

**Quantum independent-set problem and non-Abelian adiabatic mixing**Biao Wu (吴飙),<sup>1,2,3,\*</sup> Hongye Yu (余泓焯),<sup>1</sup> and Frank Wilczek<sup>4,5,2,6,7</sup><sup>1</sup>*International Center for Quantum Materials, School of Physics, Peking University, Beijing 100871, China*<sup>2</sup>*Wilczek Quantum Center, School of Physics and Astronomy, Shanghai Jiao Tong University, Shanghai 200240, China*<sup>3</sup>*Collaborative Innovation Center of Quantum Matter, Beijing 100871, China*<sup>4</sup>*Center for Theoretical Physics, MIT, Cambridge, Massachusetts 02139, USA*<sup>5</sup>*T. D. Lee Institute, Shanghai Jiao Tong University, Shanghai 200240, China*<sup>6</sup>*Department of Physics, Stockholm University, Stockholm SE-106 91, Sweden*<sup>7</sup>*Department of Physics and Origins Project, Arizona State University, Tempe, Arizona 25287, USA*

(Received 3 January 2019; revised manuscript received 1 July 2019; published 13 January 2020)

We present an efficient quantum algorithm for independent-set problems in graph theory, based on non-Abelian adiabatic mixing. We illustrate the performance of our algorithm with analysis and numerical calculations for two different types of graphs, with the number of edges proportional to the number of vertices or its square. Our quantum algorithm is compared to the corresponding quantum circuit algorithms and classical algorithms. Non-Abelian adiabatic mixing can be a general technique to aid exploration in a landscape of near-degenerate ground states.

DOI: [10.1103/PhysRevA.101.012318](https://doi.org/10.1103/PhysRevA.101.012318)**I. INTRODUCTION**

The supremacy of quantum computers over classical computers is illustrated by many significant algorithms, in particular the Shor algorithm [1] for factorization and the Grover algorithm [2] for search. These algorithms are based on discrete operations orchestrating simple quantum gates. Algorithms of this kind are called quantum circuit algorithms [3].

In another paradigm of quantum computing, algorithms are implemented through the design of Hamiltonians. Here one starts with an easy-to-prepare initial state, allows it evolve dynamically, and at some point makes appropriate measurements. (Of course, the Hamiltonians should correspond to potentially realizable circuits.) Hamiltonian-based quantum algorithms translate programming problems into physical problems, which allow one to exploit familiar physical processes to optimize algorithms. A Hamiltonian approach to quantum search was proposed in 1998 [4] and soon extended to more general “adiabatic” algorithms [5].

It has been shown that every quantum circuit algorithm can be converted into a quantum adiabatic algorithm, whose time complexity is polynomially equivalent (and vice versa) [6,7]. But the continuum approach can suggest different methods, such as the non-Abelian mixing discussed here, or resonance, as we will describe elsewhere [8].

Here we present an efficient quantum Hamiltonian algorithm for the independent-set problem (see Fig. 1). Any graph has trivial independent sets: the empty set and sets with just one vertex. Our aim is to find non-trivial independent sets, with two or, ideally, many more vertices. Independent-set problems can be rephrased in terms of all-negated 2-satisfiability (2-SAT) problems, and vice versa. Based on this

observation, we are able to construct a Hamiltonian such that its ground states are all independent sets of a given graph. We then prepare the Hamiltonian system in one of its trivial ground states and evolve it adiabatically along a closed path. This leads to non-Abelian adiabatic mixing in the sub-Hilbert space of degenerate ground states [9] and generates a quantum state that is roughly an equal-probability superposition of all ground states. As the number of non-trivial solutions is much bigger (see Figs. 3 and 5), when we make a measurement in the end, we will likely find a nontrivial solution. Numerical results indicate that we are almost certain to find a nontrivial independent set. We analyze the performance of our algorithm for two different types of graphs: those with the number of edges proportional to the number of vertices or to its square. While finding solutions to this particular problem is not a pressing issue, our technique brings in some physics which is interesting in itself, and which might find more general applications.

**II. EQUIVALENCE TO 2-SAT**

For a given graph, we can assign a Boolean variable to each of its vertices (see Fig. 1):  $x_j = 1$  when the  $j$ th vertex is chosen for an independent set and  $x_j = 0$  when it is not. When two vertices  $x_i$  and  $x_j$  are connected by an edge, it means that  $x_i$  and  $x_j$  cannot be simultaneously chosen for one independent set. This is equivalent to imposing the following two-variable clause

$$(\neg x_i \vee \neg x_j) \quad (i \neq j). \quad (1)$$

Therefore, finding an independent set of a graph  $n$  vertices with  $m$  edges is equivalent to finding a solution to a 2-SAT problem which has  $n$  variables and whose  $m$  clauses are of the above form. Since the clauses involve only negated variables, we call it an all-negated 2-SAT problem. An all-negated

\*wubiao@pku.edu.cn

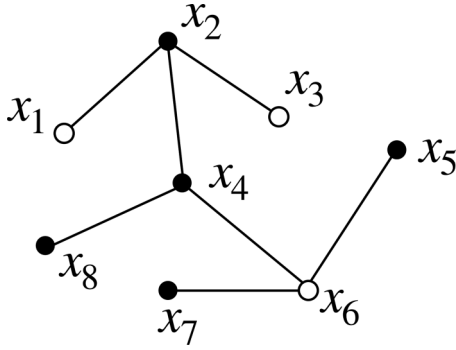


FIG. 1. An independent set of a graph is a set of vertices, no two of which are connected by an edge. Each vertex is assigned a Boolean variable:  $x_j = 1$  if the  $j$ th vertex is included in an independent set and  $x_j = 0$  if not. For example, the empty circles here form an independent set that is described by a set of Boolean numbers  $(1, 0, 1, 0, 0, 1, 0, 0)$ .

2-SAT problem manifestly has at least  $n + 1$  solutions, namely  $(0, 0, 0, \dots, 0)$  and  $n$  assignments that have exactly one variable being 1, such as  $(1, 0, 0, \dots, 0)$  and  $(0, 1, 0, \dots, 0)$ . They correspond to the trivial independent sets: the empty set and sets with only one vertex. We are interested in finding nontrivial solutions, that is, the solutions with at least two 1's. There are generic algorithms of time complexity  $O(m + n)$  to find solutions for 2-SAT problems [10,11]. However, these algorithms may well find the trivial solutions. We need different algorithms to find nontrivial solutions.

### III. QUANTUM ALGORITHM

For a clause in Eq. (1), the function  $f(x_i, x_j) = x_i x_j$  is minimized when the clause is satisfied. Notice that  $x_j = (\hat{\sigma}_j^z + 1)/2$ , where the third Pauli matrix  $\sigma_j^z \in \{1, -1\}$  in the natural basis. With these two observations, for a given graph (or a 2-SAT problem), we construct the following Hamiltonian [12]:

$$H_0 = \Delta \sum_{\langle ij \rangle} (\hat{\sigma}_i^z + \hat{\sigma}_j^z + \hat{\sigma}_i^z \hat{\sigma}_j^z), \quad (2)$$

where the summation  $\langle ij \rangle$  is over all edges (or clauses). All the independent sets are the ground states of  $H_0$  and vice versa. The energy gap between the ground states and the first excited states is  $4\Delta$ .

We rotate spin  $\hat{\sigma}_j^z$  to an arbitrary direction  $\vec{r} = \{\sin \theta \cos \varphi, \sin \theta \sin \varphi, \cos \theta\}$ , and obtain new spin operator  $\hat{\tau}_j = V_j \hat{\sigma}_j^z V_j^{-1}$  with

$$V_j = \begin{pmatrix} \cos \frac{\theta}{2} & e^{-i\varphi} \sin \frac{\theta}{2} \\ e^{i\varphi} \sin \frac{\theta}{2} & -\cos \frac{\theta}{2} \end{pmatrix} = V_j^{-1}. \quad (3)$$

If  $|u\rangle_j$  and  $|d\rangle_j$  are eigenstates of  $\hat{\sigma}_j^z$ , that is,  $\hat{\sigma}_j^z |u\rangle_j = |u\rangle_j$  and  $\hat{\sigma}_j^z |d\rangle_j = -|d\rangle_j$ , the eigenstates of  $\hat{\tau}_j$  are

$$|u_{\vec{r}}\rangle_j = \cos \frac{\theta}{2} |u\rangle_j + \sin \frac{\theta}{2} e^{i\varphi} |d\rangle_j, \quad (4)$$

$$|d_{\vec{r}}\rangle_j = \sin \frac{\theta}{2} |u\rangle_j - \cos \frac{\theta}{2} e^{i\varphi} |d\rangle_j. \quad (5)$$

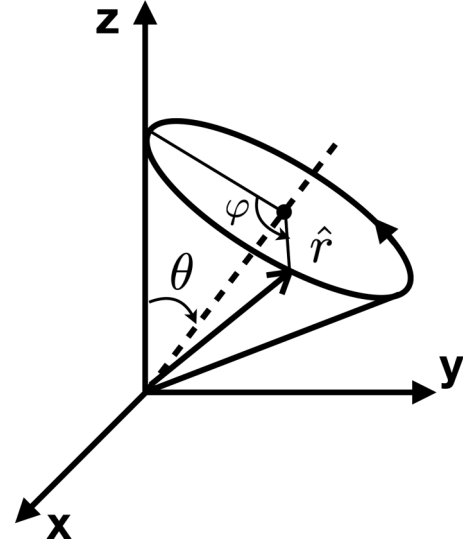


FIG. 2. Adiabatic path in the algorithm.  $\theta$  is the angle between the rotating axis and the  $z$  axis and  $\varphi$  is the angle rotated from the initial direction. Note that  $\theta$  and  $\varphi$  here are related to but different from  $\bar{\theta}$  and  $\bar{\varphi}$  in Eq. (3).

With  $U = V_1 \otimes V_2 \otimes \dots \otimes V_n$ , we can rotate all the spins to the same direction and construct a new Hamiltonian,

$$H_{\vec{r}} = U H_0 U^{-1} = \Delta \sum_{\langle ij \rangle} (\hat{\tau}_i + \hat{\tau}_j + \hat{\tau}_i \hat{\tau}_j). \quad (6)$$

It is clear that  $H_{\vec{r}}$  has the same set of eigenvalues as  $H_0$ . The eigenstates of  $H_{\vec{r}}$  can be obtained by rotating the ones of  $H_0$ , and have the following form:

$$\begin{aligned} |E_{\alpha}\rangle &= |u_{\vec{r}}\rangle_1 \otimes |d_{\vec{r}}\rangle_2 \otimes \dots \otimes |u_{\vec{r}}\rangle_j \otimes \dots \otimes |u_{\vec{r}}\rangle_n \\ &= |u_{\vec{r}}, d_{\vec{r}}, \dots, u_{\vec{r}}, \dots, u_{\vec{r}}\rangle. \end{aligned} \quad (7)$$

The Hamiltonian  $H_{\vec{r}}$  is parametrized by the direction  $\vec{r}$ . With this in mind we propose the following quantum algorithm for the independent-set problem:

(1) prepare the system at state  $\{-1, -1, \dots, -1\}$ , which corresponds to the empty set  $(0, 0, \dots, 0)$ ;

(2) set  $\vec{r}$  initially along the  $z$  axis and slowly change  $H_{\vec{r}}$  by changing  $\vec{r}$  along a closed path shown in Fig. 2;

(3) make a measurement after  $\vec{r}$  returns to the  $z$  direction.

Note that the energy gap  $4\Delta$  of  $H_{\vec{r}}$  does not change with  $\vec{r}$  and is independent of the system size  $n$ . Therefore, the evolution in the above algorithm can be made adiabatic by changing  $\vec{r}$  at a slow but constant rate. As  $\{-1, -1, \dots, -1\}$  is a ground state of  $H_0$ , when  $\vec{r}$  changes slowly, the system will stay in the sub-Hilbert space spanned by the ground states of  $H_{\vec{r}}$ . This kind of adiabatic evolution in a sub-Hilbert space of degenerate eigenstates was studied in Ref. [9], where it is found that an adiabatic evolution along a closed path is given by

$$W = P \exp i \oint A(t) dt, \quad (8)$$

where  $A$  is the gauge matrix given by  $A_{\alpha,\beta} = i \langle E_{\alpha} | \partial_t | E_{\beta} \rangle$  ( $\partial_t \equiv \partial/\partial t$ ) and  $P$  is a symbol reminding us that  $A(t)$  at different times is to be applied in time order. Note that such

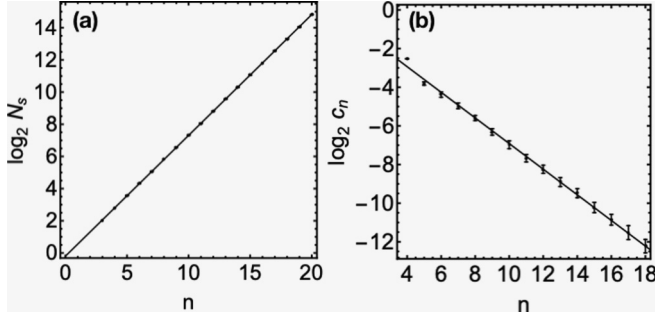


FIG. 3. (a) The number of independent sets of a graph as a function of the number of vertices  $n$  for the case  $m = n$ . The fitting line is given by  $\log_2 N_s = 0.029 + 0.748n$ . The result is averaged over 1000 instances randomly sampled out of all possible configurations of edges; the standard error of every data point is around  $10^{-3}$ . (b) The averaged probability  $c_n$  of the  $n + 1$  trivial solutions in the final state as a function of  $n$ . The fitting line is given by  $\log_2 c_n = -0.444 - 0.654n$ .

an adiabatic evolution of degenerate eigenstates was proposed to construct quantum gates [13].

We find that  $\langle E_\alpha | \partial_t | E_\beta \rangle$  is not zero only when  $|E_\alpha\rangle$  and  $|E_\beta\rangle$  differ by at most one qubit state. When  $\alpha \neq \beta$ , we have

$$A_{\alpha,\beta} = i \langle E_\alpha | \partial_t | E_\beta \rangle = i \langle u_{\bar{r}} | \partial_t | d_{\bar{r}} \rangle = \frac{\sin \theta}{2} \frac{d\varphi}{dt}, \quad (9)$$

where  $\theta$  is the angle between the rotating axis and the  $z$  axis and  $\varphi$  is the rotating angle (see Fig. 2). When  $\alpha = \beta$  and  $|E_\alpha\rangle$  has  $k$  qubits in state  $|u_{\bar{r}}\rangle$  and  $n - k$  qubits in state  $|d_{\bar{r}}\rangle$ , we have

$$\begin{aligned} A_{\alpha,\alpha} &= i \langle E_\alpha | \partial_t | E_\alpha \rangle \\ &= - \left\{ k \sin^2 \frac{\theta}{2} + (n - k) \cos^2 \frac{\theta}{2} \right\} \frac{d\varphi}{dt}. \end{aligned} \quad (10)$$

Letting  $A = \tilde{A} \frac{d\varphi}{dt}$ , we have

$$W = P \exp \int i \tilde{A}(\theta) d\varphi = \exp[2\pi i \tilde{A}(\theta)], \quad (11)$$

where the gauge matrix  $\tilde{A}$  is real and independent of time.

As the gauge matrix  $\tilde{A}$  has many off-diagonal terms, it generates a mixing in the sub-Hilbert space of the ground states, producing a quantum state that is roughly an equal-probability superposition of all the ground states. When a measurement is made at the end of the algorithm, we will likely find a nontrivial ground state since the number of nontrivial solutions is much bigger than that of trivial solutions. To illustrate the efficiency of our algorithm, we consider two typical cases: the number of edges is proportional to (I) the number of vertices or to (II) the square of the number of vertices.

*Case I.* To be specific, we choose  $m = n$ . Let  $N_s(n)$  be the number of all the independent sets of a given graph. Our numerical results in Fig. 3(a) show that  $N_s$  grows exponentially with  $n$ . The fitting gives us  $N_s(n) \approx 1.02 \times 2^{0.748n}$ . This means that the  $n + 1$  trivial sets are only a tiny part of all the independent sets when  $n$  is large.

For our quantum algorithm, for simplicity we choose  $\theta = \pi/2$ , where the gauge matrix  $\tilde{A}$  has the simplest form. We

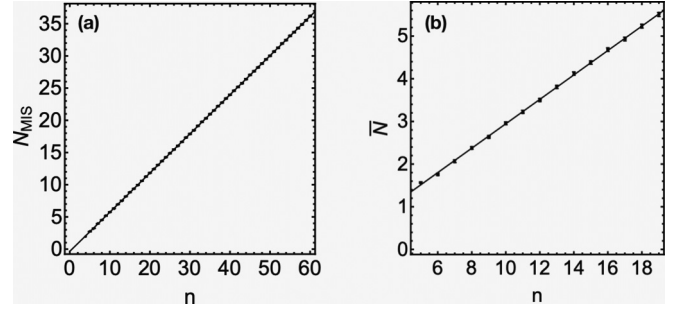


FIG. 4. (a) The averaged number of vertices  $N_{\text{MIS}}$  of maximum independent sets (MISs) of a graph as a function of the number of vertices  $n$  for the case  $m = n$ . The fitting line is given by  $N_{\text{MIS}} = -0.344 + 0.608n$ . The result is averaged over 1600 instances randomly sampled out of all possible configurations of edges; the standard error of every data point is around  $10^{-3}$ . (b) The averaged number  $\bar{N}$  of vertices for all the solutions in the final state as a function of  $n$ . The fitting line is given by  $\bar{N} = 0.08 + 0.286n$ .

numerically compute

$$|\psi_1\rangle = W |\psi_0\rangle, \quad (12)$$

where  $|\psi_0\rangle = \{-1, -1, \dots, -1\}$  is the initial state. Let  $d_n$  be the probability of the  $n + 1$  trivial solutions in the final state  $|\psi_1\rangle$  and  $c_n = d_n/(n + 1)$  be the averaged probability. Our numerical results are plotted in Fig. 3, where we see  $c_n$  decreases exponentially with  $n$ . Numerical fitting indicates  $c_n \approx 0.735 \times 2^{-0.654n}$ . Therefore, we are almost certain to find a nontrivial solution at the end of the algorithm. As the gap  $4\Delta$  is independent of the problem size  $n$ , the time that our adiabatic evolution takes to traverse one loop in Fig. 2 is independent of  $n$ . Thus the time complexity of our quantum algorithm is  $O(1)$ , and for large  $n$  it produces a nontrivial solution with near certainty.

It is interesting to gauge the potential of our algorithm in finding the maximum independent sets (MISs). For this purpose, we define the averaged number of vertices in independent sets in the final quantum state  $|\psi_1\rangle = \sum_j a_j |s_j\rangle$ :

$$\bar{N} = \sum_j |a_j|^2 N_j, \quad (13)$$

where  $N_j$  is the number of 1's in the  $j$ th solution  $|s_j\rangle$ . According to our numerical results, in the case of  $m = n$ , the average size of maximum independent set  $N_{\text{MIS}}$  grows almost linearly with  $n$  [Fig. 4(a)]. The averaged vertex number  $\bar{N}$  also grows linearly with  $n$  but with smaller coefficient. While the maximum independent set is hard to approximate [14], for sparse graphs with average degree  $d = 2m/n$ , the results in Ref. [15] show that  $N_{\text{MIS}} \approx (2 - \epsilon_d) \frac{n \ln d}{d}$ , and the classical greedy algorithm can find independent sets of nearly half size  $(1 + \epsilon'_d) \frac{n \ln d}{d}$  with high probability, where  $\epsilon_d, \epsilon'_d \rightarrow 0$ . However, no efficient algorithm is known to find independent sets of size  $(1 + \epsilon'') \frac{n \ln d}{d}$  for any fixed  $\epsilon'' > 0$ . Our quantum algorithm can on average find an independent set of size  $\bar{N} \approx 0.47 N_{\text{MIS}}$ ; the ratio is slightly less than the greedy algorithm.

*Case II.* We choose specifically  $m = \lfloor n^2/4 \rfloor$ . According to Ref. [16], for such a graph there exists with almost certainty a

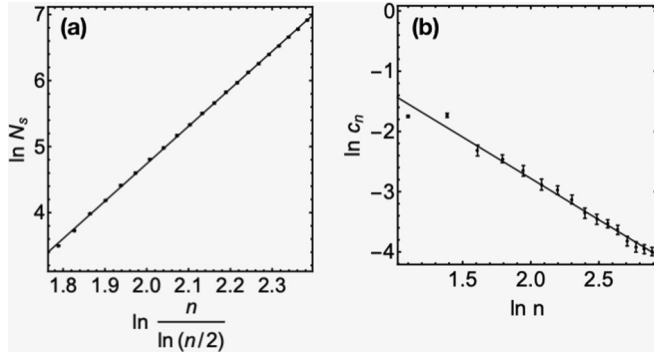


FIG. 5. (a) Number of independent sets for graph  $m = \lfloor n^2/4 \rfloor$  as a function of  $n$ . The fitting line is given by  $\ln N_s = -6.61 + 5.68 \ln \frac{n}{\ln(n/2)}$ . The result is averaged over 1000 instances randomly sampled out of all possible configurations of edges; the standard error of every data point is around  $10^{-2}$ . (b) The averaged probability  $c_n$  of the  $n + 1$  trivial solutions in the final state  $|\psi_1\rangle$  as a function of  $n$ . The fitting line is  $\ln c_n = -0.03 - 1.37 \ln n$ .

maximum independent set of the following size:

$$k = 4 \left( \ln \frac{n}{4 \ln(n/2)} + 1 \right). \quad (14)$$

Since all its subsets are also independent sets, the number of independent sets  $N_s$  is at least  $N_s \gtrsim \mathcal{O}((n/\ln n)^{4 \ln 2})$ . The numerical results in Fig. 5(a) show that

$$N_s \propto \mathcal{O}((n/\ln n)^{5.7}). \quad (15)$$

For this case, we evolve the system along the loop in Fig. 2 with  $\theta = 1.2$  to make all possible ground states more evenly distributed in the final quantum state (see later discussion with Fig. 8). Our numerical results in Fig. 3(b) show that the averaged probability of finding trivial solutions  $c_n \propto 1/n^{1.37}$ .

In this case, we find numerically that the number  $N_{\text{MIS}}$  is proportional to  $\ln \frac{n}{\ln(n/2)}$  while  $\bar{N}$  grows linearly with  $\ln n$  (see Fig. 6). In the classical algorithm, Erdős-Rényi random graphs

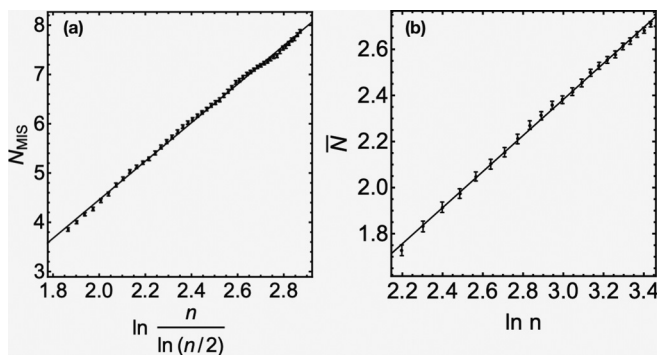


FIG. 6. (a) The averaged number of vertices  $N_{\text{MIS}}$  in the maximum independent sets (MISs) of a graph as a function of the number of vertices  $n$  for the case  $m = \lfloor n^2/4 \rfloor$ . The fitting line is given by  $\ln N_{\text{MIS}} = -3.36 + 3.91 \ln \frac{n}{\ln(n/2)}$ . The result is averaged over 1600 instances randomly sampled out of all possible configurations of edges; the standard error of every data point is around  $10^{-2}$ . (b) The averaged number  $\bar{N}$  of vertices for all the solutions in the final state as a function of  $n$ . The fitting line is given by  $\bar{N} = 0.035 + 0.783 \ln n$ .

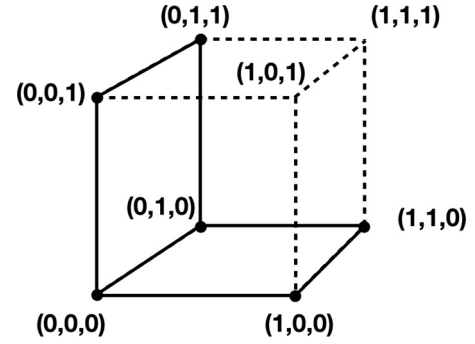


FIG. 7. A median graph embedded in a cube for a graph that has three vertices  $x_1, x_2, x_3$  and one edge connecting  $x_1$  and  $x_3$ . Each point represents an independent set and the solid line connects a pair of independent sets that differ by only one element.

$G(n, 1/2)$ , which behave similarly to our model when  $n$  is large, almost have a MIS of size  $2 \log_2 n(1 + o(1))$ , but it is still an open problem to find in polynomial time an independent set of size  $(1 + \epsilon) \log_2 n$  while a greedy algorithm can reach  $\log_2 n$  [17,18]. Our results are worse than that, but can still on average find an  $\mathcal{O}(\log n)$  size independent set in  $\mathcal{O}(1)$  time.

#### IV. QUANTUM DIFFUSION IN MEDIAN GRAPH

Our algorithm centers on the quantum non-Abelian adiabatic mixing in a sub-Hilbert space of degenerate ground states. We find that such a dynamics process can also be viewed as a quantum diffusion in a median graph which can be embedded in an  $n$ -dimensional cube (see Fig. 7).

As the solutions of an all-negated 2-SAT problem form a median graph [19–21], all the independent sets of a graph form a median graph: each independent set is represented by a point, and a pair of points are connected by a line when the two independent sets differ by only one vertex. This median graph can be embedded in an  $n$ -dimensional cube, as shown in Fig. 7 for  $n = 3$ . Our Hermitian gauge matrix  $\tilde{A}(\theta)$  can be regarded as a Hamiltonian defined on this median graph: the on-site energy is  $\tilde{A}_{\alpha,\alpha}$  while off-diagonal element  $\tilde{A}_{\alpha,\beta}$  gives the hopping amplitude between two points  $\alpha$  and  $\beta$ . If we start with an initial wave function localized at  $(0, 0, \dots, 0)$ , this wave function will spread in the graph and the diffusion process is given by

$$|\psi(t)\rangle = \exp[it\tilde{A}(\theta)] |\psi_0\rangle. \quad (16)$$

When  $t = 2\pi$ , we recover the adiabatic mixing in Eq. (12). So, the adiabatic evolution in Fig. 2 is just a special case of quantum diffusion in a median graph for  $t = 2n_l\pi$  ( $n_l$  is a positive integer).

Let us expand  $|\psi(t)\rangle$  in terms of all the solutions

$$|\psi(t)\rangle = \sum_j a_j(t) |s_j\rangle, \quad (17)$$

where  $|s_j\rangle$  is the  $j$ th solution. To characterize how widely the wave function is diffused over the median graph, we define a

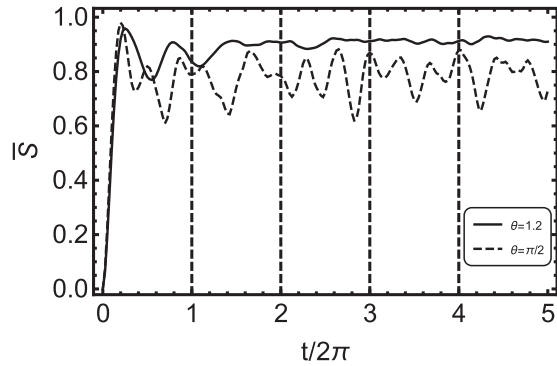


FIG. 8. The time evolution of entropy  $\bar{S}$  for the quantum diffusion in the median graph. The dashed line is for a typical independent set with  $\theta = \pi/2$ ; the solid line is for a different typical independent set with  $\theta = 1.2$ . The averaged or equilibrium value of the entropy is  $S \approx 0.77$  for  $\theta = \pi/2$  and  $S \approx 0.88$  for  $\theta = 1.2$ .  $n = m = 12$ .

quantum entropy

$$S(t) = - \sum_j |a_j(t)|^2 \ln |a_j(t)|^2. \quad (18)$$

It is called generalized Wigner–von Neumann entropy in Ref. [22]. It is clear that the maximum of  $S(t)$  is  $\ln N_s$ . We define  $\bar{S} = S/(\ln N_s)$  and plot  $\bar{S}$  as a function of  $t$  in Fig. 8. We again consider first the special case  $\theta = \pi/2$  (dashed line in Fig. 8). We observe an interesting behavior of  $\bar{S}$ : it starts at zero, quickly rises up to a value very close to 1, and eventually oscillates around an equilibrium value. At  $t = 2\pi, 4\pi, 6\pi, \dots$ , which correspond to adiabatically evolving along the loop in Fig. 2 over one, two, three,  $\dots$  rounds, we have  $\bar{S} \approx 0.75$ . This means that the probability is roughly evenly distributed among all possible solutions. We checked numerically how probability is distributed among different sets of the solutions. For example, if the number of solutions with three 1's is  $N_3$ , then the probability of  $|\psi_1\rangle$  in these solutions is approximately  $N_3/N_s$ .

We can reduce the fluctuations of  $\bar{S}$  and raise its equilibrium value by choosing a different  $\theta$ . In Fig. 8, we have plotted  $\bar{S}$  for  $\theta = 1.2$  (solid line). We see much smaller oscillations around a larger equilibrium value. At  $t = 2\pi, 4\pi, 6\pi, \dots$ , we have  $\bar{S} \sim 0.85$ .

The behavior of  $\bar{S}$  in Fig. 8 resembles how a similar quantum entropy behaves in quantum chaotic systems [22–25]: it rises up rapidly from a low initial value and quickly settles into an equilibrium value. By comparing the two lines in Fig. 8 we see that, when  $\theta$  deviates from the special value  $\pi/2$ ,  $\tilde{A}(\theta)$  tends to be more chaotic.

## V. PERSPECTIVE AND APPLICATIONS

The conventional wisdom is that the time complexity for quantum adiabatic algorithm is solely determined by the energy gap [5]. So far, for most quantum adiabatic algorithms, their energy gaps decrease exponentially with the system size  $n$  [26]. In these cases, the polynomial time spent on preparing the system and checking the final results can be comfortably neglected in counting for time complexity. This appears no

longer to be the case for our quantum algorithm since the running time of our algorithm, which is  $O(1)$  as indicated by the constant energy gap, is much shorter than the preparation time and checking time when  $n$  is large. As a result, one may feel that it is necessary to include the preparation time and checking time in time complexity. As there are  $m$  terms in the Hamiltonian, the preparation should scale with  $m$ . The checking time is clearly proportional to  $m$ . In this sense, our algorithm is of time complexity  $O(m)$ .

There is another way to assess the time complexity of our algorithm: convert it to quantum circuits to see how the number of quantum logic gates scales with  $n$ . The caveat is that, although there is an optimal way to convert a quantum circuit algorithm to a quantum adiabatic algorithm [7], there is no optimal way to convert a quantum adiabatic algorithm to a quantum circuit algorithm. If we adopt the scheme in Ref. [26], the converted quantum circuit algorithm is of time complexity at least  $O(n^3)$ . If the gauge matrix  $A$  is treated as a sparse matrix, the converted quantum circuit algorithm may be of time complexity  $O(n^2)$  [27]. One would expect that the optimally converted quantum circuit algorithm cannot be better than  $O(n)$ .

When the independent-set problem is regarded as a 2-SAT problem, there exist both quantum and classical algorithms. In Ref. [28], Farhi *et al.* proposed a quantum algorithm for a class of restricted quantum 2-SAT problems. This algorithm is applicable to our independent-set problem and the time complexity is  $O(m^4 n^2)$ . This is slower than the generic classical algorithm for 2-SAT problems, which is of time complexity  $O(m + n)$ .

There is another classical algorithm where one simply picks up two variables and sets them to 1. For the graph with  $m = n$ , the chance of this randomly picked solution being wrong is proportional to  $2m/n(n-1) \sim 1/n$ , which decreases polynomially with the graph size  $n$ . In comparison, in our quantum algorithm, the chance of being wrong is exponentially small. For the graph with  $m = \lfloor n^2/4 \rfloor$ , the chance of this randomly picked solution being wrong is about  $n^2/2n(n-1) \sim 1/2$ , which is independent of  $n$ . One may improve this classical algorithm by running it in parallel. For example, for the latter case, if one runs the algorithm in parallel on  $k$  different computers, the chance of success can increase substantially and becomes  $1 - 1/2^k$ . However, in this algorithm, one has to take the checking time into account for time complexity; otherwise, one would not know which computer produces the correct answer. The time complexity, as a result, is at least  $O(km)$ .

The key of our algorithm, adiabatic non-Abelian mixing, can be applied to other problems that have multiple solutions with one or more solutions that are easy to find or already found. For example, a class of quantum 2-SAT problems have multiple solutions and one of their trivial solutions is precisely  $|-1, -1, -1, \dots, -1\rangle$  [28–31].

The maximum independent-set problem for a graph is an NP-hard problem. Our analysis above puts this problem in a different perspective. The maximum independent set corresponds to the point which is farthest from the original point  $(0, 0, \dots, 0)$ . In our algorithm, a quantum particle originally at  $(0, 0, \dots, 0)$  will indeed arrive at this farthest point through quantum diffusion, but with very small probability.

Our understanding of quantum diffusion may help us to find a way to increase this probability significantly.

## VI. CONCLUSION

In sum, we have presented an efficient quantum algorithm for independent-set problems which exploits the non-Abelian adiabatic mixing in a sub-Hilbert space of degenerate eigenstates. We expect our algorithm will have potential applications in many other problems.

## ACKNOWLEDGMENTS

B.W. and H.Y. are supported by the The National Key R&D Program of China (Grants No. 2017YFA0303302 and No. 2018YFA0305602) and National Natural Science Foundation of China (Grant No. 11921005). F.W. is supported by the Swedish Research Council under Contract No. 335-2014-7424. In addition, this work was supported by the US Department of Energy under Contract No. de-sc0012567 and by the European Research Council under Grant 742104.

- 
- [1] P. W. Shor, *SIAM J. Comput.* **26**, 1484 (1997).
  - [2] L. K. Grover, *Phys. Rev. Lett.* **79**, 325 (1997).
  - [3] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, Cambridge, 2000).
  - [4] E. Farhi and S. Gutmann, *Phys. Rev. A* **57**, 2403 (1998).
  - [5] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser, [arXiv:quant-ph/0001106](https://arxiv.org/abs/quant-ph/0001106).
  - [6] D. Aharonov, W. van Dam, J. Kempe, Z. Landau, S. Lloyd, and O. Regev, *SIAM J. Comput.* **37**, 166 (2007).
  - [7] H. Yu, Y. Huang, and B. Wu, *Chin. Phys. Lett.* **35**, 110303 (2018).
  - [8] F. Wilczek, H. Hu, and B. Wu (unpublished).
  - [9] F. Wilczek and A. Zee, *Phys. Rev. Lett.* **52**, 2111 (1984).
  - [10] B. Aspvall, M. F. Plass, and R. E. Tarjan, *Inf. Proc. Lett.* **8**, 121 (1979).
  - [11] S. Even, A. Itai, and A. Shamir, *SIAM J. Comput.* **5**, 691 (1976).
  - [12] E. Farhi, J. Goldstone, S. Gutmann, J. Lapan, A. Lundgren, and D. Preda, *Science* **292**, 472 (2001).
  - [13] P. Zanardi and M. Rasetti, *Phys. Lett. A* **264**, 94 (1999).
  - [14] J. Hastad, *Acta Math.* **182**, 105 (1999).
  - [15] A. Coja-Oghlan and C. Efthymiou, *Random Struct. Algorithms* **47**, 436 (2015).
  - [16] A. M. Frieze, *Discrete Math.* **81**, 171 (1990).
  - [17] C. Banderier, H. kwei Hwang, V. Ravelomanana, and V. Zacharovas (unpublished).
  - [18] G. R. Grimmett and C. J. H. McDiarmid, *Math. Proc. Cambridge Philos. Soc.* **77**, 313 (1975).
  - [19] T. J. Schaefer, in *Proceedings of the Tenth Annual ACM Symposium on the Theory of Computing, STOC '78* (ACM, New York, 1978), pp. 216–226.
  - [20] H.-J. Bandelt and V. Chepoi, *Contemp. Math.* **453**, 49 (2008).
  - [21] <https://en.wikipedia.org/wiki/2-satisfiability>
  - [22] Z. Hu, Z. Wang, and B. Wu, *Phys. Rev. E* **99**, 052117 (2019).
  - [23] J. von Neumann, *Z. Phys.* **57**, 30 (1929).
  - [24] J. von Neumann, *Eur. Phys. J. H* **35**, 201 (2010).
  - [25] X. Han and B. Wu, *Phys. Rev. E* **91**, 062106 (2015).
  - [26] W. van Dam, M. Mosca, and U. Vazirani, in *Proceedings of the 42nd IEEE Symposium on Foundations of Computer Science* (IEEE, Piscataway, NJ, 2001), pp. 279–287.
  - [27] D. W. Berry, A. M. Childs, R. Cleve, R. Kothari, and R. D. Somma, in *Proceedings of the Forty-sixth Annual ACM Symposium on Theory of Computing, STOC '14* (ACM, New York, 2014), pp. 283–292.
  - [28] E. Farhi, S. Kimmel, and K. Temme, [arXiv:1603.06985](https://arxiv.org/abs/1603.06985).
  - [29] S. Bravyi, [arXiv:quant-ph/0602108](https://arxiv.org/abs/quant-ph/0602108).
  - [30] N. de Beaudrap and S. Gharibian, in *31st Conference on Computational Complexity (CCC 2016)*, edited by R. Raz, Leibniz International Proceedings in Informatics (LIPIcs), Vol. 50 (Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 2016), pp. 27:1–27:21.
  - [31] I. Arad, M. Santha, A. Sundaram, and S. Zhang, [arXiv:1508.06340](https://arxiv.org/abs/1508.06340).