Towards Shortest Paths Via Adiabatic Quantum Computing

C. Bauckhage, E. Brito, K. Cvejoski, C. Ojeda, J. Schücker, and R. Sifa
Fraunhofer Center for Machine Learning and Fraunhofer IAIS
St. Augustin, Germany

ABSTRACT

Since first working quantum computers are now available, accelerated developments of this technology may be expected. This will likely impact graph- or network analysis because quantum computers promise fast solutions for many problems in these areas. In this paper, we explore the use of adiabatic quantum computing in finding shortest paths. We devise an Ising energy minimization formulation for this task and discuss how to set up a system of quantum bits to find minimum energy states of the model. In simulation experiments, we numerically solve the corresponding Schrödinger equations and observe our approach to work well. This evidences that shortest path computation can at least be assisted by quantum computers.

KEYWORDS

shortest path finding, adiabatic quantum computing

ACM Reference Format:
C. Bauckhage, E. Brito, K. Cvejoski, C. Ojeda, J. Schücker, and R. Sifa. 2018. Towards Shortest Paths Via Adiabatic Quantum Computing. In *Proceedings of 14th Int. Workshop Mining and Learning with Graphs (MLG'18).* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

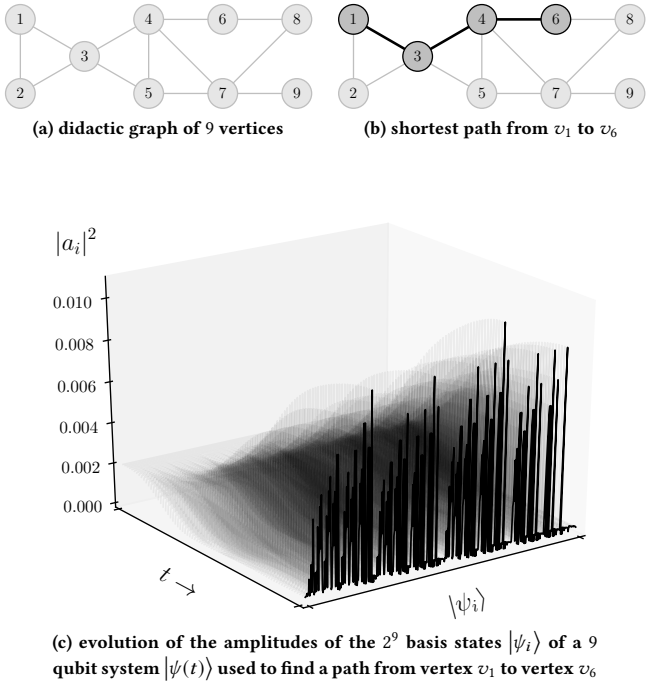
1 INTRODUCTION

Quantum computing exploits quantum mechanical phenomena such as superposition or entanglement for information processing and is now becoming a technical reality. First quantum computers are commercially available [7], governments and industry invest heavily into research and development [6, 8, 13, 14, 21, 24], and a growing number of voices predicts further rapid progress [5, 10]. These developments will likely impact graph analysis because quantum computing promises efficient solutions to many of the search- and optimization problems occurring in this ara. Examples include but are not limited to computing vertex- or clique covers, graph colorings Hamiltonian cycles and paths [18], or graph cuts and clusterings [2, 23].

In this paper, we present first steps towards quantum shortest path computation. For the time being, we consider a simple scenario encountered in route planning and assume that we have access to a matrix of path length distances between vertices of an undirected graph. Incorporating this information, we discuss how adiabatic quantum computing can identify vertices that lie on a shortest path between a source and a target vertex.

MLG'18, August 2018, London, UK

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM.
https://doi.org/10.1145/nnnnnnn.nnnnnnn



(a) didactic graph of 9 vertices **(b) shortest path from v_1 to v_6**

(c) evolution of the amplitudes of the 2^9 basis states $|\psi_i\rangle$ of a 9 qubit system $|\psi(t)\rangle$ used to find a path from vertex v_1 to vertex v_6

Figure 1: Adiabatic quantum computing of shortest paths. (c) visualizes the adiabatic evolution of a system of 9 qubits set up to determine which vertices of the graph in (a) form a shortest path from v_1 to v_6. At time t, the system is in a superposition of $2^9 = 512$ basis states $|\psi_i\rangle$ each of which indexes a possible subset of vertices. At the beginning, it is equally likely to find the system in any of these states. At the end, one basis state has a noticeably higher amplitude $|a_i|^2$ than the others and is thus more likely to be measured; this state is $|101101000\rangle$ and indexes the nodes highlighted in (b).

Our interest in adiabatic quantum computing stems from the fact that devices following this paradigm solve optimization problems that are not entirely alien to researchers in machine learning and data mining. Indeed, devices such as produced by D-Wave Systems [3, 9, 17] are tailored towards finding minimum energy states of Ising models and therefore solve quadratic unconstrained binary optimization problems of the following form

$$\boldsymbol{s}^* = \underset{s \in \{-1,+1\}^n}{\operatorname{argmin}} \sum_{i,j=1}^{n} Q_{ij}\, s_i\, s_j + \sum_{i=1}^{n} q_i\, s_i. \qquad (1)$$

Ising models were originally conceived to represent systems of magnetic dipoles [15, 22]. An intuitive interpretation of (1) therefore is the following: the 2^n vectors $\boldsymbol{s}$ are possible global states of a

system of n entities each of which can be in one of two local states (either $+1$ or -1). The coupling matrix $\boldsymbol{Q} \in \mathbb{R}^{n \times n}$ models internal interactions among the entities and the vector $\boldsymbol{q} \in \mathbb{R}^n$ models external influences.

However, Ising models are of broader use as they can be seen to represent bi-partitioning problems where n entities need to be divided into two disjoint subsets whose elements are labeled $+1$ and -1, respectively. Indeed, many problems are bi-partitioning problems in disguise. For instance, the problem of finding a shortest path in a graph can be understood as the problem of dividing its vertices into those that are *on* and *off* the shortest path.

Yet, Ising models only make sense if we intend to solve them on a quantum computer. For example, for a graph of n vertices there are 2^n possible bi-partitions which generally prevents exhaustive searches for solutions. On an adiabatic quantum computer, however, we can prepare a system of n quantum bits (qubits) such that it is in a quantum mechanical superposition of all 2^n possible partitions and more likely to be measured in a configuration that reflects the desired solution. Hence, once (adiabatic) quantum computers reach quantum supremacy, i.e. once they can harness the phenomena of quantum mechanics to their full potential, they will be able to search more efficiently than digital computers[1].

One strategy for running graph algorithms on adiabatic quantum computers is thus to attempt to (re)formulate them in terms of minimization objectives whose solutions correspond to the minimum energy states of an Ising model.

In section 2, we therefore devise Ising models for the problem outlined above and show how vertices on a shortest path can be identified through Ising energy minimization.

We will distinguish two general cases. In the first case, there is a single shortest path from a source to a target vertex; in the second case, there exist several shortest paths. We shall see that Ising models for the first case are almost trivial whereas the second one requires more elaborate models. For simplicity, our focus in this paper will thus be on the first case and we leave most aspects of the second case to future work.

In order for this paper to be self contained and accessible to a wider audience, we then review quantum computing in general and adiabatic quantum computing in particular. That is, in section 3, we provide a brief and concise overview of basic concepts of quantum computing and introduce the corresponding mathematical notation.

In section 4, we then discuss details as to how to set up a system of qubits as well as corresponding Hamiltonian operators such that our Ising models for shortest path computation can be solved via adiabatic quantum computing.

Finally, in section 5, we present simulation experiments which demonstrate that our adiabatic quantum computing approach to the shortest path problem is feasible and yields reasonable, useful results.

In short, the work reported in this paper presents initial steps towards quantum computing for graph analysis and exemplifies how to set up a well known problem such that it could be solved on current generation quantum computers.

[1]Indeed, the theoretically best performance for quantum computational search is known to be $O(\sqrt{n})$ [11, 25] and we note that, while simple adiabatic quantum computing as considered in this paper may not reach this level of efficiency, more elaborate versions will [20].

2 ISING MODELS FOR SHORTEST PATHS

In this section, we devise simple Ising models for the shortest path problem as it occurs in the context of route planning. Throughout, we consider undirected, weighted graphs

$$G = (V, E, w) \tag{2}$$

with vertices $V = \{v_1, \ldots, v_n\}$, edges $E \subseteq V \times V$, and edge labeling function $w : E \longrightarrow \mathbb{R}_+$ and we note that an unweighted graph is but a weighted one where $w(v_i, v_j) = 1$ for all $(v_i, v_j) \in E$.

Given such a graph, we can run Dijkstra's algorithm to compute a distance matrix $\boldsymbol{D} \in \mathbb{R}_+^{n \times n}$ such that its elements

$$D_{i,j} = d(v_i, v_j) \tag{3}$$

indicate path lengths between pairs of vertices $v_i, v_j \in V$.

Note that, although $\boldsymbol{D}$ provides information as to the lengths of shortest paths, the paths themselves are not immediately available. Nevertheless, $\boldsymbol{D}$ is of practical value. In route planning, it allows for running (variants of) the A^* algorithm once a source and target vertex have been decided for. Seen in this light, this section thus suggests how to replace A^* computations by a quantum computing approach.

Given G and $\boldsymbol{D}$, we next discuss Ising models for two situations that may occur in context of shortest path computation. First, we address the rather simple case where there is a single shortest path from the source vertex v_s to the target vertex v_t. Second, we consider the more general and more elaborate case where there exist several shortest paths.

2.1 Case I: A Single Shortest Path

Our derivation of an Ising model for the shortest path problem for the case where there is a unique path between source v_s and target v_t is based on the following observation: if v_i is a vertex on the shortest path from v_s to v_t, then

$$d(v_s, v_t) = d(v_s, v_i) + d(v_i, v_t) \tag{4}$$

or, equivalently

$$D_{s,i} + D_{i,t} - D_{s,t} = 0. \tag{5}$$

This immediately provides a criterion for testing whether or not a vertex lies on a shortest path. In order to incorporate it into an Ising model, we let $\boldsymbol{d}_s = D_{s,:}$ be the row of matrix $\boldsymbol{D}$ containing distances w.r.t. v_s and $\boldsymbol{d}_t = D_{t,:}$ be the corresponding row for v_t. Then,

$$\boldsymbol{d} = \boldsymbol{d}_s + \boldsymbol{d}_t - D_{s,t} \cdot \mathbf{1} \geq \mathbf{0} \tag{6}$$

is an n-dimensional vector whose zero elements indicate vertices on a shortest path from v_s to v_t.

For instance, for the unweighted graph in Fig. 1, the entries of the distance matrix reflect "hop counts" between the nine vertices. For source v_1 and target v_6 we would therefore find $\boldsymbol{d} = [010010223]^\intercal$ which indicates that v_1, v_3, v_4, and v_6 lie on the shortest path between source and target.

This way, the problem of identifying a shortest path becomes the problem of identifying the smallest elements of $\boldsymbol{d} \in \mathbb{R}^n$.

For an unweighted graph, we can read the number k of vertices on the path from v_s to v_t from the associated "hop count" matrix and find $k = D_{s,t} + 1$. Identifying the k smallest elements of $\boldsymbol{d}$ is then tantamount to finding a bipolar vector $\boldsymbol{s}$ with k elements equal to $+1$ and $n-k$ elements equal to -1 such that the positive entries of

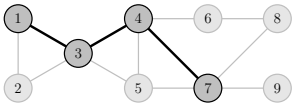


Figure 2: One of the two shortest paths from v_1 to v_7.

$\boldsymbol{s}$ index the sought after elements of $\boldsymbol{d}$. Formally, this corresponds to the following quadratically constrained linear optimization problem

$$\begin{aligned} \boldsymbol{s}^* = \underset{\boldsymbol{s}\in\{-1,+1\}^n}{\operatorname{argmin}} \quad & \boldsymbol{s}^\mathsf{T}\boldsymbol{d} \\ \text{s.t.} \quad & \left(\boldsymbol{s}^\mathsf{T}\mathbf{1} - c_k\right)^2 = 0 \end{aligned} \tag{7}$$

where $c_k \equiv k \cdot +1 + (n-k)\cdot -1 = 2k - n$. Expanding the quadratic constraint

$$\left(\boldsymbol{s}^\mathsf{T}\mathbf{1} - c_k\right)^2 = \boldsymbol{s}^\mathsf{T}\mathbf{1}\mathbf{1}^\mathsf{T}\boldsymbol{s} - 2c_k\boldsymbol{s}^\mathsf{T}\mathbf{1} + c_k^2 \tag{8}$$

where the constant c_k^2 is independent of $\boldsymbol{s}$, the Lagrangian of the problem in (7) amounts to

$$\mathcal{L}(\boldsymbol{s},\lambda) = \boldsymbol{s}^\mathsf{T}\boldsymbol{d} + \lambda\boldsymbol{s}^\mathsf{T}\mathbf{1}\mathbf{1}^\mathsf{T}\boldsymbol{s} - 2\lambda c_k\boldsymbol{s}^\mathsf{T}\mathbf{1} \tag{9}$$

$$= \lambda\boldsymbol{s}^\mathsf{T}\mathbf{1}\mathbf{1}^\mathsf{T}\boldsymbol{s} + \boldsymbol{s}^\mathsf{T}\left(\boldsymbol{d} - 2\lambda c_k\boldsymbol{k}\right). \tag{10}$$

Treating the Lagrange multiplier λ as a constant, we then have $\mathcal{L}(\boldsymbol{s}) = \boldsymbol{s}^\mathsf{T}\boldsymbol{Q}\boldsymbol{s} + \boldsymbol{s}^\mathsf{T}\boldsymbol{q}$ where $\boldsymbol{Q} = \lambda\mathbf{1}\mathbf{1}^\mathsf{T}$ and $\boldsymbol{q} = \boldsymbol{d} - 2\lambda c_k\boldsymbol{k}$. hence, the problem in (7) can be expressed as

$$\boldsymbol{s}^* = \underset{\boldsymbol{s}\in\{-1,+1\}^n}{\operatorname{argmin}} \quad \boldsymbol{s}^\mathsf{T}\boldsymbol{Q}\boldsymbol{s} + \boldsymbol{s}^\mathsf{T}\boldsymbol{q} \tag{11}$$

which we recognize as an Ising energy minimization problem.

For weighted graphs, we cannot read the number k of vertices along a shortest path from the distance matrix. Nevertheless, we may still use (11) to solve our problem. We simply let $c_k = n$ and, for weighted and unweighted graphs, choose λ such that $|\lambda c_k| \in O(1)$.

2.2 Case II: Several Shortest Paths

For situations where there are several shortest paths from v_s to v_t, the optimization objective derived above is overly simplistic. This is because vector $\boldsymbol{d}$ will contain more zeros than there are vertices along either of the shortest paths but the above model does not incorporate structural constraints which would allow for distinguishing between different paths.

For example, for the unweighted graph in Fig. 1 with source v_1 and target v_7 we have $\boldsymbol{d} = [010002022]^\mathsf{T}$ meaning that v_1, v_3, v_4, v_5, and v_7 all lie on a shortest path between source and target. This is certainly correct but neither of the two shortest paths from v_1 to v_7 actually contains all these vertices.

Nevertheless, there is a principled solution, albeit one where we need to consider considerably larger sets of bipolar parameters. Our idea is based on work by Lucas on finding Hamiltonian paths [18] and we shall briefly sketch it here.

We assume that the "hop count" diameter δ of the graph G is known so that the maximum number of vertices along any shortest path is $\Delta = \delta + 1$. Using this, we introduce a bipolar matrix $\boldsymbol{S}$ of size $n \times \Delta$ with elements $S_{v,p}$ where v indexes a vertex of G and p its position along a shortest path from v_s to v_t. Any such matrix $\boldsymbol{S}$ that represents a solution to a shortest path problem is supposed to have $S_{v,p} = +1$ if vertex v is at position p along the path; all other elements of $\boldsymbol{S}$ should be -1. For instance, the shortest path between v_1 and v_7 highlighted in Fig. 2 would be encoded as

	1	2	3	4	5
v_1	+1	−1	−1	−1	−1
v_2	−1	−1	−1	−1	−1
v_3	−1	+1	−1	−1	−1
v_4	−1	−1	+1	−1	−1
v_5	−1	−1	−1	−1	−1
v_6	−1	−1	−1	−1	−1
v_7	−1	−1	−1	+1	−1
v_8	−1	−1	−1	−1	−1
v_9	−1	−1	−1	−1	−1

from which we recognize it to be $v_1 \to v_3 \to v_4 \to v_7$.

This example also provides us with a set of algebraic criteria which any representation of any shortest path must obey.

First, rows s and t of $\boldsymbol{S}$ must contain an entry of +1 and in row s this entry must occur in column one. This constraint is captured by the following minimization term

$$T_1 = \left(1 - S_{s,1}\right)^2 + \left(2 - \Delta - \sum_p S_{t,p}\right)^2. \tag{12}$$

Second and third, for any valid solution, there can be at most one entry of +1 per column and row; this can be enforced by minimizing

$$T_2 = \sum_p \left(1 - \sum_v S_{v,p}\right)^2 \tag{13}$$

$$T_3 = \sum_v \left(1 - \sum_p S_{v,p}\right)^2. \tag{14}$$

Forth, we must penalize non-exiting transitions between vertices

$$T_4 = \sum_{(u,v)\notin E} \sum_p S_{u,p}\, S_{v,p+1}. \tag{15}$$

Finally, the length of any path found this way must of course be small and this can be enforced incorporating vector $\boldsymbol{d}$ from above

$$T_5 = \sum_p \left(\sum_v d_v\, S_{v,p}\right)^2. \tag{16}$$

Putting it all together, we obtain the overall minimization objective

$$T = \sum_{i=1}^{5} \alpha_i\, T_i \tag{17}$$

where the α_i are weighting coefficients that allow for trading off the individual objectives.

Looking at (17), we realize that this objective is quadratic in the $S_{v,p}$. Vectorizing matrix $\boldsymbol{S}$ and rewriting the above terms in form of appropriate coupling matrices and influence vectors therefore provides an Ising model of the form in (11) which would allow for quantum computing solutions to the general shortest path problem. Further details will, however, be left to future work.

2.3 Discussion

While our Ising models for the case where there is a unique shortest path between source and target are almost trivial, the prospective model for the general case of several shortest paths is more involved and more resource intensive. Indeed, solving the former on an adiabatic quantum computer would require a device with only n quantum bits (qubits) whereas the latter would require $n\,\Delta$ qubits.

Moreover, while one might argue that, in practical scenarios such as route planning between, say, cities, shortest path will typically be unique, our simple models can only identify vertices along a shortest path but not the order in which they occur. The extended model, on the other hand, accomplishes identification and ordering simultaneously. In this sense, our experimental results reported below are but a first step in the direction of quantum computing solutions for the shortest path problem.

3 QUANTUM COMPUTING IN A NUTSHELL

Quantum computing harnesses quantum mechanics for information processing and obstacles that prevent it from receiving wider attention in computer science are likely threefold: first, quantum mechanical phenomena such as superposition, entanglement, or decoherence seem abstruse for they cannot be observed in everyday life. Second, the mathematics required to model these phenomena is arguably abstract. And, third, the mathematical notation used in quantum computing appears even more abstract and needs getting used to. Here, we therefore briefly introduce basic terminology and concepts of quantum computing.

The basic units of information on a classical computer are bits and the mathematics that describes their behavior is Boolean algebra. On a quantum computer, the basic units of information are qubits (quantum bits) and the mathematics used to reason about their behavior is complex linear algebra.

While a classical bit assumes one and only one of two possible states (0 or 1), a qubit exists in a superposition of two states simultaneously and only if it is measured does it collapse to either one of them. Examples of physical systems where this phenomenon occurs include the polarization of a photon (vertical or horizontal) or the spin of an electron (up or down).

In order to mathematically describe the behavior of qubits, they are modeled as unit vectors in a two-dimensional Hilbert space. Using the Dirac notation[2], we write a qubit as a linear combination

$$\big|\psi\big\rangle = a\,\big|0\big\rangle + b\,\big|1\big\rangle \tag{18}$$

where the coefficients $a, b \in \mathbb{C}$ are called the amplitudes of the two basis states $\big|0\big\rangle$ and $\big|1\big\rangle$. They obey the normalization condition

$$|a|^2 + |b|^2 = 1 \tag{19}$$

and are interpreted as follows: if a measurement is performed on $\big|\psi\big\rangle$, the probability of finding it in state $\big|0\big\rangle$ is $|a|^2$ whereas the probability of finding it in state $\big|1\big\rangle$ amounts to $|b|^2$.

[2]Readers not familiar with this notation may think of the two basis *kets* $|0\rangle$ and $|1\rangle$ in terms of the Euclidean vectors $[1, 0]^T$ and $[0, 1]^T$, respectively. Yet, any other pair of orthogonal vectors would work as well. One of the reasons why physicists prefer Dirac's notation is that it allows for great symbolic flexibility. For example the two basis polarizations of a photon could be written as $|\updownarrow\rangle$ and $|\leftrightarrow\rangle$ and the two basis spins of an electron could be written as $|\uparrow\rangle$ and $|\downarrow\rangle$, respectively.

Importantly, measurements of $\big|\psi\big\rangle$ are irreversible operations as they constitute interactions with the outside world and thus lead to quantum decoherence. In other words, once a qubit has collapsed to either one of its basis states, it behaves like a classical bit.

Operations on qubits that preserve their quantum mechanical nature are called reversible. Mathematically, these correspond to unitary linear operators $U = e^{-iHt/\hbar}$ where H is another operator called the Hamiltonian. It represents the total energy of a quantum system in the sense that its spectrum is the set of possible outcomes of measurements of the system's total energy.

Just as classical bits can form bit registers, qubits can form qubit registers. Yet, while a single qubit $\big|\psi\big\rangle$ exists in a superposition of 2 basis states, a quantum register $\big|\boldsymbol{\psi}\big\rangle$ of n qubits exists in a superposition of 2^n basis states. Mathematically, this is to say that

$$\big|\boldsymbol{\psi}\big\rangle = \sum_{i=0}^{2^n-1} a_i\,|\boldsymbol{\psi}_i\rangle \tag{20}$$

where the amplitudes obey $\sum_i |a_i|^2 = 1$ and the basis states $|\boldsymbol{\psi}_i\rangle$ of the register are 2^n-dimensional tensor products[3] of single qubit basis states.

For instance, for a quantum register where $n = 3$, we would have the following $2^3 = 8$ basis states

$$|\boldsymbol{\psi}_0\rangle = \big|0\big\rangle \otimes \big|0\big\rangle \otimes \big|0\big\rangle \equiv \big|000\big\rangle \tag{21}$$

$$\vdots$$

$$|\boldsymbol{\psi}_7\rangle = \big|1\big\rangle \otimes \big|1\big\rangle \otimes \big|1\big\rangle \equiv \big|111\big\rangle. \tag{22}$$

There are two distinct approaches towards quantum computing: quantum gate computing and adiabatic quantum computing. The former attempts problem solving by means of sequencing quantum mechanical operators just as digital computers work by sequencing Boolean operators. The latter is the paradigm we are concerned with in this paper.

Adiabatic quantum computing (AQC) is based on the adiabatic theorem [4] which states that if a quantum system starts in the lowest energy state of a Hamiltonian which then gradually changes over a period of time, the system will end up in the ground state of the resulting Hamiltonian. To harness this for problem solving, one prepares a system in the ground state of a simple, problem independent Hamiltonian and evolves it towards a Hamiltonian whose ground state represents a desired solution [1].

Hence, one of the challenges in adiabatic quantum computing is to devise suitable problem Hamiltonians. However, if the problem at hand can be expressed as an Ising energy minimization problem, this challenge is minor because, as we shall see next, it is easy to construct Hamiltonians for Ising models.

4 AQC FOR SHORTEST PATHS

To perform adiabatic quantum shortest path computation on a graph of n vertices based on an Ising model such as in (11), we consider a quantum register of n qubits that is in a time-dependent

[3]The tensor product of two kets can be thought of as the Kronecker product of the corresponding Euclidean vectors. For instance $|0\rangle \otimes |1\rangle$ can be understood as $[1, 0]^T \otimes [0, 1]^T = [0, 0, 1, 0]^T$.

superposition of 2^n basis states

$$\left|\psi(t)\right\rangle = \sum_{i=0}^{2^n-1} a_i(t) \left|\psi_i\right\rangle. \tag{23}$$

Looking at (23), we point out that the time dependence of the system is confined to the amplitudes $a_i \in \mathbb{C}$ whereas the corresponding basis states $\left|\psi_i\right\rangle$ themselves are constants.

With respect to these basis states, we assume that each represents one of the 2^n possible partitions of the graph. In other words, we understand the

$$\left|\psi_0\right\rangle = \left|00\ldots000\right\rangle \tag{24}$$

$$\left|\psi_1\right\rangle = \left|01\ldots000\right\rangle \tag{25}$$

$$\left|\psi_2\right\rangle = \left|10\ldots000\right\rangle \tag{26}$$

$$\left|\psi_3\right\rangle = \left|11\ldots000\right\rangle \tag{27}$$

$$\vdots$$

as indicator vectors which indicate whether a vertex belongs to a path or not.

If a quantum system such as the one in (23) evolves under the influence of a time-dependent Hamiltonian operator $H(t)$, it is governed by the Schrödinger equation

$$\frac{\partial}{\partial t}\left|\psi(t)\right\rangle = -i\, H(t) \left|\psi(t)\right\rangle \tag{28}$$

where we have set $\hbar = 1$. In adiabatic quantum computing, we consider periods ranging from $t = 0$ to $t = T$ and assume the Hamiltonian at time t to be given as a convex combination of two static Hamiltonians, namely

$$H(t) = \left(1 - \tfrac{t}{T}\right) H_B + \tfrac{t}{T} H_P. \tag{29}$$

We say that H_B is the *beginning Hamiltonian* whose ground state is easy to construct and that H_P is the *problem Hamiltonian* whose ground state encodes the solution to the problem we want to solve.

Above, we introduced Ising models for the shortest path problem. For models such as these, there are by now standard suggestions for how to set up a suitable problem Hamiltonian [1]. In particular, we may define

$$H_P = \sum_{i,j=1}^{n} Q_{ij}\, \sigma_z^i\, \sigma_z^j \tag{30}$$

where σ_z^i denotes the Pauli spin matrix

$$\sigma_z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \tag{31}$$

acting on the ith qubit. In other words, σ_z^i is given by a tensor product of n operators, namely

$$\sigma_z^i = \underbrace{I \otimes I \otimes \ldots \otimes I}_{i-1 \text{ terms}} \otimes \sigma_z \otimes \underbrace{I \otimes I \ldots \otimes I}_{n-i \text{ terms}}. \tag{32}$$

The beginning Hamiltonian is then typically chosen to be orthogonal to the problem Hamiltonian, for instance

$$H_B = -\sum_{i=1}^{n} \sigma_x^i \tag{33}$$

where σ_x^i is defined as above, however, this time with respect to the Pauli spin matrix σ_x.

To compute a path, we evolve $\left|\psi(t)\right\rangle$ from $\left|\psi(0)\right\rangle$ to $\left|\psi(T)\right\rangle$ where $\left|\psi(0)\right\rangle$ is the ground state of the beginning Hamiltonian. That is, if λ denotes the smallest eigenvalue of H_B, the initial state $\left|\psi(0)\right\rangle$ of our quantum register corresponds to the solution of the following eigenvector/eigenvalue problem

$$H_B\left|\psi(0)\right\rangle = \lambda \left|\psi(0)\right\rangle. \tag{34}$$

Finally, upon termination of its evolution, a measurement is performed on the qubit register. This will cause the wave function $\left|\psi(T)\right\rangle$ to collapse to a particular basis state and the probability for this state to be $\left|\psi_i\right\rangle$ is given by the amplitude $|a_i(T)|^2$. However, since the adiabatic evolution was steered towards the problem Hamiltonian, basis states that correspond to ground states of H_P are more likely to be found.

On an adiabatic quantum computer, all these ingredients of our quantum shortest path algorithm can be prepared correspondingly and the adiabatic evolution be carried out physically. On a digital computer, we may simulate this quantum mechanical process by numerically solving

$$\left|\psi(T)\right\rangle = -i \int_0^T H(t) \left|\psi(t)\right\rangle dt. \tag{35}$$

The experimental results we present next were obtained using this latter approach.

5 PRACTICAL EXAMPLES

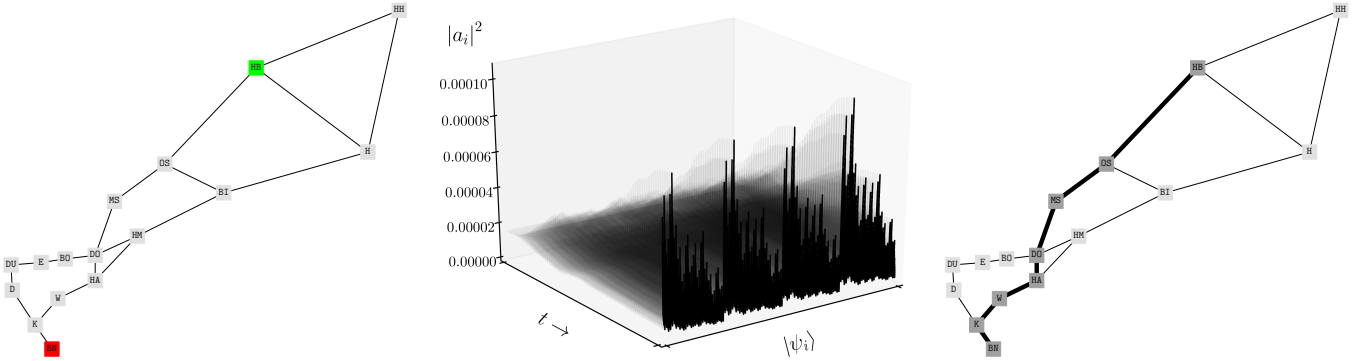
In this section, we present simulation experiments which illustrate the feasibility and inner workings of adiabatic quantum computing for shortest path finding. Our simulations were carried out in *Python* using the scientific computing modules *NumPy* and *NetworkX* and, notably, the quantum computing toolbox *QuTiP* [12, 16, 19].

Both our examples address the case where there is a unique shortest path between a source and a target vertex. In other words, both our examples deal with the Ising model in (11). In both examples, we set up the ground state $\left|\psi(0)\right\rangle$ and the Hamiltonians H_B and H_P according to equations (34), (33), and (30) and considered an evolution of $T = 75$ steps. We used methods provided in *QuTiP* to numerically solve (35) for each $0 \leq t \leq T$ so as to be able to visualize the evolution of the probability amplitudes $|a_i|^2$ and thus to provide insights into the inner workings of the quantum computing approach discussed in this paper.

In our first experiment, we consider the unweighted graph of 9 vertices in Fig. 1 and attempt to identify the shortest path between v_1 and v_6.

Figure 1(c) shows the temporal evolution of the amplitudes $|a_i(t)|^2$ of the $2^9 = 512$ basis states $\left|\psi_i\right\rangle$ the 9 qubit system $\left|\psi(t)\right\rangle$ used for this task can be in. At $t = 0$, all basis states are equally likely but over time their amplitudes begin to increase or decrease. At $t = T$, one basis state has a noticeably higher amplitude than the others so that a measurement of the quantum system will likely cause it to collapse to this most probable state. This state is $\left|101101000\right\rangle$ and can be understood as a vector indicating vertices v_1, v_3, v_4, and v_6 which indeed form the sought after shortest path in Fig. 1(b).

In our second experiment, we consider the weighted real world graph in Fig. 3. It represents high-speed railroad connections among 16 cities in northwest Germany and the task is to find the shortest path (in terms of kilometers traveled) between Bonn and Bremen.



(a) railroad network in northwest Germany (b) adiabatic evolution of a 16 qubit system (c) shortest path between Bonn and Bremen

Figure 3: Real world example of adiabatic quantum path finding. The network in (a) represents high speed railroad connections among 16 German cities; the task is to plan a trip from Bonn (BN) to Bremen (HB). Panel (b) visualizes the adiabatic evolution of a system of 16 qubits that was set up to solve this task. During their evolution over time t, the qubits are in a superposition of $2^{16} = 65536$ basis states $|\psi_i\rangle$ each of which represents a possible subset of cities. Upon termination, state $|1100101000101011\rangle$ has the highest amplitude $|a_i|^2$ and is thus most likely to be measured. It indexes the nodes BN, K, W, HA, DO, MS, OS, and HB which, when sorted according to their distance to the source node, form the path highlighted in (c).

Figure 3(b) depicts the amplitude evolution of the 65336 basis states the corresponding 16 qubit system can be in. At $t = T$, the most likely state to be measured is $|1100101000101011\rangle$. It indexes vertices BN, K, W, HA, DO, MS, OS, and HB which, when sorted w.r.t. their distance to the source, form the path in Fig. 3(c).

6 SUMMARY AND OUTLOOK

Quantum computing is on the verge of becoming an established technology and thus will likely impact graph analysis and mining because many common problems in these areas are known to be solvable on quantum computers [2, 18, 23].

In this paper, we presented first steps towards finding shortest paths by means of adiabatic quantum computing. We devised Ising models for the simple case where there is a unique shortest path between a source and a target vertex, discussed how to set the models up for computation, and presented simulation experiments which demonstrated the feasibility and effectiveness of our ideas.

We also sketched the ingredients of an Ising model for the more general case where there exist several shortest paths between source and target. While we leave further details as to this idea to future work, we believe that our results so far suggest that further research in the direction of (adiabatic) quantum computing for graph mining is worthwhile and may lead to solutions of practical importance.

REFERENCES

[1] T. Albash and D.A. Lidar. 2016. Adiabatic Quantum Computing. *arXiv:1611.04471 [quant-ph]* (2016).

[2] C. Bauckhage, E. Brito, K. Cvejoski, C. Ojeda, R. Sifa, and S. Wrobel. 2017. Ising Models for Binary Clustering via Adiabatic Quantum Computing. In *Proc. EMMCVPR*. Springer.

[3] Z. Bian, F. Chudak, W.G. Macready, and G. Rose. 2010. *The Ising Model: Teaching an Old Problem New Tricks.* Technical Report. D-Wave Systems.

[4] M. Born and V.A. Fock. 1928. Beweis des Adiabatensatzes. *Zeitschrift für Physik* 51, 3–4 (1928).

[5] D. Castelvecchi. 2017. Quantum Computers Ready to Leap out of the Lab. *Nature* 541, 7635 (2017).

[6] E. Connover. 2018. Google Moves toward Quantum Supremacy with 72-qubit Computer. *Science News* 193, 6 (2018).

[7] D-Wave 2017. D-Wave Announces D-Wave 2000Q Quantum Computer and First System Order. press release. (Jan 2017). www.dwavesys.com/press-releases/

[8] Daimler 2018. Daimler Joins Forces with Google to Research the Application of Quantum Computers. press release. (Mar 2018). media.daimler.com

[9] V.S. Denchev, S. Boixo, S.V. Isakov, N. Ding, R. Babbush, V. Smelyanskiy, J. Martinis, and H. Neven. 2016. What is the Computational Value of Finite Range Tunneling? *Phys. Rev. X* 6, 3 (2016).

[10] E. Gibney. 2017. Quantum Computer Gets Design Upgrade. *Nature* 541, 7638 (2017).

[11] L.K Grover. 2001. From Schrödinger's Equation to the Quantum Search Algorithm. *J. of Physics* 56, 2 (2001).

[12] A.A. Hagberg, D.A. Schult, and P.J. Swart. 2008. Exploring Network Structure, Dynamics, and Function Using NetworkX. In *Proc. Python in Science Conf. (SciPy).*

[13] IBM 2017. IBM Announces Advances to IBM Quantum Systems & Ecosystem. press release. (Nov 2017). www-03.ibm.com/press/us/en/pressrelease/

[14] Intel 2018. Intel Advances Quantum and Neuromorphic Computing Research. press release. (Jan 2018). newsroom.intel.com/news/

[15] E. Ising. 1925. Beitrag zur Theorie des Ferromagnetismus. *Zeitschrift für Physik* 31, 1 (1925).

[16] J.R. Johansson, P.D. Nation, and F. Nori. 2013. QuTiP 2: A Python Framework for the Dynamics of Open Quantum Systems. *Computer Physics Communications* 184, 4 (2013).

[17] M.W. Johnson, M.H.S. Amin, S. Gildert, T. Lanting, F. Hamze, N. Dickson, R. Harris, A.J. Berkley, J. Johansson, P. Bunyk, E.M. Chapple, C. Enderud, J.P. Hilton, K. Karimi, E. Ladizinsky, N. Ladizinsky, T. Oh, I. Perminov, C. Rich, M.C. Thom, E. Tolkacheva, C.J.S. Truncik, S. Uchaikin, J. Wang, B. Wilson, and G. Rose. 2011. Quantum Annealing with Manufactured Spins. *Nature* 473, 7346 (2011).

[18] A. Lucas. 2014. Ising Formulations of Many NP Problems. *Frontiers in Physics* 2, 5 (2014).

[19] T.E. Oliphant. 2007. Python for Scientific Computing. *Computing in Science & Engineering* 9, 3 (2007).

[20] J. Roland and N.J. Cerf. 2002. Quantum Search by Local Adiabatic Evolution. *Physical Review A* 65, 4 (2002).

[21] R. Srivastava, I. Choi, and T. Cook. 2016. *The Commercial Prospects of Quantum Computing.* Technical Report. NQIT.

[22] D.L. Stein and C.M. Newman. 2015. *Spin Glasses and Complexity.* University Press Group Ltd.

[23] H. Ushijima-Mwesigwa, C.F.A. Negre, and S.M. Mniszewski. 2017. Graph Partitioning Using Quantum Annealing on the D-Wave System. In *Proc. Int. Workshop on Post Moore's Era Supercomputing.* ACM.

[24] VW 2017. Volkswagen Group and Google Work Together on Quantum Computers. press release. (Nov 2017). www.volkswagen-media-services.com

[25] C. Zalka. 1999. Grover's Quantum Searching Algorithm Is Optimal. *Physical Review A* 60, 4 (1999).