



21, rue d'Artois, F-75008 PARIS
<http://www.cigre.org>

CIGRE US National Committee 2021 Grid of the Future Symposium

Experimental Quantum Computing to Solve Network DC Power Flow Problem

R. ESKANDARPOUR
Resilient Entanglement
USA

K. GHOSH, A. KHODAEI
University of Denver
USA

A. PAASO
Commonwealth Edison
USA

SUMMARY

The power grid is changing ever faster, not only to adopt new technologies but to adapt to climate change. Climate change is driving the world to reduce its use of carbon-based fuels to slow the warming trend by increasing intermittent renewable sources. However, intermittent sustainable energy sources, electric vehicles, and smart loads will vastly change the behavior of the electric power grid and will result in new stochastics, dynamics. Our objective is to provide new analytical approaches that, leveraging exascale computing, can result in the best achievable bounds on these changes and thus resulting in potentially billions of dollars a year in savings by producing the best planning. This paper investigates how a fundamental grid problem, namely DC power flow, can be solved using quantum computing. Power flow is the most widely used power system analysis technique, either as a stand-alone application or embedded in other applications; therefore, its fast and accurate solution is of utmost significance for grid operators. We base our studies on the Harrow-Hassidim-Lloyd (HHL) quantum algorithm, which has a proven theoretical speedup over classical algorithms in solving a system of linear equations. Practical studies on a quantum computer are conducted using the WSCC 9-bus system.

KEYWORDS

Quantum Computing, Resiliency, Sustainable energy, DC power flow

rozhin.eskandarpour@resiliententanglement.us

1. INTRODUCTION

The power flow analysis is the keystone of electric utilities' decision making in power grid operation, control, and planning. The solution of the power flow determines the amount of flow of electricity on various transmission and distribution lines, and voltage values on various nodes. Power flow is used in a wide range of applications from resource management, portfolio optimization, security analysis and grid upgrades, among other purposes. However, it is a daunting task to solve this problem accurately and efficiently due to nonlinearity induced by the laws of physics. The growing proliferation of clean renewable energy resources, the need for increased security and resilience against natural and human-made disasters, and growing expectations of electricity customers to high reliability and quality power have highlighted the importance of fast and accurate power flow studies more than ever.

The traditional methods to solve the power flow problem involved reformulation, approximations or parallel processing to speed up its computation time. Unfortunately, applying these methods to emerging smart grids remains prohibitively ineffective due to a lack of mathematical convergence guarantee and unscalable analytics. In this paper, we propose to model, simulate, and experimentally solve the DC power flow problem on a quantum computer. A quantum computer can achieve enormous processing power by executing multiple functions simultaneously using possible permutations [1][2]; therefore, we expect a computational speedup over classical methods. This is discussed in detail through a sensitivity analysis of respective quantum and classical time complexities.

2. QUANTUM ALGORITHM FOR SOLVING DC POWER FLOW PROBLEM

Solving a system of linear equations (SLE) is fundamental in many science and engineering problems. Quantum computers are proven to provide an exponential speedup for SLE [3][4]. The DC power flow analysis is an SLE that can be efficiently solved using a quantum computer. DC power flow can be illustrated as: given a matrix $B \in \mathbb{R}^{N \times N}$ and a vector $p \in \mathbb{R}^N$, find $\theta \in \mathbb{R}^N$ while satisfying $B\theta = p$. Here B represents the admittance matrix, p denotes nodal injections (positive if generation, and negative if load), and θ that are to be determined, represent the nodal voltage angles.

This SLE is called s -sparse if B has at most s nonzero entries per row or column, which is the case for the DC power flow problem. Solving an s -sparse system of size N with a classical computer requires $O(N s k \log(1/\epsilon))$ running time using the conjugate gradient method [3]. However, a quantum computer using the Harrow-Hassidim-Lloyd (HHL) quantum algorithm can achieve of $O(\log(N) s^2 k^2 / \epsilon)$ running time [5]. Here k and E denote the condition number of the system and the accuracy of the approximation, respectively. The HHL model of the DC power flow problem is achieved as discussed in the following. The DC power flow equations should be rescaled/normalized for quantum applications. We perform this by normalizing p and θ and mapping to the respective quantum states $|p\rangle$ and $|\theta\rangle$.

Matrix B is already Hermitian (has a conjugate transpose similar to itself), so no change on B is needed. The HHL algorithm is implemented, as shown in Figure 1, We will use three types of quantum registers: The first register is used to store a binary representation of the eigenvalues of B and is denoted by α . The second register contains the vector solution indicated by β , where $N = 2\beta$. The third register is for ancilla qubits. All three registers are set to $|0\rangle$ at the beginning of each computation and restored to $|0\rangle$ at the end.

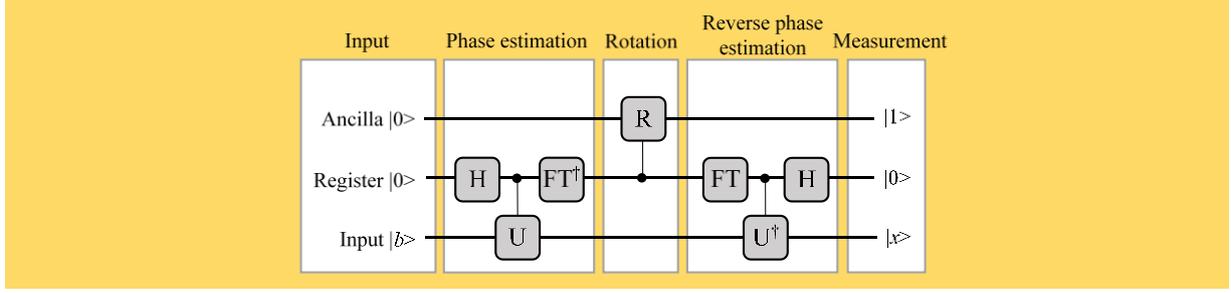


Figure 1. HHL algorithm process

To start the process, we initially load the data into $|p\rangle$, through the following transformation:

$$B|\theta\rangle = |p\rangle \quad (1)$$

Once loaded, we will apply a Quantum Phase Estimation (QPE) to find the eigenvector of B . This is a controlled unitary with a change of basis that maps the eigenvalues onto the working memory. We will define QPE with $U = e^{iBt}$, where

$$e^{iBt} := \sum_j e^{i\lambda_j t} |u_j\rangle\langle u_j|, \quad (2)$$

Then, for the eigenvector $|u_j\rangle_\beta$, which has eigenvalue $e^{i\lambda_j t}$, QPE will output $|\tilde{\lambda}_i\rangle_\alpha |u_i\rangle_\beta$, where $\tilde{\lambda}_i$ represents an α -bit binary approximation to $2^\alpha \lambda_j / 2\pi$. Therefore, if each λ_j can be correctly represented with α bits,

$$QPE(e^{iB2\pi} \sum_j p_j |0\rangle_\alpha |u_j\rangle_\beta) = \sum_j p_j |\lambda_j\rangle_\alpha |u_j\rangle_\beta \quad (3)$$

The quantum state of the register stated in the eigenbasis of B is:

$$\sum_j p_j |\lambda_j\rangle_\alpha |u_j\rangle_\alpha \quad (4)$$

where $|\lambda_j\rangle_\alpha$ is the α -bit binary representation of λ_j . Now that the register is ready, we will add the ancilla qubit and apply a controlled rotation conditioned on $|\lambda_j\rangle$.

2) Controlled Rotation. An ancilla qubit is added and applies a rotation conditioned on $|\lambda_j\rangle$:

$$\sum_j p_j |\lambda_j\rangle_\alpha |u_j\rangle_\alpha \left(\sqrt{1 - \frac{C^2}{\lambda_j^2}} |0\rangle + \frac{C}{\lambda_j} |1\rangle \right) \quad (5)$$

where C is a normalization constant. Now we can apply a reverse QPE,

3) Uncomputation. This step applied a reverse QPE to uncompute the results of the previous step:

$$\sum_j p_j |0\rangle_\alpha |u_j\rangle_\alpha, \left(\sqrt{1 - \frac{C^2}{\lambda_j^2}} |0\rangle + \frac{C}{\lambda_j} |1\rangle \right) \quad (6)$$

4) Measurement. This step measures the ancilla qubit. The register is in the post-measurement state when the outcome is 1:

$$\left(\frac{1}{\sqrt{\sum_j |p_j|^2 / |\lambda_j|^2}} \right) \sum_j p_j |0\rangle_\alpha |u_j\rangle_\alpha \quad (7)$$

In the readout register we obtain a normalized quantum state is the normalized solution of the DC power flow problem. The normalization constant is obtained by measuring the ancilla qubit in (7).

3. NUMERICAL STUDIES

The WSCC 9-bus system, shown in Fig. 2, is used for solving the DC power flow problem using the HHL algorithm. This system approximates the western US power grid, called the Western System Coordinating Council (WSCC), to an equivalent grid with nine nodes and three generators. The system data can be found in [6]. The quantum circuit was designed with Qiskit [7], an open-source software development kit for working with quantum computers at the circuits level and simulated on the IBM Q Experience cloud quantum computing platform.

3.1 Implementation

We attempt to select a proper value for t to help improve the accuracy of the solution returned by HHL. This value would help rescale the eigenvalues during QPE. The fidelity of the solution should be near 1 in an ideal case, however the noise added by the quantum hardware will result in the fidelity of less than 1. Table I demonstrates the results of the studied DC power flow problem. The obtained fidelity is 0.87. This high fidelity shows the promise in finding accurate solutions, despite the noise and the large size of the quantum circuit.

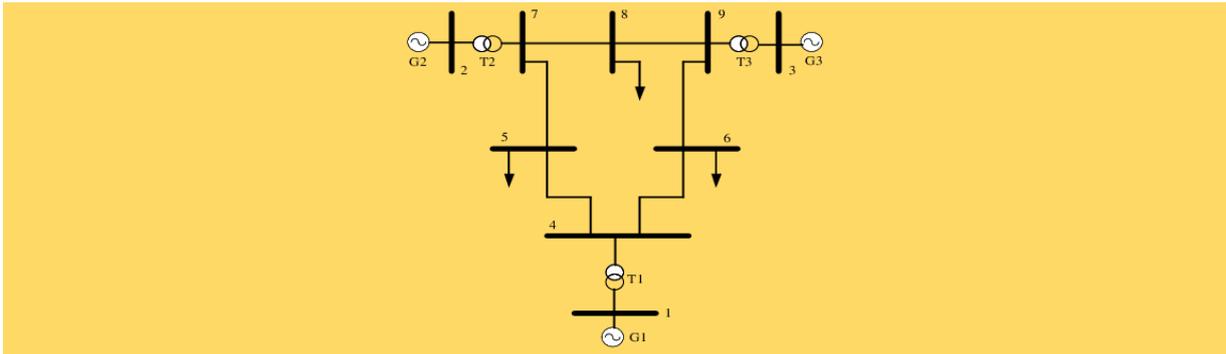


Figure 2. WSCC 9-bus test system.

TABLE I
Quantum Simulation Results

	Results
Fidelity	0.872159
Probability	0.537650
Classical Solution	[0.1157, 0.0948, -0.0713, 0.1316, -0.0644, 0.0118, -0.0514, 0.0220]
Quantum Solution	[0.1125, 0.0599, -0.0299, -0.0826, -0.0458, 0.0047, -0.0858, -0.0040]

Table II shows the characteristics of the quantum circuit used to solve this problem. The depth is the maximum number of gates applied to a single qubit. The width is the required number of qubits. The number of CNOTs is also provided as this number, along with the width, provide a sense of whether running the circuit on current practical hardware is feasible.

TABLE II:
Characteristics of The Quantum Circuit

	Results
Circuit Width	9
Circuit Depth	129
CNOT Gates	70

3.1 Time Complexity and Quantum Speedup

The best known time complexity for the classical solution of the SLE is given by $O(N s k \log(1/\epsilon))$, whereas the time complexity for the HHL algorithm is $O(\log(N) s^2 k^2 / \epsilon)$. We calculate the parameters for the studied problem as follows:

- N is the row/column dimension of the square matrix B , which is the number of system buses minus 1. This is true for larger systems as well. For the studied 9-bus system, N is equal to 8.
- The HHL algorithm assumes that the singular values of B lie between $1/k$ and 1; for our system, and based on the eigenvalues of B , the condition number k is calculated as 0.017.
- The sparsity, s , is the maximum number of nonzero elements in each row. For the studied system, s is 4, stemming from the fact that no node is connected to more than three other nodes (resulting in three non-diagonal elements and one diagonal element in B). The sparsity does not grow significantly with the system size, as the number of nodes connected to a single node is commonly limited in practical power grids.
- The approximation accuracy, t , is the additive error achieved in the output state, related to the desired precision. The HHL algorithm returns an approximate state $|x'\rangle$ of the state $|x\rangle$ of the exact solution. This means $\| |x\rangle - |x'\rangle \| < \epsilon$. Here we assume $\epsilon = \sqrt{2(1 - \sqrt{\text{fidelity}})}$ where fidelity = $\langle |x\rangle - |x'\rangle | \rangle$. For the studied system, fidelity is obtained as 0.87, results in $t = 0.37$.

Using the above-mentioned parameters and the theoretical time complexity for classical and quantum solutions, we can compare the performance of classical and quantum computers in solving the DC power flow problem. For the classical solution, the condition number and sparsity will be similar to the values discussed above. However, considering a much higher fidelity of 0.99 to represent the maturity of the classical computing technology, we obtain $t = 0.1$.

Now we have to make a base assumption of how long it would take for each technology to solve a 2×2 SLE, and then use the time complexities to rescale to the studied 9-bus test system. Plugging the numbers into time complexities, we observe that if the classical computer is 34 times faster than the quantum computer in solving the 2×2 SLE, both technologies will solve the 9-bus system with the same speed. This faster speed in solving a base problem ties the simulation time to physical parameters, including the quantum hardware. Analogously, if we assume that the computation times for both methods are the same in solving the base case, the quantum computer will solve the DC power flow for the 9-bus system 34 times faster than the classical computer.

We build upon this observation, considering a base computational speed difference of 34 times, and re-study the time complexity of both technologies based on more conservative estimates. By increasing either the condition number or the sparsity, the quantum speed will be reduced more than its classical counterpart. Therefore, we consider $k = 0.1$ and $s = 6$ to ensure conservativity, and obtain the results shown in Figure 3 based on a various number of buses. The intersection of these the curves is at $N = 230$, advocating that for 230-bus and larger systems, the quantum computer will find the solution faster than a classical computer. Note this is based on conservative assumptions. This speedup will be more considerable for larger systems, as shown in the figure.

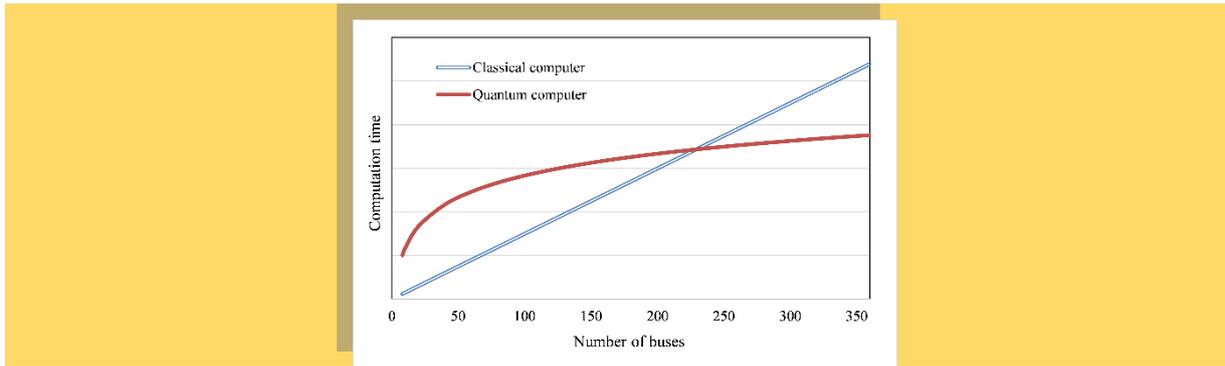


Fig. 3: Comparison between classical and quantum computation time for a various number of buses.

4. Discussions

There are multiple points to consider regarding the obtained results:

- The currently available quantum computers are noisy (fitting under the category of Noisy Intermediate-Scale Quantum (NISQ) computers), so we cannot yet expect an exact solution from this technology, even if tested on small-scale problems as discussed in this paper.
- The discussed quantum speedup is theoretical. Given the limited size of existing quantum computers, experimental speedup comparisons are not possible at this point in time.
- The proposed solution algorithm, i.e., HHL, may present scalability issues when the studied system becomes larger; however, the proposed extended studies were based on the assumption that this algorithm can be applied to larger test systems.

Despite the more theoretical perspective, the studied system demonstrates an obvious superior quantum performance compared to classical solution in terms of computation time. Moreover, the mentioned challenges are expected to be overcome as the quantum computing technology progresses and matures.

5. CONCLUSION

This paper presented an experimental quantum computing solution to the widely-used DC power flow problem. The results showed that a quantum computer could provide higher processing power and solve this fundamental grid problem much faster than a classical computer. It was further shown that this conclusion holds true even for cases that quantum hardware adds delays to computations. As the next step, our mission is to develop and offer a SOFTWARE-as-a-SERVICE SaaS product called Quantum-Power-Flow software to support the deployment of renewable energy generation and responsive demand strategies. We also expect this successful experimental study to enable broader research and applications of quantum computing to power grids in the near future.

BIBLIOGRAPHY

- [1] Committee on Technical Assessment of the Feasibility and Implications of Quantum Computing, Computer Science and Telecommunications Board, Intelligence Community Studies Board, Division on Engineering and Physical Sciences, and National Academies of Sciences, Engineering, and Medicine, Quantum Computing: Progress and Prospects. Washington, DC: National Academies Press, 2019.
- [2] J. Preskill, "Quantum Computing in the NISQ era and beyond," *Quantum*, vol. 2, p. 79, Aug. 2018, doi: 10.22331/q-2018-08-06-79.
- [3] D. Dervovic, M. Herbster, P. Mountney, S. Severini, N. Usher, and L. Wossnig, "Quantum linear systems algorithms: a primer," *ArXiv Prepr. ArXiv180208227*, 2018
- [4] A. Montanaro, "Quantum algorithms: an overview," *Npj Quantum Inf.*, vol. 2, no. 1, pp. 1-8, 2016.

- [5] A. W. Harrow, A. Hassidim, and S. Lloyd, "Quantum Algorithm for Linear Systems of Equations," Phys. Rev. Lett. vol. 103, 150502, 2009.
- [6] A. Delavari, WSCC 9-bus test system IEEE benchmark (<https://www.mathworks.com/matlabcentral/fileexchange/65385-wsc-9-bus-test-system-ieee-benchmark>), MATLAB Central File Exchange. Retrieved October 7, 2020.
- [7] <https://qiskit.org/documentation/stubs/qiskit.providers.aer.QasmSimulator.html>