Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2020) 28: 2088 – 2095
© TÜBİTAK
doi:10.3906/elk-1910-99

The quantum version of the shifted power method and its application in quadratic binary optimization

Ammar DAŞKIN[*]
Department of Computer Engineering, Istanbul Medeniyet University, Istanbul, Turkey,

Received: 15.10.2019 • **Accepted/Published Online:** 03.01.2020 • **Final Version:** 29.07.2020

Abstract: In this paper, we present a direct quantum adaptation of the classical shifted power method. The method is very similar to the iterative phase estimation algorithm; however, it does not require any initial estimate of an eigenvector, and as in the classical case its convergence and the required number of iterations are directly related to the eigengap. If the amount of the gap is in the order of $1/poly(n)$, then the algorithm can converge to the dominant eigenvalue in $O(poly(n))$ time. The method can be potentially used for solving any eigenvalue related problem and finding minimum/maximum of a data set in lieu of Grover's search algorithm. In addition, if the solution space of an optimization problem with n parameters is encoded as the eigenspace of an 2^n dimensional unitary operator in $O(poly(n))$ time and the eigengap is not too small, then the solution for such a problem can be found in $O(poly(n))$. As an example, using the quantum gates, we show how to generate the solution space of the quadratic unconstrained binary optimization as the eigenvectors of a diagonal unitary matrix and find the solution for the problem.

Key words: Quantum algorithms, discrete optimization, power iteration

1. Introduction

On quantum computers, the quantum phase estimation algorithm [1] can be used to compute an eigenpair of a given unitary matrix. The algorithm is the main part of the integer factoring[2] and HHL[3] algorithms and quantum chemistry calculations [4, 5]. In addition to adiabatic quantum computing [6, 7] as a global optimization method and approximation algorithms such as [8]; the introduction of HHL algorithm for solving linear systems of equations has paved the way for more research in iterative methods. These methods so far include the quantum least square [9], the quantum versions of the gradient descent [10, 11], and conjugate gradient method [12] proposed for optimization problems. There is also a considerable research interest for quantum semidefinite programming which can also be used for discrete optimization problems. Quantum speed-ups for semidefinite programming are reported in [13, 14] by forming the solution as a quantum state. Ref.[15] has considered the query complexity of a general convex optimization given by oracles, and provided the conditions to reduce the complexity on quantum computers. Recently, Shao [12] has used the quantum version of the power iteration based on HHL algorithm as a preliminary technique for the quantum Arnoldi algorithm.

In this paper, without using HHL algorithm, we sketch the steps of the shifted power method based on just measurement and Hadamard gates. In the following section, after explaining the classical shifted power method, we describe its quantum version. Then, we analyze its complexity, discuss how it can be used in

[*]Correspondence: adaskin25@gmail.com

the eigenvalaue problems, and present numerical simulations. In Sec.III, we discuss possible applications. In particular, we present a quantum circuit to map a discrete optimization problem such as quadratic binary optimization and quadratic assignment problems to a diagonal unitary matrix, and use the quantum power method to solve these problems.

2. Shifted power method

The most basic numerical eigenvalue algorithms on classical computers are the power method and its variants such as the shifted power method [16]. The shifted power method is a well-known simple classical algorithm used to find the dominant eigenvalue and the eigenvector of a matrix. For a complex matrix U of dimension 2^n with eigenvalues $\lambda_1, \lambda_2, \dots, \lambda_{2^n}$, the shifted power method is described by the following algorithmic steps:

1. An initial nonzero eigenvector $\mathbf{v_0}$ is chosen.

2. For $k = 1 \dots maxIterate$, the following computation is iterated until the convergence:

$$\mathbf{v_k} = \frac{(I-U)\mathbf{v_{k-1}}}{\alpha_k}, \text{ with } \alpha_k = \|(I-U)\mathbf{v_{k-1}}\|. \tag{1}$$

Here, as α_k converges to the dominant eigenvalue of $(I-U)$, $\mathbf{v_k}$ converges to its associated eigenvector. If we assume that $|1-\lambda_1| \geq |1-\lambda_2| \geq \dots \geq |1-\lambda_{2^n}|$, then $\alpha_k \approx |1-\lambda_1|$. Since $(I-U)$ and U have the same eigenvectors, $\mathbf{v_k}$ is also the eigenvector of U associated with the eigenvalue λ_1.

The convergence of this classical algorithm is related to the ratio of the two largest eigenvalues: i.e. in each iteration, the residual error in the estimated eigenvector decreases by a factor of $\frac{|1-\lambda_2|}{|1-\lambda_1|}$. In addition, the total required number of iterations depends on this ratio and scales as:

$$O\left(\frac{n}{\log\frac{|1-\lambda_1|}{|1-\lambda_2|}}\right). \tag{2}$$

This indicates that if the eigengap is a polynomial of n^{-1}, and $||U||_2 = O(poly(n))$, then the required number of iterations is $O(poly(n))$. In particular for an eigengap 2^{-m} and $||U||_2 \leq 1$, the algorithm requires $O(n2^m)$ number of iterations.

2.1. Quantum version

In the quantum version of this algorithm, we will first assume that U is a unitary matrix with the eigenvalues $\lambda_j = e^{i\phi_j}$ and for simplicity $\phi_j \in [0, \pi/2]$. For the eigen-phases $\phi_1 \geq \dots \geq \phi_{2^n}$, this assumption maintains the same order for the magnitudes of the eigenvalues of $(I-U)$: i.e., $|1-\lambda_1| \geq \dots \geq |1-\lambda_{2^n}|$. Therefore, the power iteration converges to the first eigenvalue. The kth iteration of the classical method can be emulated by the quantum circuit presented in Figure 1. As shown in Figure 1, the kth iteration in the algorithm starts with the initial state that involves the output vector of the previous iteration. The final quantum state before the measurement operation on the first qubit is as follows:

$$\frac{1}{2}\left(|0\rangle\left(I+U\right)|\mathbf{v_{k-1}}\rangle + |1\rangle\left(I-U\right)|\mathbf{v_{k-1}}\rangle\right). \tag{3}$$

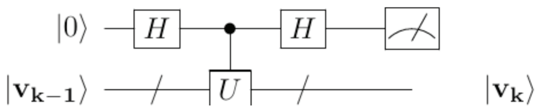

Figure 1. Quantum circuit for the kth iteration of the shifted power method.

After the measurement, the state on the second register collapses onto either $\frac{(I+U)|\mathbf{v_{k-1}}\rangle}{\alpha_{0k}}$ or $\frac{(I-U)|\mathbf{v_{k-1}}\rangle}{\alpha_{1k}}$ with probabilities respectively α_{0k}^2 and α_{1k}^2 which are the squares of the norms of the numerators. If in every iteration, we collapse the state onto the same side (let us say the state where $|1\rangle$ on the first register), then we basically implement the shifted power iteration described above. At the end, α_{1k} converges to $|1-\lambda_1|$ and $|\mathbf{v_k}\rangle$ is to the eigenvector of U. In addition, ϕ_1 can be computed from the statistics of α_{1k} by using the cosine and sine components of $\lambda_1 = \cos\phi_1 + i\sin\phi_1$: Since $(1-\cos\phi_1)^2 + \sin^2\phi = \alpha_{1k}^2$ and $\cos^2\phi_1 + \sin^2\phi = 1$,

$$\phi_1 = \arccos\left(1 - \frac{\alpha_{1k}^2}{2}\right). \tag{4}$$

Here, note that we have used $(I-U)$, however this can be generalized into $(\eta I - U)$ with a real coefficient η by simply not using equal superposition state on the first qubit: i.e. using a different set of gates in lieu of the Hadamard gates on the first qubit. This shifts the origin of the eigenvalues without changing the eigenvector and affects the convergence rate. After computing one eigenpair, shifts along with other deflation techniques[16] can be adjusted for the quantum power method as well computing the rest of the eigenpairs. Also note that when the dominant eigenvalue is multiplicative, the method will generate a superposition of the eigenvectors which can be used for principal component analysis.

2.2. Number of qubits

Although the first register in Figure 1 is drawn with 1 qubit, in the actual implementation the number of qubits can be as much as the parameter $maxIterate$. In addition, the convergence of the method can be determined by forming a simple state tomography of the measured qubit after every few iterations and comparing it with the previous tomography. If it remains the same, then the iteration is stopped.

2.3. Numerical experiment on the number of iterations

As indicated before, the number of iterations is directly related to the eigengap. It is polynomial if the eigengap is polynomially small. As a numerical example, we use a diagonal operator with random phases in $[0, \pi/3]$. Then we fix the eigengap (the difference between the first and the second phases) to 0.01. Then using the equal superposition state, we also set the initial probabilities of each eigenvector to $\frac{1}{2^n}$ which is exponentially small in the number of qubits. Figure 2 shows the mean number of iterations of 15 runs for each n value in order to reach the success probability ≥ 0.5 (the probability of having the dominant eigenvector on the second register). As seen in Figure 2, the number of iterations grows linearly with the number of qubits because of the fixed polynomial eigengap.

In Figure 3, we also show how the number of iterations is affected by the eigengap for $n = 20$ qubits. As expected from Eq.(2), the number of iterations to reach the success probability ≥ 0.5 grows exponentially with the eigengap.

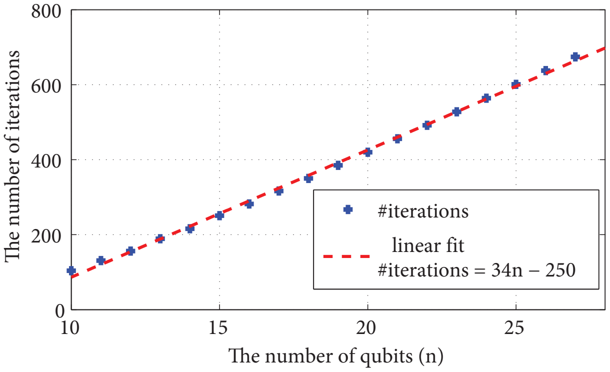

Figure 2. For different n values, the mean number of iterations for 15 runs for a fixed eigengap 0.01. The required number of iterations in general scales as in Eq.(2).

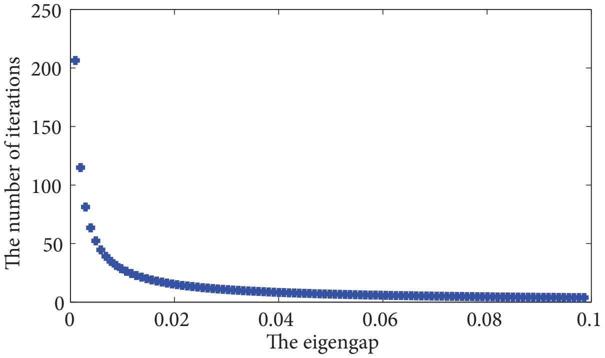

Figure 3. For $n = 20$ qubits, the number of iterations versus the eigengap.

3. Possible Applications

3.1. Eigenpair estimation

On quantum computers, the phase estimation algorithm is the standard procedure for estimating an eigenpair of a given unitary matrix. The algorithm requires a good estimate for the initial state and uses the powers of the given matrix. Therefore, the quantum power method can be considered as an alternative to the phase estimation particularly when an initial good estimate for the eigenvector is not available or the powers of the matrix cannot be computed in an efficient way. However, when the gap between dominant eigenvalues is too small, the algorithm may require too many iterations.

3.2. Unconstrained Optimization

For $\mathbf{x} \in R^n$ and $f : R^n \to R$, $\min_{\mathbf{x}} f(\mathbf{x})$ is an unconstrained optimization problem that arises in the formulation of many problems. When f is smooth, Newton based iterative methods are used in the solution of this optimization by starting with an initial guess $\mathbf{x_0}$ and updating it toward the solution by using the gradient.

In the discrete optimization, $\mathbf{x}$ takes values from a discrete set such as $\{0, 1\}^n$. In terms of quantum states, we can simply formulate this as:

$$\min_{|\mathbf{x}\rangle} |f(\mathbf{x})\rangle \,. \tag{5}$$

Representing each parameter x_i with a qubit q_i, the solution space (the feasible set) can be described by the standard basis set $\{|\mathbf{0}\rangle, \ldots, |\mathbf{2^n-1}\rangle\}$. Thus, $\{|f(\mathbf{0})\rangle, \ldots, |f(\mathbf{2^n-1})\rangle\}$ describes the set of the associated fitness values for each basis (combination). Suppose we have a mechanism U_f to generate the following state:

$$\sum_{\mathbf{x}=\mathbf{0}}^{\mathbf{2^n-1}} \frac{1}{\sqrt{2^n}} |f(\mathbf{x})\rangle |\mathbf{x}\rangle . \tag{6}$$

Then, after finding the minimum $f(\mathbf{x})$ with the Grover's search algorithm [17, 18], we can obtain the corresponding combination for x_is from the second register. Grover's search in total requires $\approx \left(\frac{\pi\sqrt{N}}{4}\right)$ iterations for obtaining the solution with probability ≈ 1. Since any classical minimum finding algorithm would require N operations in the worst case, there is a quadratic speedup in the quantum version. However, this speed-up is not sufficient to overcome the curse of dimensionality that inhibits finding exact solutions to many difficult problems.

In our case, If we have a unitary operator U_f with the eigenvalues $\exp(if(\mathbf{x}))$ and associated eigenvectors $|\mathbf{x}\rangle$s, then the solution of Eq.(5) becomes equivalent to finding the groundstate of the Hamiltonian H in $U = e^{iH}$. Or if we are given U_f that generates Eq.(6), using the quantum Fourier transform, $|f(\mathbf{x})\rangle$ can be mapped to the phases $\exp(if(\mathbf{x}))$. Then the quantum power method can be used to find the maximum $f(\mathbf{x})$ and its associated eigenvector $|\mathbf{x}\rangle$. This may provide exponential speed-up over the search algorithm when the eigengap is $\Omega(poly(1/n))$.

3.3. Quadratic unconstrained binary optimization

A quantum computer cannot solve a general instance of an NP problem in polynomial time unless polynomial-hierarchy in the complexity theory collapses completely. However, it may allow more efficient optimization and approximation algorithms over the known classical ones. Many NP problems can be formulated as a quadratic unconstrained optimization problem (QUBO) used in adiabatic quantum computers (e.g.[19]). Here, without describing a particular NP problem, we will show how to map the general formulation of QUBO into eigenvalues of a diagonal matrix U_f. As in the maximum finding, the quantum power method can be used to extract a maximal solution. Note that this requires exponentially large number of steps when the eigengap is the exponential of n, the number of parameters.

QUBO is defined as maximizing or minimizing the following equation:

$$H = \sum_{j=0}^{n-1} c_j x_j + \sum_{j=0}^{n-1} \sum_{k=j+1}^{n-2} q_{jk} x_j x_k, \tag{7}$$

where c_j and q_{jk} are given real coefficients and x_j and $x_k \in \{0, +1\}$. Considering x_j and x_k as $\in \{-1, +1\}$, this problem is mapped to the Ising type Hamiltonian used in the adiabatic quantum computation [6, 7].

For mapping to a diagonal unitary, we represent each parameter by a qubit. Then, we use one qubit gates Z_js for the summands in the first summation and two qubit gates Z_{jk}s for the ones in the second summation: We define a one-qubit gate for the summand in the first summation as:

$$Z_j = \begin{pmatrix} e^{ic_j\alpha\pi} & 0 \\ 0 & e^{ic_j\beta\pi} \end{pmatrix}, \tag{8}$$

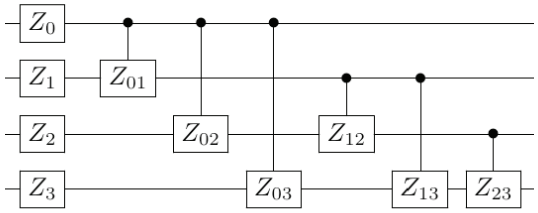

Figure 4. The explicit circuit for the unitary operator of QUBO illustrated for $n = 4$ qubits: For $0 \leq j, k \leq n-1$; while Z_j represents a phase gate on the jth qubit, Z_{jk} is a standard two qubit controlled phase gate on the jth and kth qubits. There are $\frac{n^2+n}{2}$ quantum gates in total.

where either $\alpha = 0$ and $\beta = 1$ or $\alpha = 1$ and $\beta = -1$. Then two qubit controlled gate is defined for the terms in the second summation:

$$Z_{jk} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & e^{iq_{jk}\alpha\pi} & 0 \\ 0 & 0 & 0 & e^{iq_{jk}\beta\pi} \end{pmatrix}. \tag{9}$$

The circuit in Figure 4 represents the alignment of these quantum gates for 4 parameters. This circuit is equivalent to a diagonal matrix whose eigenvectors indicate the possible candidate solutions and eigenphases are the fitness values of these candidates. Using an initial superposition state, the quantum power method can be used to extract the maximum (or minimum) eigenphase in $O(poly(n))$ time if the eigengap is $\Omega(poly(1/n))$.

3.4. Finding groundstates of Hamiltonians

A k-local quantum Hamiltonian can be considered as a binary optimization by mapping Pauli spin matrix σ_z into their eigenvalues $\{-1, 1\}$. Hence, using the QUBO formulation, one can also try to find the groundstate of the Hamiltonian in the following form:

$$H = \sum_{i,j>i}^{n} C_{ij}\sigma_z^i\sigma_z^j, \tag{10}$$

where C_{ij}s are coefficients and σ_z^i indicates a Pauli spin operator on the ith qubit.

3.5. Generalization to the quadratic assignment problem

Quadratic assignment problem is another formulation used to represent many discrete NP-hard optimization problems such as traveling salesman problem. Given three matrices $F = [f_{ij}]$, $D = [d_{kp}]$, and $B = [b_{ik}]$ representing respectively the flow between facilities, the distances between locations, and the allocation costs to the locations; the general form of the quadratic assignment problem of order n is defined as follows[20, 21]:

$$\min \sum_{i,j=1}^{n} \sum_{k,p=1}^{n} f_{ij}d_{kp}x_{ik}x_{jp} + \sum_{k,p=1}^{n} b_{ik}x_{ik}, \tag{11}$$

such that:

$$x_{ij} \in \{0, 1\} \text{ for } 1 \leq i, j \leq n, \tag{12}$$

$$\sum_{i=1}^{n} x_{ij} = 1 \text{ for } 1 \leq j \leq n, \tag{13}$$

and

$$\sum_{j=1}^{n} x_{ij} = 1 \text{ for } 1 \leq i \leq n. \tag{14}$$

In this formulation, $(f_{ij}d_{kp})$ indicates the cost to simultaneously assign the facilities i and j to the locations k and p. Here, note that disregarding the linear term above does not simplify the problem [20].

This problem has also many different mathematical formulations such as trace and Kronecker product: The trace formulation is defined by the following objective function:

$$\min_{X \in \Pi} f(X) = trace(FXD + BX^T), \tag{15}$$

where Π is the set of permutation matrices and X^T denotes the transpose of the matrix X. Considering the vector form, $vec(X)$ of the matrix X, the objective function can be rewritten in the following equivalent Kronecker product form:

$$\min vec(X)^T\,(F \otimes D)\,vec(X) + vec(B)^T vec(X). \tag{16}$$

3.5.1. Formulation of QAP with a quantum circuit

To get the solution from eigenstate, we need to use n^2 number of qubits each of which represents an assignment. In the output if the qubit x_{ij} is 1, then facility i is assigned to the location j. Here, we basically have the same formulation as in QUBO. However, in this case, we use n^2 qubits and $\frac{n^2(n^2-1)}{2} + n^2 = \frac{n^4+n^2}{2}$ number of operations in total.

4. Conclusion

In this paper, we have sketched the steps for the quantum power method and discussed its possible applications. In addition to any eigenvalue related problems, the method can be used to find maximum/minimum of a data set in polynomial time if in the mapped solution space the gap between eigenphases is sufficiently large. With the polynomial mapping, it can be also used as a part of an optimization algorithm to solve many difficult problems. In particular, we formulate the quadratic unconstrained binary optimization and quadratic assignment problem as finding the minimum eigenvalue of a diagonal operator. While on classical computer this formulation requires $O(2^n)$ space, it can be simulated on quantum computers by using $O(poly(n))$ qubits and operations. Although the quantum version can be implemented efficiently on a possible quantum computer based on open quantum systems, the required number of qubits growing linearly with the number of iterations would inhibit the simulation of the method on the current quantum computer technologies for nontrivial n values.

References

[1] Nielsen MA, Chuang I. Quantum computation and quantum information. Cambridge, UK: Cambridge University Press, 2002.

[2] Shor PW. Algorithms for quantum computation: Discrete logarithms and factoring. In: 35th Annual Symposium on Foundations of Computer Science; New York, USA; 1994. pp. 124-134.

[3] Harrow AW, Hassidim A, Lloyd S. Quantum algorithm for linear systems of equations. Physical Review Letters 2009; 103 (15): 150502.

[4] Kassal I, Whitfield JD, Perdomo-Ortiz A, Yung MH, Aspuru-Guzik A. Simulating chemistry using quantum computers. Annual Review of Physical Chemistry 2011; 62: pp. 185-207.

[5] Brown KL, Munro WJ, Kendon VM. Using quantum computers for quantum simulation. Entropy 2010; 12 (11): pp. 2268-2307.

[6] Farhi E, Goldstone J, Gutmann S, Sipser M. Quantum computation by adiabatic evolution. arXiv preprint quant-ph/0001106 2000.

[7] Farhi E, Goldstone J, Gutmann S, Lapan J, Lundgren A, Preda D. A quantum adiabatic evolution algorithm applied to random instances of an np-complete problem. Science 2001; 292 (5516): pp. 472-475.

[8] Farhi E, Goldstone J, Gutmann S. A quantum approximate optimization algorithm. arXiv preprint arXiv:1411.4028 2014.

[9] Wossnig L, Zhao Z, Prakash A. Quantum linear system algorithm for dense matrices. Physical Review Letters 2018; 120 (5): 050502.

[10] Rebentrost P, Schuld M, Wossnig L, Petruccione F, Lloyd S. Quantum gradient descent and Newton's method for constrained polynomial optimization. arXiv preprint arXiv:1612.01789 2016.

[11] Kerenidis I, Prakash A. Quantum gradient descent for linear systems and least squares. arXiv preprint arXiv:1704.04992 2017.

[12] Shao, C. Quantum arnoldi and conjugate gradient iteration algorithm. arXiv preprint arXiv:1807.07820 2018.

[13] Brandao FG, Svore KM. Quantum speed-ups for solving semidefinite programs. In: 58th Annual Symposium on Foundations of Computer Science (FOCS); Berkeley, California, USA; 2017. pp. 415-426.

[14] Van Apeldoorn J, Gilyén A, Gribling S, de Wolf, R. Quantum sdp-solvers: Better upper and lower bounds. In: 58th Annual Symposium on Foundations of Computer Science (FOCS); Berkeley, California, USA; 2017. pp. 403-414.

[15] van Apeldoorn J, Gilyén A, Gribling S, de Wolf R. Convex optimization using quantum oracles. arXiv preprint arXiv:1809.00643 2018.

[16] Saad Y. Numerical Methods for Large Eigenvalue Problems. Philadelphia, PA, USA: Siam, 2011.

[17] Grover LK. Quantum mechanics helps in searching for a needle in a haystack. Physical Review Letters 1997; 79 (2): 325.

[18] Durr C, Hoyer P. A quantum algorithm for finding the minimum. arXiv preprint quant-ph/9607014 1996.

[19] Lucas A. Ising formulations of many np problems. Frontiers in Physics 2014; 2: 5.

[20] Loiola EM, de Abreu NMM, Boaventura-Netto PO, Hahn P, Querido T. A survey for the quadratic assignment problem. European Journal of Operational Research 2007; 176 (2): pp. 657-690.

[21] Rendl F, Pardalos P, Wolkowicz H. The quadratic assignment problem: A survey and recent developments. In: Proceedings of the DIMACS Workshop on Quadratic Assignment problems; New Jersey, USA; 1993. pp. 1-42.