it to be commutative.

In future works, we plan to explore the possibilities of applying quantum computing approaches (both with the quantum circuit model and with quantum adiabatic algorithms) to other problems related to the classification of finite-dimensional algebras in general and of semifields in particular, including, for instance, determining when an algebra is associative and finding all the matrices over the underlying field K that represent a semifield for a given dimension.

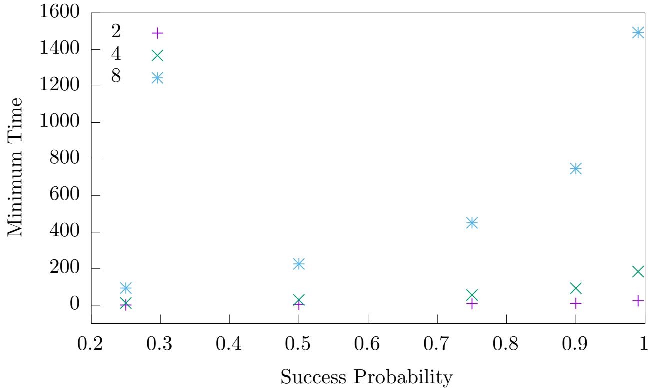

FIGURE 9 Number of iterations required to achieve a given probability of detecting the unique pair of noncommutative witnesses for $\mathbb{Z}/2\mathbb{Z}-$algebras of dimensions $n = 2, 4, 8$ with global schedule

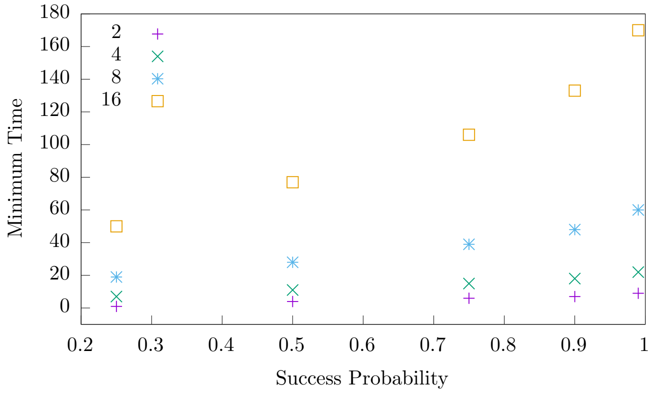

FIGURE 10 Number of iterations required to achieve a given probability of detecting the unique pair of noncommutative witnesses for $\mathbb{Z}/2\mathbb{Z}-$algebras of dimensions $n = 2, 4, 8, 16$ with local schedule

ACKNOWLEDGEMENTS

This work has been partially supported by TEC2015-67387-C4-3-R and MTM2017-83506-C2-2-P.

References

1. Feynman R. Simulating physics with computers. *International Journal of Theoretical Physics* 1982; 21 (6): 467-488.
2. Manin Y. Vychislimoe i nevychislimoe. *Sov. Radio* 1980: 13-15.
3. Benioff P. The computer as a physical system: a microscopic quantum mechanical Hamiltonian model of computers as represented by Turing machines. *Journal of statistical physics* 1980; 22 (5): 563-591.
4. Deutsch D. Quantum theory, the Church-Turing principle and the universal quantum computer. *Proc. Roy. Soc. London Ser. A* 1985; 400(1818): 97–117.
5. Grover LK. A Fast Quantum Mechanical Algorithm for Database Search. In: STOC '96. ACM; 1996; New York, NY, USA: 212–219.
6. Shor PW. Algorithms for Quantum Computation: Discrete Logarithms and Factoring. In: SFCS '94. IEEE Computer Society; 1994; Washington, DC, USA: 124–134

7. Jordan S. Quantum Algorithm Zoo. http://math.nist.gov/quantum/zoo/; 2011-18.

8. Shor P. Why Haven't More Quantum Algorithms Been Found?. *Journal of the ACM* 2003; 50 (1): 87-90.

9. Montanaro A. Quantum algorithms: an overview. *Npj Quantum Information* 2016; 2: 15023.

10. Pak I. Testing commutativity of a group and the power of randomization. *LMS Journal of Computation and Mathematics* 2012; 15: 38-43.

11. Magniez F, Nayak A. Quantum complexity of testing group commutativity. *Algorithmica* 2007; 48(3): 221-232.

12. Combarro EF, Ranilla J, Rúa IF. A quantum algorithm for the commutativity of finite dimensional algebras. Submitted 2018.

13. Nielsen MA, Chuang IL. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press . 2011.

14. IBM . IBM Quantum Experience. https://www.research.ibm.com/ibm-q/; .

15. Farhi E, Goldstone J, Gutmann S, Sipser M. *Quantum computation by adiabatic evolution*. arXiv:quant-ph/0001106v1 . 2000.

16. Farhi E, Goldstone J, Gutmann S, Lapan J, Lundgren A, Preda D. A quantum adiabatic evolution algorithm applied to random instances of an NP-Complete problem. *Science* 2001; 292 (5516): 472-475.

17. D-Wave . D-Wave Systems. https://www.dwavesys.com; .

18. Lidl R, Niederreiter H. *Finite Fields*. Encyclopedia of Mathematics and its Applications (20)Addison-Wesley . 1983.

19. Lang S. *Algebra*. Addison-Wesley . 1965.

20. Schafer RD. An Introduction to Nonassociative Algebras. *Pure and Applied Mathematics* 2011; 22.

21. Knuth DE. Finite semifields and projective planes. *Journal of Algebra* 1965; 2: 182-217.

22. Aharonov D, Dam vW, Kempe J, Landau Z, Lloyd S, Regev O. Adiabatic Quantum Computation is Equivalent to Standard Quantum Computation. *SIAM J. Comput.* 2007; 37(1): 166–194.

23. Babbush R, Love PJ, Aspuru-Guzik A. Adiabatic Quantum Simulation of Quantum Chemistry. *Scientific Reports* 2014; 4: 6603 EP -.

24. Wang H, Wu LA. Ultrafast adiabatic quantum algorithm for the NP-complete exact cover problem. *Scientific Reports* 2016; 6: 22307 EP -.

25. Liu J, Spedalieri FM, Yao KT, et al. Adiabatic Quantum Computation Applied to Deep Learning Networks. *Entropy* 2018; 20(5): 380. doi: 10.3390/e20050380

26. Born M, Fock V. Beweis des Adiabatensatzes. *Zeitschrift für Physik* 1928; 51 (3-4): 165-180.

27. McGeoch CC. *Adiabatic Quantum Computation and Quantum Annealing*. Morgan&Claypool Publishers: Synthesis Lectures on Quantum Computing . 2014.

28. Messiah A. *Quantum mechanics. Vol. I*. Translated from the French by G. M. TemmerNorth-Holland Publishing Co., Amsterdam; Interscience Publishers Inc., New York . 1961.

29. Clark L, Gaitan F. Graph Isomorphism and Adiabatic Quantum Computing. *Physical Review A* 2014; 89 (2): 22342-20.

30. Roland J, Cerf NJ. Quantum Search by Local Adiabatic Evolution. *Physical Reviews A* 2002; 65 (4): 042308.

31. Johansson JR, Nation PD, Nori F. QuTiP 2: A Python framework for the dynamics of open quantum systems. *Comp Phys. Comm* 2013; 184: 1234-1240.

32. Johansson JR, Nation PD, Nori F. QuTiP: An open-source Python framework for the dynamics of open quantum systems. *Comp Phys. Comm* 2012; 183: 1760-1772.

33. Dalzell AM, Theodore JY, Chuang IL. Fixed-Point Adiabatic Quantum Search. *Physical Review A* 2017; 95(1).

34. Li FG, Bao WS, Wang X, et al. Success Rate Versus Finite Run Time in Local Adiabatic Quantum Search Algorithm. *Commun. Theor. Phys* 2017; 67: 355-359.