



A Quantum-Inspired Tensor Network Algorithm for Constrained Combinatorial Optimization Problems

Tianyi Hao¹, Xuxin Huang^{2,3}, Chunjing Jia^{3,4*} and Cheng Peng^{3*}

¹Department of Computer Science, Stanford University, Stanford, CA, United States, ²Department of Applied Physics, Stanford University, Stanford, CA, United States, ³Stanford Institute for Materials and Energy Sciences, SLAC National Accelerator Laboratory, Menlo Park, CA, United States, ⁴Department of Physics, University of Florida, Gainesville, FL, United States

Combinatorial optimization is of general interest for both theoretical study and real-world applications. Fast-developing quantum algorithms provide a different perspective on solving combinatorial optimization problems. In this paper, we propose a quantum-inspired tensor-network-based algorithm for general locally constrained combinatorial optimization problems. Our algorithm constructs a Hamiltonian for the problem of interest, effectively mapping it to a quantum problem, then encodes the constraints directly into a tensor network state and solves the optimal solution by evolving the system to the ground state of the Hamiltonian. We demonstrate our algorithm with the open-pit mining problem, which results in a quadratic asymptotic time complexity. Our numerical results show the effectiveness of this construction and potential applications in further studies for general combinatorial optimization problems.

Keywords: tensor network algorithm, quantum many body models, combinatorial optimization problem, open-pit mining, quantum-inspired algorithm

1 INTRODUCTION

Combinatorial optimization is the process of finding an optimal object from a discrete and finite set of objects. Combinatorial optimization has extensive applications in almost every field of industry, such as supply chain optimization [1], transportation networks and power grids [2], and finance [3]. The search space of a combinatorial optimization problem increases rapidly with the problem size. Problems like the Boolean satisfiability problem can have an exponentially large solution space, making an exhaustive search inapplicable for large-scale problems. From the complexity theory perspective, many combinatorial problems fall into the class of NP – *hard*, which is generally believed to be unsolvable in polynomial time on classical computers. Classical algorithms often use heuristics and approximations to find nearly optimal solutions [4]. Quantum algorithms [5], on the other hand, harness the power of randomness, superposition, entanglement, and interference from quantum mechanics, which might lead to an advantage in exploring the solutions of a combinatorial optimization problem [6–8]. The implementation of quantum algorithms is currently limited by small-scale, noisy, and error-prone contemporary hardware [9]; nevertheless, they view the problems from a different perspective, motivating many quantum-inspired classical algorithms to appear [10, 11].

Tensor networks (TNs) have undergone rapid development in the last 2 decades, gaining tremendous success in quantum many-body physics, quantum information sciences, statistical physics, and so on. Tensor network algorithms based on matrix product states (MPS) [12, 13], projected entangled pair states (PEPS) [14, 15], and variational renormalization group

OPEN ACCESS

Edited by:

Shi-Ju Ran,
Capital Normal University, China

Reviewed by:

Tao Xin,
South University of Science and
Technology, China
David Amaro,
Cambridge Quantum Computing
Limited, United Kingdom

*Correspondence:

Chunjing Jia
chunjing@stanford.edu
Cheng Peng
cpeng18@stanford.edu

Specialty section:

This article was submitted to
Statistical and Computational Physics,
a section of the journal
Frontiers in Physics

Received: 28 March 2022

Accepted: 13 June 2022

Published: 08 July 2022

Citation:

Hao T, Huang X, Jia C and Peng C
(2022) A Quantum-Inspired Tensor
Network Algorithm for Constrained
Combinatorial Optimization Problems.
Front. Phys. 10:906590.
doi: 10.3389/fphy.2022.906590

methods [16–18] are very efficient in simulating a large class of quantum many-body systems. The tensor network structure can encode the combinatorial optimization problem with local constraints, providing an idea of utilizing the tensor network to solve combinatorial optimization problems.

This paper presents a quantum-inspired tensor network algorithm to solve constrained combinatorial optimization problems and demonstrates the algorithm with a particular problem with numerical results. The paper is structured as follows: The general quantum-inspired tensor network algorithm is proposed in **Section 2**, and the open-pit mining problem is provided as an example in **Section 3**. **Section 4** shows our numerical results with the quantum-inspired tensor network algorithm and concludes with a discussion of open questions and directions for future work.

2 A GENERAL TENSOR NETWORK ALGORITHM FOR COMBINATORIAL OPTIMIZATION

In this section, we divide our algorithm into four components and describe each in detail. **Section 2.1** explains how to construct the Hamiltonian for a classical combinatorial problem to transform it to a quantum problem. **Section 2.2** serves as inspiration for our core idea, **Section 2.3**, where the former studies how to satisfy the constraints without introducing a penalty term, and the latter details how to construct a tensor network state that represents the superposition of all feasible solutions. Finally, in **Section 2.4**, we show how to find the optimal solution by evolving the tensor network state to the ground state of the problem Hamiltonian.

2.1 Hamiltonian Construction/Problem Mapping

We use an objective function f and a discrete set of feasible solutions \mathbf{x} to specify the combinatorial optimization problem. In many such problems, each solution involves a binary selection of the individual components under certain constraints, denoted as $\mathbf{x} = (x_1, \dots, x_n)$ with binary variables $x_n \in \{0, 1\}$, satisfying constraints $\mathbf{c} = (c_1, \dots, c_m)$. The goal is to find the solution \mathbf{x}^* in all feasible solutions to maximize the objective function f , i.e. $\mathbf{x}^* = \arg \max_{\mathbf{x} \in \{0,1\}^n} f(\mathbf{x})$. This goal can be achieved by mapping the problem to a Hamiltonian and looking for the ground state in the Hilbert space. The Hamiltonian is written as

$$\hat{H} = \sum_{\mathbf{x} \in \{0,1\}^n} a_{\mathbf{x}} |\mathbf{x}\rangle \langle \mathbf{x}|, \tag{1}$$

where $a_{\mathbf{x}}$ are real values representing the elements of the Hamiltonian and $|\mathbf{x}\rangle = |x_1\rangle \otimes |x_2\rangle \otimes \dots \otimes |x_n\rangle$. Note that in this paper, we are interested in problems that require n qubits, where n is the number of binary variables. Each variable x_i of the set of solutions \mathbf{x} is assigned on the basis of the Pauli operator

\hat{Z}_i on the i -th qubit, that is, $\hat{Z}_i = |0\rangle\langle 0| - |1\rangle\langle 1|$ with $|x_i\rangle \in \{|0\rangle, |1\rangle\}$. Thus, a general quantum state can be expressed as

$$|\Psi\rangle = \sum_{\mathbf{x} \in \{0,1\}^{\otimes n}} b_{\mathbf{x}} |\mathbf{x}\rangle. \tag{2}$$

Here $b_{\mathbf{x}}$ represents the linear coefficient that meets the normalized condition, i.e. $\sum_{\mathbf{x}} |b_{\mathbf{x}}|^2 = 1$. Consider a general unconstrained binary linear optimization problem:

$$\min \{\mathbf{w}^T \mathbf{x} : \mathbf{x} \in \{0, 1\}^{\otimes n}\} \tag{3}$$

with weights $\mathbf{w} = (w_1, \dots, w_n)$. Then the Hamiltonian is transformed as

$$\hat{H} = \sum_i \frac{w_i}{2} (\hat{Z}_i - \hat{I}), \tag{4}$$

where I is the identity. Such transformations take linear time in the number of variables.

The ground state of this Hamiltonian, denoted as $|\psi_g\rangle$, is the state that minimizes the energy defined as $E(|\Psi\rangle) = \langle \Psi | \hat{H} | \Psi \rangle / \langle \Psi | \Psi \rangle$. With an appropriate mapping, the original optimization problem is equivalent to solving the ground state of the Hamiltonian. Then the ground-state solving techniques in quantum many-body physics can be utilized to achieve an optimized solution of the combinatorial optimization problem.

2.2 Postselection: A Penalty-Free Approach

For constrained problems, additional terms are usually needed in the Hamiltonian to penalize constraint violations. These penalty terms should be conditioned by multiplying with appropriate penalty factors to ensure that the overall Hamiltonian behaves as intended. Optimization of such a Hamiltonian containing penalty terms inevitably involves tuning additional hyperparameters, which greatly increases the difficulty of the overall optimization process.

In our study, we present a hyperparameter-free algorithm that directly projects a randomized initial quantum state into a subspace that satisfies the constraints. We use a projector, defined as $\hat{P}|\Psi\rangle = |\Psi_v\rangle$, to eliminate states that violate constraints. $|\Psi\rangle$ is an arbitrary state in the Hilbert space, and $|\Psi_v\rangle$ is the projected state that belongs to the subspace denoted by V satisfying the constraints. The projector is then constructed as

$$\hat{P} = \sum_{\mathbf{x} \in V} |\mathbf{x}\rangle \langle \mathbf{x}|. \tag{5}$$

It satisfies $\hat{P}^2 = \hat{P}$. The ground state $|\psi_g\rangle$ under the projection is written as follows:

$$|\psi_g\rangle = \min_{|\Psi\rangle} E(|\Psi\rangle), \tag{6}$$

$$E(|\Psi\rangle) = \frac{\langle \Psi_v | \hat{H} | \Psi_v \rangle}{\langle \Psi_v | \Psi_v \rangle} = \frac{\langle \Psi | \hat{P} \hat{H} \hat{P} | \Psi \rangle}{\langle \Psi | \hat{P} | \Psi \rangle}. \tag{7}$$

We can then extract the optimal solution to the combinatorial optimization problem encoded in $|\psi_g\rangle$. It is worth noting that the solver only works for a normalized state, i.e. $\langle \Psi | \Psi \rangle = 1$, while the projected states are not normalized since a general projector \hat{P} is

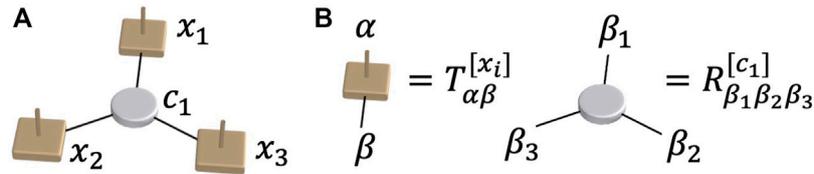


FIGURE 1 | Tensor network construction for a three-variable constrained problem. **(A)** The schematic for the three-variable constrained problem with $x_1, x_2, x_3 \in \{0, 1\}$ and the constraint c_1 . **(B)** The tensor definitions in the general form of the tensor network for the three-variable constrained problem, with α as the physical index and β as the virtual index.

not unitary. However, with prior knowledge that the ground state is always a product state obeying the constraints, and thus $\langle \Psi_0 | \hat{P} | \Psi_0 \rangle = 1$, we could simply apply the solver for the normalized ground state of $\hat{P}\hat{H}\hat{P}$.

2.3 Tensor Network Construction of the Projected State

Directly constructing the projector \hat{P} incurs an exponential cost in time and memory. However, by encoding the constraints in a tensor network state, we can significantly reduce the cost to $O(mn)$, or $O(m + n)$ if the constraints are “local”, *i.e.* the number of variables involved in each constraint is limited by a constant and vice versa. The general form of the tensor network state can be written as

$$|\Psi\rangle = \sum_{\{\alpha, \beta, \dots\}} \prod_{i=1}^n T_{\alpha_i \beta_1 \dots \beta_x}^{[x_i]} R_{\beta_1 \dots \beta_c}^{[c_i]} |\alpha_1 \dots \alpha_n\rangle. \tag{8}$$

There are two types of tensors, $T^{[x]}$ and $R^{[c]}$. Each $T^{[x]}$ has a single physical index labeled α with a bond dimension $d = 2$ reflecting the binary variable, that is, $\alpha = 0$ and 1 , and multiple virtual indices labeled β also with the same bond dimension. Each auxiliary tensor $R^{[c]}$ encodes the constraint c and connects to $T^{[x]}$ tensors whose variable x is involved in the constraint c . To map the selection of the binary variable from the physical index to other indices, $T_{0 \dots 0}^{[x]}$ and $T_{1 \dots 1}^{[x]}$ are set to 1. The $R^{[c]}$ tensors are constructed by traversing the allowed assignments of the constrained variables and setting the elements located in the corresponding indices to 1. We initialize the $T^{[x]}$ and $R^{[c]}$ tensors as zeros. For example, if we have variables $x_1, x_2, x_3 \in \{0, 1\}$ and a single constraint $c_1 : x_1 + x_2 + x_3 \bmod 2 = 1$, then we can use four sparse tensors, *i.e.* $T_{\alpha\beta_1}^{[x_1]}$, $T_{\alpha\beta_2}^{[x_2]}$, $T_{\alpha\beta_3}^{[x_3]}$ and $R_{\beta_1\beta_2\beta_3}^{[c_1]}$ to encode this particular constrained problem, with nonzero elements of the tensors specified as:

$$T_{00}^{[x_1]} = T_{11}^{[x_1]} = 1 \tag{9}$$

$$T_{00}^{[x_2]} = T_{11}^{[x_2]} = 1 \tag{10}$$

$$T_{00}^{[x_3]} = T_{11}^{[x_3]} = 1 \tag{11}$$

$$R_{001}^{[c_1]} = R_{010}^{[c_1]} = R_{100}^{[c_1]} = R_{111}^{[c_1]} = 1. \tag{12}$$

The four possibilities to satisfy the constraint c_1 have been included in $R^{[c_1]}$; for example, $R_{001}^{[c_1]} = 1$ with indices 001 represents $x_1 = x_2 = 0$ and $x_3 = 1$. The construction of these tensors is visualized in **Figure 1**.

By this construction, we encode all feasible variable assignments in the initialized tensor network. Since traversing the allowed assignments of each constraint requires exponential time in the number of variables involved, the overhead of our method is much lower for problems with local constraints, compared to directly building the projector. Additionally, existing tensor algebra methods and tensor network algorithms are applicable to structured problems, helping us solve problems efficiently.

2.4 Finding the Optimal Solution

The optimal solution is one of the basis vectors in **Eq. 8**, denoted by $|\alpha_1 \dots \alpha_i \dots \alpha_n\rangle$, with $\alpha_i = 0, 1$ on the i -th binary variable x_i . Having the superposition of all allowed variable assignments, we can screen out the optimal solution by imaginary time evolution (ITE), a technique to project the initial state to the ground state of an objective Hamiltonian. ITE effectively performs a power iteration by repetitively applying the ITE operator $e^{-\tau\hat{H}}$ to find the ground state of \hat{H} . The ground state minimizes the energy, which is equivalent to minimizing or maximizing the objective function. The simulation of the ITE takes the form of

$$|\Psi\rangle = \lim_{\tau \rightarrow \infty} \frac{e^{-\tau\hat{H}}|\Psi_0\rangle}{\|e^{-\tau\hat{H}}|\Psi_0\rangle\|}, \tag{13}$$

where τ is referred to as the imaginary time. If the Hamiltonian only contains the summation of commuting terms, then $e^{-\tau\hat{H}}$ can be directly rewritten as the product of local evolution operators, *i.e.* $e^{-\tau\hat{H}} = \prod_i e^{-\tau\hat{h}_i}$. Otherwise, the Trotter-Suzuki decomposition [19, 20], an approximation for doing the ITE if containing non-commuting terms in the Hamiltonian, should be involved.

To extract the optimal solution from the tensor networks, we take inspiration from measuring the spin magnetic moment. The assignment of the variable x_i is calculated as

$$x_i^* = 1 \left(\frac{\langle \Psi | \hat{Z}_i | \Psi \rangle}{\langle \Psi | \Psi \rangle} < 0 \right), \tag{14}$$

where \hat{Z}_i is the Pauli matrix. That is, if the expectation value is negative, x_i is 1; otherwise, it is 0. Since we need to do this calculation for each variable, there is an $O(n)$ factor multiplied by the complexity of computing a single expectation value, which we will discuss shortly.

Note that there could be certain variables whose either assignment gives the same final objective value and obeys the constraints, known as the degeneracy. In this case, if we prefer x_i

to be assigned a specific value, for example, zero, then the measurement operator can be adjusted as

$$\hat{Z}_i = \begin{bmatrix} 1 & 0 \\ 0 & -\mu \end{bmatrix}, \tag{15}$$

where μ is a value larger than 1 so as to create a slight difference between the expectation values of the 0 and 1 variable assignments. Then we should measure

$$\tilde{x}_i^* = 1 \left(\frac{\langle \Psi | \hat{Z}_i | \Psi \rangle}{\langle \Psi | \Psi \rangle} < \nu \right), \tag{16}$$

where ν is a threshold value related to μ used to detect the slight difference. In practice, μ can be slightly larger than 1 and ν is a small positive number approaching zero, such as $\mu = 3$ and $\nu = 0.04$, depending on how many degenerate states exist. We can adjust μ until the degenerate states are properly separated; then we can find our preferred one by setting an appropriate threshold ν .

We calculate the expectation values $\langle \Psi | \hat{Z}_i | \Psi \rangle$ by performing tensor network contractions [21–23], which sometimes take exponential time and space to obtain an exact result. Tensor decomposition methods, such as singular value decomposition (SVD), are regularly used in tensor network contractions to limit the rapidly increasing bond dimension. Specifically, we can preserve a predetermined number of columns and rows in the decomposed matrices with the largest singular values. An alternative common practice is to preserve columns and rows whose singular values are above a threshold. Using these techniques cleverly can reduce the complexity of the whole contraction to be polynomial in m and n , while still resulting in an accurate enough value. Moreover, for problems that have a structured tensor network construction, we can utilize existing tensor network algorithms to further optimize performance.

3 THE OPEN-PIT MINING PROBLEM

3.1 Problem Description

We use a real-world combinatorial optimization problem, the open-pit mining problem, as an example to demonstrate our algorithm. The goal of designing optimal open-pit mines is to maximize ore production while avoiding unnecessary mining of rocks. The planning is also subject to a variety of constraints on the size, shape, and form of the mine, making it a computationally taxing process. Theoretical studies of the problem often apply simplifying assumptions, converting the problem into a simpler and more well-understood form [24].

In particular, the 2D open-pit mining problem can be formally stated as a combinatorial optimization problem. Consider a 2D square lattice of mining blocks, where each block i has an associated profit w_i . The coordinate of block i is denoted as (i_x, i_y) . A feasible solution should follow physical constraints, *i.e.* if the block i is excavated, then all its child blocks j should be excavated as well. In this work, we consider the 45° slope constraint: $j \in \{(i_x - 1, i_y - 1), (i_x - 1, i_y), (i_x - 1, i_y + 1)\}$, as

illustrated in **Figure 2A**. Equivalently, we can consider an n -node directed graph $G = (V, E)$ with node weight w_i , as shown in **Figure 2B**. G is structured into levels, where each level contains two more nodes than the previous level, starting from one node in the first level. Each node before the last level has exactly three child nodes at the next level, and each node after the first level has one to three parent nodes.

By assigning $x_i = 0$ (unexcavated) or $x_i = 1$ (excavated) to each node i , we can write the open-pit mining problem as

$$\max \sum_{i=1}^n w_i x_i \tag{17}$$

subject to

$$x_i \in \{0, 1\} \text{ for } i = 1, 2, \dots, n \tag{18}$$

$$\sum_{j \in \text{children}(i)} x_i (1 - x_j) = 0 \text{ for } i = 1, 2, \dots, n, \tag{19}$$

where $\text{children}(i)$ denotes the set of child nodes of node i .

Traditionally, the open-pit mining problem is solved by reducing to the maximum closure problem or the maximum flow problem and utilizing efficient graph algorithms. In particular, the Lerchs-Gorssman (LG) algorithm [25, 26] was the most widely used algorithm in the mining industry, giving a provably optimal solution in polynomial time. In recent years, it has been surpassed by the more efficient Pseudoflow algorithm [27, 28], an $O(|V||E| \log |V|)$ algorithm for the maximum flow problem. Recently, a quantum computing approach was proposed as the first attempt to solve this problem with quantum computers [29]. It modifies the objective function to

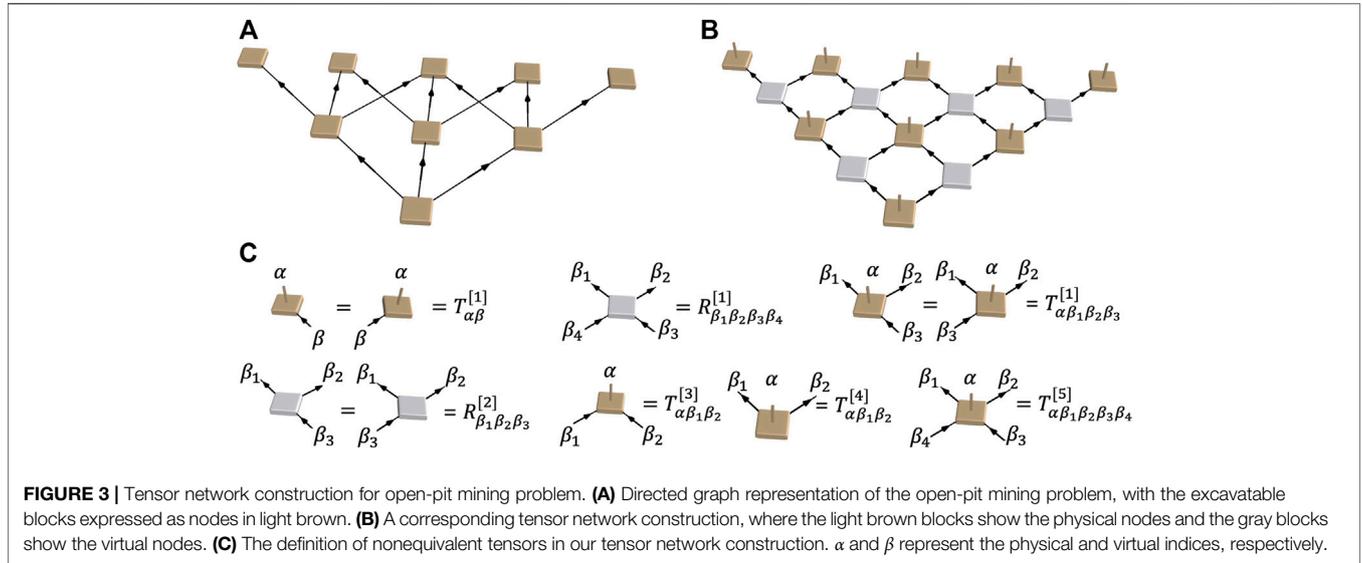
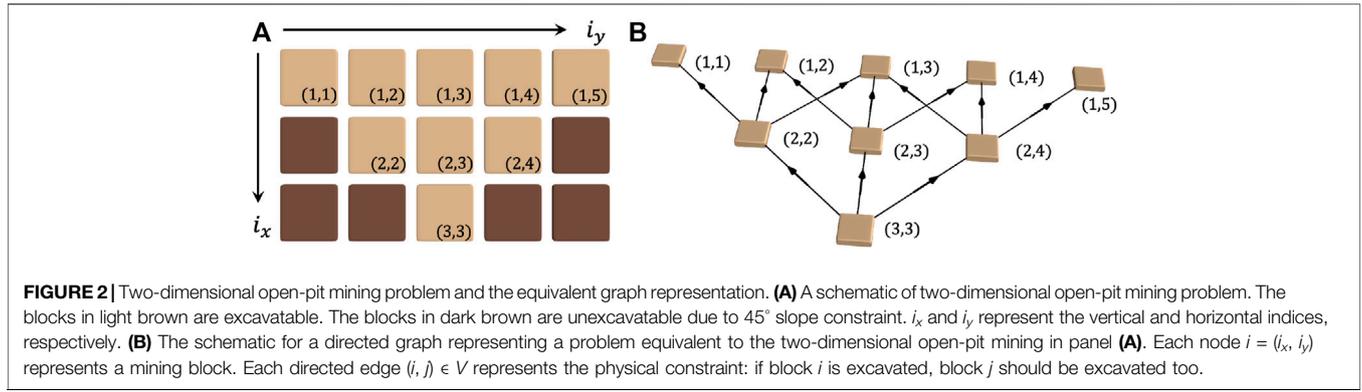
$$\max \left(\sum_i w_i x_i - \lambda \sum_{i,j \in p(i)} x_i (1 - x_j) \right), \tag{20}$$

where λ is a hyperparameter introduced to regularize the penalty for constraint violations. Then the problem Hamiltonian is constructed using the method described in **Section 2.1**.

Our algorithm takes inspiration from the quantum computing approach but is intrinsically different. Using tensor networks to represent quantum states, our algorithm can use powerful non-unitary operations, which are unavailable to quantum algorithms. This fact allows us to directly construct the superposition state of all feasible solutions and avoids having to optimize the regularization of the penalty term. Our algorithm is also completely different from the graph algorithms. It provides a new perspective on such problems, utilizing the rapidly developing tensor network methods. Moreover, the core ideas can be applied to general combinatorial problems, not limited to just this one.

3.2 The Tensor Network Framework

We construct the configurations of all allowed states obeying the smoothness constraints as a tensor network state. **Figure 3** visualizes the construction process using the 5×3 mine as an example. Referring to the smoothness constraints that the walls of the pit should not exceed a maximum steepness, the sites marked dark brown in **Figure 2A** would, if excavated, violate the



smoothness constraints, which therefore should be excluded from the pit profile as well as the initialized tensor network construction. The values of the tensors as defined in **Figure 3C** are as follows (here we group the tensors corresponding to different variables or constraints but have the same form together):

$$T_{11}^{[1]} = T_{00}^{[1]} = 1 \tag{21}$$

$$T_{1111}^{[2]} = T_{0000}^{[2]} = 1 \tag{22}$$

$$T_{111}^{[3]} = T_{000}^{[3]} = 1 \tag{23}$$

$$T_{111}^{[4]} = T_{000}^{[4]} = 1 \tag{24}$$

$$T_{11111}^{[5]} = T_{00000}^{[5]} = 1 \tag{25}$$

$$R_{1111}^{[1]} = R_{1110}^{[1]} = R_{1101}^{[1]} = R_{1100}^{[1]} = R_{1000}^{[1]} = R_{0100}^{[1]} = R_{0000}^{[1]} = 1 \tag{26}$$

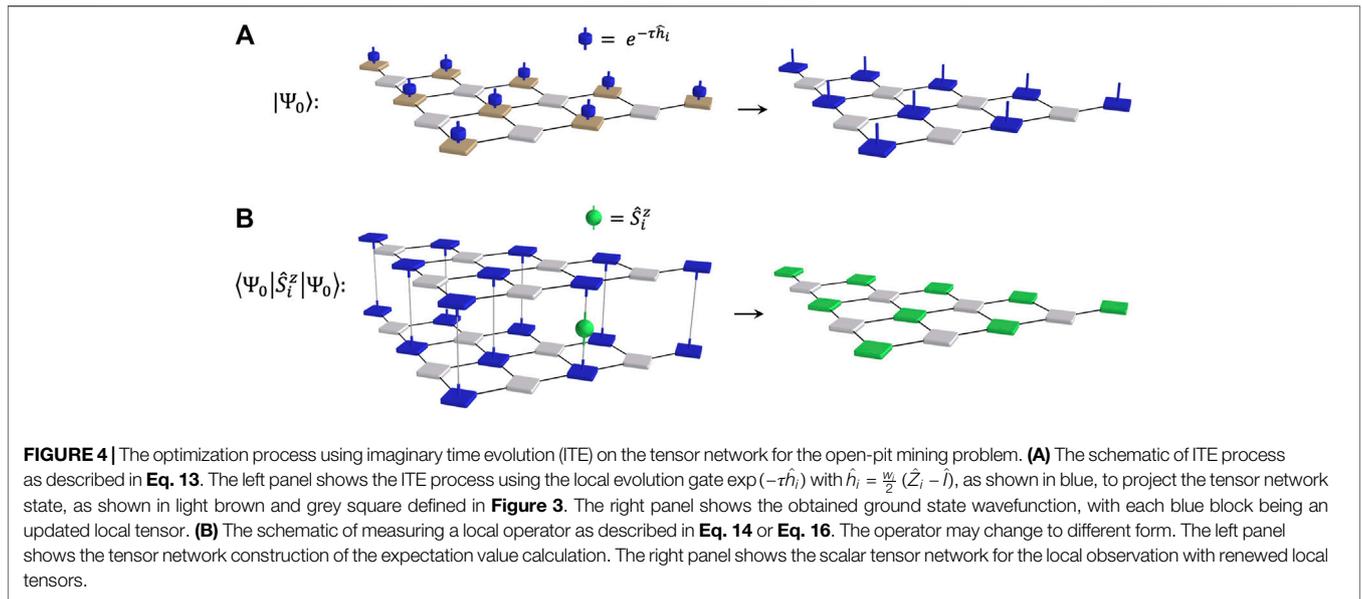
$$R_{111}^{[2]} = R_{110}^{[2]} = R_{100}^{[2]} = R_{010}^{[2]} = R_{000}^{[2]} = 1. \tag{27}$$

All remaining elements are zero, which means that the corresponding projections are forbidden. The first index of the T tensors is the physical index, where $\alpha = 0$ represents an unexcavated block and $\alpha = 1$ represents an excavated block. The virtual indices of T are used to transfer the status of

neighboring tensors. As in the illustration of the tensors shown in **Figure 3C**, the arrow on each index is used to distinguish the directions of the transferring status of the geometrical bonds, where the indices with incoming arrows carry the status originated from their parent blocks and vice versa. The constraints of the excavated ore have been reflected in our tensor network definition with very limited nonzero elements of the tensor. Under the constraint that if a block (i_x, i_y) is excavated, so must its parent blocks $(i_x - 1, i_y - 1)$, $(i_x - 1, i_y)$ and $(i_x - 1, i_y + 1)$, but an excavated block itself does not have to have child blocks.

Since there is only a very limited number of nonzero elements in the initialized tensor network, there is a lot of room for optimization in terms of computational memory and time cost. In addition, the three-dimensional open-pit mine can also be defined in a similar way, just by adjusting the local tensors T and the auxiliary tensors R to higher-order tensors and designing the nonequivalent tensor for boundary conditions in the three-dimensional mine.

The schematic of ITE is shown in **Figure 4A**. As the Hamiltonian for the open-pit mining problem, the same as



defined in Eq. 4, only contains single site commuting terms, we can rewrite the evolution operator for the whole system, *i.e.* $e^{-\tau\hat{H}}$ in Eq.13 into the product of the local evolution operators, *i.e.* $e^{-\tau\hat{H}} = \prod_i e^{-\tau\hat{h}_i}$, with $\hat{h}_i = \frac{w_i}{2}(\hat{Z}_i - \hat{I})$. Then directly set the imaginary time τ as a relatively large number, such as $\tau = 10$, to achieve the ground state in the complexity of $O(n)$.

4 RESULTS AND DISCUSSIONS

We implement our algorithm for the mining problem as an open source Python library, available at [30]. Specifically, the tensor network construction described in Section 3.2 is a 2D tensor network, which can be viewed as projected entangled pair states (PEPS) [15], with trivial physical indices on the R tensors. We use Koala [22], a high-performance PEPS simulation library, to perform the imaginary time evolution and calculate the expectation values by contractions.

We perform numerical experiments on open-pit mining problems with different size scales, as defined in Section 3.1. Problem instances are generated randomly, where the value of each site follows a normal distribution with a mean of 0.1 and a standard deviation of 1. Each problem size is repeated five times with different mine instances to account for the possible fluctuating running condition of the computing device. In an ideal world, the runtime should be the same given a fixed problem size, since the algorithm is purely deterministic. The ITE is conducted with $\tau = 6$, which is large enough to achieve the ground state, but not too large to cause numerical instability, such as exceeding the maximum allowed value of the complex type in Python. Note that one can also set τ with a heuristic related to problem size to better achieve the goals. We obtain the solutions by calculating the expectation values of the Pauli Z operators following Eq. 14.

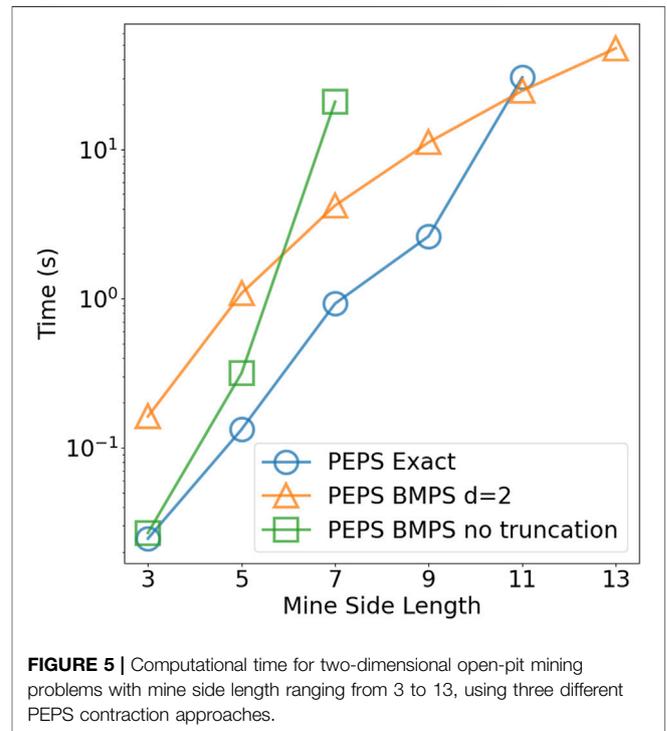


Figure 5 shows a comparison of our algorithm’s runtime with different contraction approaches. If no bond dimension truncation is desired, the “PEPS Exact” method in the figure contracts a PEPS in a generally optimal order [23]. The boundary matrix product state (BMPS) method has poorly scaled performance on its own, but can be executed with bond dimension truncation to significantly lower time and memory complexities [22]. Following the references, we can calculate the asymptotic time complexities of the three contraction

approaches. While “PEPS Exact” and “BMPS no truncation” both take $2^{O(\sqrt{n})}$, BMPS with truncation only takes $O(d^n) = O(n)$, where n is the number of sites. Since we need to do n full contractions, the total time to calculate the expectation values with truncation is $O(n^2)$. As all the other parts of the algorithm run in linear time for the open-pit mining problem, the overall time complexity is $O(n^2)$. In comparison, the state-of-the-art classical algorithm Pseudoflow finds the optimal solution in $O(|V||E| \log |V|) = O(n^2 \log n)$ [27]. Note that although our algorithm is asymptotically faster, its constant factor is larger, and it makes approximations when truncating bond dimensions. In **Figure 5**, we see that truncating the bond dimension to $d = 2$ incurs an additional overhead but has a polynomial runtime in contrast to the exponential runtime of both exact methods, which agrees well with our theoretical analyses. We compare the results produced with the optimal solution given by Pseudoflow. For all problem instances that we run, given an appropriate ITE evolution time, our algorithm can always find the optimal solution with maximized profit and zero constraint violations. Thus, we conclude that truncating the bond dimension to as small as $d = 2$ does not affect the accuracy of our algorithm.

It is worth noting that the open-pit mining problem has some desirable properties for our algorithm. First, the system has a large energy gap between the ground state and the first excited state, allowing ITE to efficiently separate the optimal solution. Second, local and structured constraints decrease the complexity of constructing the tensor networks and make them reducible to well-studied tensor network forms. Third, the Hamiltonian contains only local interactions, resulting in well-controllable long-range entanglement, thus allowing aggressive bond-dimension truncations. It would be intriguing to see how the algorithm performs for more complicated and less suitable problems. Furthermore, by increasing the initial bond dimension, our algorithm can be extended beyond binary variables. A study of the performance sacrifice due to the increased initial bond dimension would be of interest.

REFERENCES

1. Eskandarpour M, Dejax P, Miemczyk J, Péton O. Sustainable Supply Chain Network Design: An Optimization-Oriented Review. *Omega* (2015) 54:11–32. doi:10.1016/j.omega.2015.01.006
2. Yao P, Guo L. Fast Approximation Algorithms for Computing Constrained Minimum Spanning Trees. In: *Combinatorial Optimization and Applications (COCOA 2017)*. Lecture Notes in Computer Science, Shanghai, China, December 16–18, 2017 10627 (2017). p. 103–10.
3. Korte B, Vygen J. *Combinatorial Optimization*, 2. Berlin, Heidelberg: Springer (2012).
4. Lin S, Kernighan BW. An Effective Heuristic Algorithm for the Traveling-Salesman Problem. *Operations Res* (1973) 21(2):498–516.
5. Montanaro A. Quantum Algorithms: An Overview. *Npj Quan Inf* (2016) 2(1): 1–8. doi:10.1038/npjqi.2015.23
6. Moll N, Barkoutsos P, Bishop LS, ChowChowChow JM, Cross A, Egger DJ, et al. Quantum Optimization Using Variational Algorithms on Near-Term Quantum Devices. *Quan Sci. Technol.* (2018) 3(3):030503. doi:10.1088/2058-9565/aab822
7. Farhi E, Goldstone J, Gutmann S. A Quantum Approximate Optimization Algorithm. arXiv e-prints, page arXiv:1411.4028 (2014).

Moreover, since most of the tensor elements are zero in our construction, sparse tensor techniques could be applied, in addition to parallel computing implementations, to further improve performance.

DATA AVAILABILITY STATEMENT

The datasets presented in this study can be found in online repositories. The names of the repository/repository and accession number(s) can be found in the article/Supplementary Material.

AUTHOR CONTRIBUTIONS

CP and TH conceived the study. TH and XH performed numerical simulations. All authors assisted in data interpretation and contributed to writing the manuscript.

FUNDING

This work was supported by the Department of Energy, Office of Science, Basic Energy Sciences, Materials Sciences and Engineering Division, under Contract No. DE-AC02-76SF00515.

ACKNOWLEDGMENTS

The authors thank Mario Motta, Joseph A. Latone, Jay Borenstein, Sreeram Venkatarao, Meltem Tolunay, Allyson Stoll and Stanford CS210 class. The code used to generate the data presented in this study can be publicly accessed on GitHub at [30].

8. Peruzzo A, McClean J, Shadbolt P, Yung M-H, Zhou X-Q, Love PJ, et al. A Variational Eigenvalue Solver on a Photonic Quantum Processor. *Nat Commun* (2014) 5:4213. doi:10.1038/ncomms5213
9. Preskill J. Quantum Computing in the Nisq Era and beyond. *Quantum* (2018) 279:79. doi:10.22331/q-2018-08-06-79
10. Han K-H, Kim J-H. Quantum-Inspired Evolutionary Algorithm for a Class of Combinatorial Optimization. *IEEE Trans Evol Computat* (2002) 6(6):580–93. doi:10.1109/tevc.2002.804320
11. Tang E. A Quantum-Inspired Classical Algorithm for Recommendation Systems. In: *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, Phoenix, Arizona, June 26, 2019 (2019). p. 217–28. doi:10.1145/3313276.3316310
12. Derrida B, Evans MR. Exact Correlation Functions in an Asymmetric Exclusion Model with Open Boundaries. *J Phys France* (1993) 3(2):311–22. doi:10.1051/jp1:1993132
13. Derrida B, Evans MR, Hakim V, Pasquier V. Exact Solution of a 1D Asymmetric Exclusion Model Using a Matrix Formulation. *J Phys A: Math Gen* (1993) 26(26):1493–517. doi:10.1088/0305-4470/26/7/011
14. Verstraete F, Cirac JL. Valence-bond States for Quantum Computation. *Phys Rev A* (2004) 70:060302. doi:10.1103/physreva.70.060302
15. Verstraete F, Cirac JL. Renormalization Algorithms for Quantum-Many Body Systems in Two and Higher Dimensions. arXiv e-prints, pages cond-mat/0407066 (2004).

16. Wilson KG. The Renormalization Group: Critical Phenomena and the Kondo Problem. *Rev Mod Phys* (1975) 47:773–840. doi:10.1103/revmodphys.47.773
17. White SR. Density Matrix Formulation for Quantum Renormalization Groups. *Phys Rev Lett* (1992) 69:2863–6. doi:10.1103/PhysRevLett.69.2863
18. White SR. Density-Matrix Algorithms for Quantum Renormalization Groups. *Phys Rev B* (1993) 48:10345–56. doi:10.1103/physrevb.48.10345
19. Suzuki M. General Theory of Fractal Path Integrals with Applications to Many-Body Theories and Statistical Physics. *J Math Phys* (1991) 32(2):400–7. doi:10.1063/1.529425
20. Suzuki M. Fractal Decomposition of Exponential Operators with Applications to many-body Theories and Monte Carlo Simulations. *Phys Lett A* (1990) 146(6):319–23. doi:10.1016/0375-9601(90)90962-n
21. Ran S-J, Tirrito E, Peng C, Chen X, Tagliacozzo L, Su G, et al. *Two-Dimensional Tensor Networks and Contraction Algorithms*. Cham: Springer International Publishing (2020). p. 63–86. doi:10.1007/978-3-030-34489-4_3
22. Pang Y, Hao T, Dugad A, Zhou Y, Solomonik E. Efficient 2d Tensor Network Simulation of Quantum Systems. In: SC20: International Conference for High Performance Computing, Networking, Storage and Analysis, November 9–19, 2020. IEEE (2020). p. 1–14. doi:10.1109/sc41405.2020.00018
23. Guo C, Liu Y, Xiong M, Xue S, Fu X, Huang A, et al. General-purpose Quantum Circuit Simulator with Projected Entangled-Pair States and the Quantum Supremacy Frontier. *Phys Rev Lett* (2019) 123(19):190501. doi:10.1103/physrevlett.123.190501
24. Lerchs H. Optimum Design of Open-Pit Mines. *Trans CIM* (1965) 68:17–24.
25. Giannini LM. *Optimum Design of Open Pit Mines*. PhD thesis. Curtin University (1990).
26. Khalokakaie R, Dowd PA, Fowell RJ. Lerchs-Grossmann Algorithm with Variable Slope Angles. *Mining Technol* (2000) 109(2):77–85. doi:10.1179/mnt.2000.109.2.77
27. Hochbaum DS. The Pseudoflow Algorithm: A New Algorithm for the Maximum-Flow Problem. *Operations Res* (2008) 56(4):992–1009. doi:10.1287/opre.1080.0524
28. Muir DCW. Pseudoflow, New Life for Lerchs-Grossmann Pit Optimisation. *Spectr Ser Orebody Model Strateg Mine Plann* (2008) 14:97–104.
29. Hindy Y, Pointing J, Tolunay M, Venkatarao S, Motta M, Latone JA. A Quantum Computational Approach to the Open-Pit Mining Problem. arXiv preprint arXiv:2107.11345 (2021).
30. Hao T, Huang X, Cheng P, Jia C. *Q Ore*. Github Repository (2021). Available at: <https://github.com/haoty/qore> (Accessed March 28, 2022).

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 Hao, Huang, Jia and Peng. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.