Communication

Improvement of Quantum Approximate Optimization Algorithm for Max–Cut Problems

Javier Villalba-Diez [1,2] , **Ana González-Marcos** [3] **and Joaquín B. Ordieres-Meré** [4,*]

1   Hochschule Heilbronn, Fakultät Management und Vertrieb, Campus Schwäbisch Hall,
    74523 Schwäbisch Hall, Germany; javier.villalba-diez@hs-heilbronn.de
2   Complex Systems Group, Universidad Politécnica de Madrid, Av. Puerta de Hierro 2, 28040 Madrid, Spain
3   Department of Mechanical Engineering, Universidad de La Rioja, San José de Calasanz 31,
    26004 Logroño, Spain; ana.gonzalez@unirioja.es
4   Escuela Técnica Superior de Ingenieros Industriales (ETSII), Universidad Politécnica de Madrid,
    José Gutiérrez Abascal 2, 28006 Madrid, Spain
*   Correspondence: j.ordieres@upm.es

Abstract: The objective of this short letter is to study the optimal partitioning of value stream networks into two classes so that the number of connections between them is maximized. Such kind of problems are frequently found in the design of different systems such as communication network configuration, and industrial applications in which certain topological characteristics enhance value–stream network resilience. The main interest is to improve the Max–Cut algorithm proposed in the quantum approximate optimization approach (QAOA), looking to promote a more efficient implementation than those already published. A discussion regarding linked problems as well as further research questions are also reviewed.

Keywords: Industry 4.0; quantum approximate optimization algorithm; value–stream networks; optimization

1. Introduction

Value chains linked to Industry 4.0 (I4.0) involve complex cyber-physical networks in which information is processed efficiently by humans and machines to deliver the desired product to a customer [1–3]. I4.0 and the Industrial Internet of Things (IIoT) both describe further emerging landscapes for an integrated human–machine interaction [4,5]. Together, the two concepts are grounded in intelligent, interconnected cyber–physical manufacturing systems that are fully equipped and capable of controlling the process flow of industrial production. Given that many decisions are made independently by machines interoperating with production planning and fabrication systems, the integration of human users requires new paradigms [6].

In the realm of IIoT I4.0 manufacturing, I4.0 vision has advanced the notions of smart fabrication and smart factory by augmenting all assets with sensor–based connectivity [7]. These intelligent sensors generate a large amount of manufacturing data that helps to create digital twins as support for a live mirror of physical processes [8,9]. The ambition is to capture process variability within this approach, with the capability to process all relevant information by analyzing big data in cloud computation so that manufacturers are able to find bottlenecks in manufacturing processes, identify the causes and impacts of problems in such a way that the effective application of measures is useful for both product design and manufacturing engineering, including maintenance, repair and overhaul [10].

Quantum near–term simulations in classical computers have been recently used to solve different applications [6,11], including Industry 4.0 challenges such as the modelling of organizational decision networks as quantum circuits [12]. In this work, with the help of quantum simulations, a new solution for the combinatorial optimization problem is proposed, which can be applied to a wide range of applications including in the Industry

4.0 environment. It consists in finding the "optimal" partitioning of a value chain into two classes, such that the number of connections between them is maximized. Direct applications are linked to introduce flexibility in value chain models by enabling extra resilience to it, no matter whether the related processes are logistics or production related ones. Solving this industrial process design problem potentially allows maximizing the interaction between the elements of the value chain and thus maximizes its productivity [13,14]. Other applications can be connected to the Narrowband Internet of Things (NB–IoT) technology. NB–IoT is a cellular radio–based access protocol specified by 3GPP to tackle the quickly growing market for low–power wide–area connectivity significantly targeting mobile use cases. To realize the global outreach and broad adoption of NB–IoT services, mobile network operators (MNOs) need to guarantee end-to-end devices and services across several vendors connected to the deployed NB–IoT systems, and that the data transport capacity and connection modes are well understood. In this context, efficient dynamic partitions depending on the low power network available are relevant for providing a robust integrative configuration with limited transport overhead.

A solution to these problems can be formulated in terms of a combinatorial optimization approach, which involves finding an optimal object out of a finite set of objects. In this particular case, it involves finding "optimal" bitstrings composed of 0's and 1's among a finite set of bitstrings. Such bitstring represents a partition of nodes of a graph into two sets, such that the number of edges between the sets is maximum. Each of the sets represents the allocation of nodes in the value stream network or nodes in the IoT system to specific managerial structures giving a maximal flexibility by providing the highest degree of connectivity.

This optimization challenge is already known as the Max–Cut problem, and it is one of the most studied combinatorial optimization problems because of its wide range of applications and because of its connections with other fields of discrete mathematics [15]. Different solutions have been proposed for the Max–Cut type of problems, as it belongs to the so-called NP-hard complexity class problems, where no known algorithms are able to solve NP-hard problems in polynomial time and thus exact methods rapidly become intractable. Such solutions include search-based algorithms [16], Machine Learning alternatives [17], as well as Recurrent Neural Networks and Reinforcement Learning [18,19].

Quantum approaches were also proposed with a quantum approximate optimization algorithm (QAOA) by [20]. The QAOA belongs to the class of hybrid algorithms and requires, in addition to the execution of shallow quantum circuits, a classical optimization process to improve the quantum circuit itself. The QAOA is an algorithm that uses unitary transformations $U(\beta_i, \gamma_i)$, depends on two parameters β_i and γ_i, and is arranged in alternating blocks a number p of times ($i \in \{1, \ldots, p\}$) given by

$$|\psi(\vec{\beta}, \vec{\gamma})\rangle = \underbrace{U(\beta_i)U(\gamma_i)...U(\beta_i)U(\gamma_i)}_{p \text{ times}} |\psi_0\rangle \qquad (1)$$

where $|\psi_0\rangle$ is a suitable state and parameters $\vec{\beta}, \vec{\gamma} \in \mathbb{R}^p$.

The goal of the algorithm is to find the combination of parameters that allows a quantum state $|\psi(\vec{\beta_{opt}}, \vec{\gamma_{opt}})\rangle$ to yield the optimal solution [21]. This heuristic algorithm produces then a bit string $x \in \{0,1\}^n$ that with high probability is expected to give a good approximation of the theoretical solution. The algorithm follows a classical optimization scheme: first prepares a parameterized quantum state $|\psi(\vec{\beta}, \vec{\gamma})\rangle$ (called the ansatz), then computes the parameters $(\vec{\beta_{opt}}, \vec{\gamma_{opt}})$ such that the expectation value of the quantum state is given by

$$F_p = \langle \psi(\vec{\beta}, \vec{\gamma})| H_p |\psi(\vec{\beta}, \vec{\gamma})\rangle \qquad (2)$$

is maximized with respect to the problem Hamiltonian H_p, and finally performs a classical optimization until some convergence criterion is reached. An overview of this is shown schematically in Figure 1.
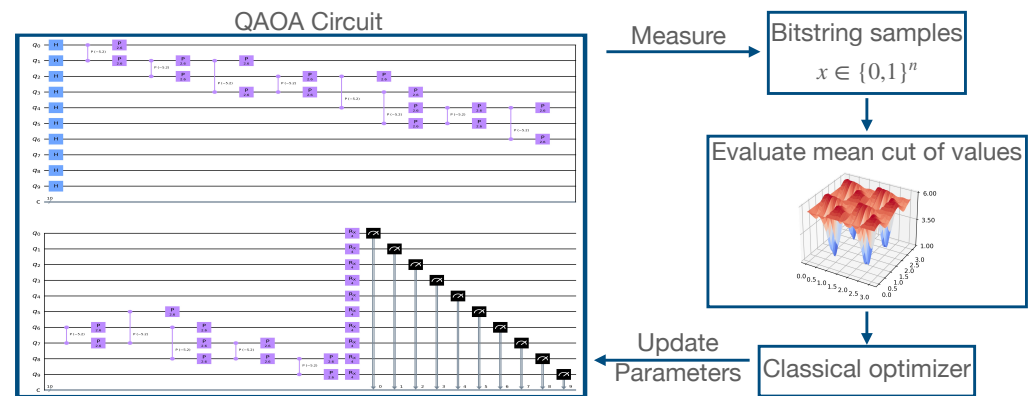
Figure 1. QAOA Overview.

The convergence criterion is in the Max–Cut cost function given by

$$C(x) = \sum_{i,j=1}^{n} x_i(1 - x_j), \tag{3}$$

which can be mapped to a Hamiltonian that is diagonal in the computational basis by

$$H = \sum_{x \in \{0,1\}^n} C(x) |x\rangle \langle x|, \tag{4}$$

in which $x \in \{0,1\}^n$ labels the computational basis states $|x\rangle \in \mathbb{C}^{2^n}$. The expansion of $Z_i = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$ Pauli–Z operators can be obtained from the canonical expansion of $C(x)$ by substitution of every variable $x_i \in \{0,1\}$ by the matrix $\frac{1}{2}(1 - Z_i)$.

As indicated in the abstract, this paper aims to show that the already suggested approximate solution can be improved. The proposal for an alternative quantum algorithm configuration improves the existing solutions up to thirty nodes. The optimization algorithm proposed in Farhi et al. [20] promotes a specific sequence of unitary operators, which means an effective expression for the Hamiltonian. Finally, such a sequence of transformations will perform differently when the size of the circuit evolves. Our approach can be understood in the end as a proposal for a different sequence of unitary operators, providing a different configuration for the Hamiltonian. Then, what it is claimed is that our algorithm (our effective expression for the Hamiltonian) performs much better than the existing one.

The solution is implemented in a simulated quantum hardware environment, however there are already studies showing the time and noise effects over these algorithms when implemented in real hardware [22].

We structure the rest of the work hereinafter as follows: Section 2 outlines the modified architecture in a reasoned manner. Then, Section 3 presents the results of the algorithm as compared with the analytical solution, which for when $|\psi(\overrightarrow{\beta_{opt}}, \overrightarrow{\gamma_{opt}})\rangle$ is not too deep can be computed classically, and the results previously obtained by [20]. Finally, Section 4 briefly discuss the obtained results, outlines future lines of research, and presents limitations in the presented work.

2. Modified QAOA

In this section, we present the results of the algorithm applied to a value stream network of $n = 10$ nodes. The complete results for other configurations are available in open access in [23].

We start by representing in Figure 2 the value stream network as a graph $G = \{n, e\}$ of $n = 10$ unlabeled nodes and $e = 13$ edges.
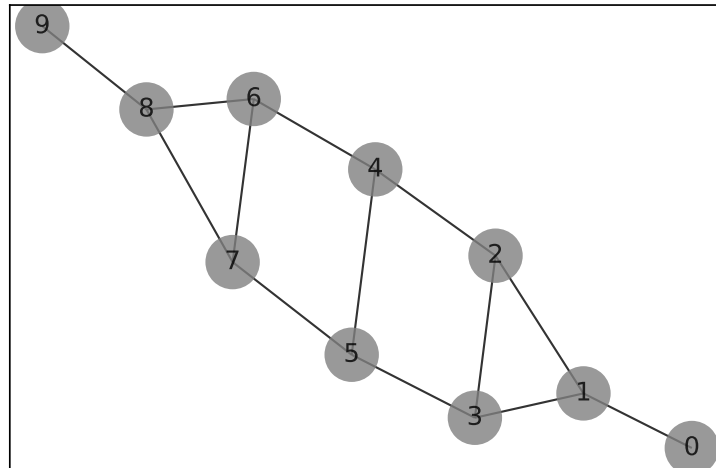
Figure 2. Value stream network with $n = 10$ nodes.

If the graph coincides with the connectivity of our logical network (either IoT topology or value stream network), the cost function $C(x)$ coincides with the hamiltonian H used to generate the state.

For a shallow approximation with $p = 1$, the analytical solution for the expectation value is given by

$$F_1(\beta, \gamma) = \langle \psi_1 | H | \psi_1 \rangle \tag{5}$$

Combining Equations (3) and (4), the Hamiltonian H makes use of the expectation value to measure the edges individually:

$$f_{i,k}(\gamma, \beta) = \frac{1}{2} \langle \psi_1(\gamma, \beta) | (1 - Z_i Z_k) | \psi_1(\gamma, \beta) \rangle \tag{6}$$

There are two types of edges: those that connect a node with degree one (A), and those that connect a node with degree three (B). For the A–class edges, an example of the encoding of the optimization function between nodes (0) and (1) is given by

$$2f_A = 1 - \langle +^1 | U_{01}(\gamma) U_{12}(\gamma) U_{13}(\gamma) X_0(\beta) X_1(\beta) Z_0 Z_1 X_1^\dagger(\beta) X_0^\dagger(\beta) U_{01}^\dagger(\gamma) U_{12}^\dagger(\gamma) U_{13}^\dagger(\gamma) | +^1 \rangle \tag{7}$$

and for the B–class edges, the encoding of the optimization function between nodes (1) and (2) is given by

$$2f_B = 1 - \langle +^3 | U_{21}(\gamma) U_{24}(\gamma) U_{23}(\gamma) X_1(\beta) X_2(\beta) Z_1 Z_2 X_1^\dagger(\beta) X_2^\dagger(\beta)(\gamma) U_{12}^\dagger(\gamma) U_{23}^\dagger(\gamma) U_{24}^\dagger(\gamma) | +^3 \rangle \tag{8}$$

in which $|+^n\rangle = \sum_{x \in \{0,1\}^n} \frac{1}{\sqrt{2^n}} |x\rangle$ prepares for an equal superposition state followed by a sequence of parametrized unitary operations. As shown in Equations (7) and (8) these unitary operations are a combination of parametrized Hamiltonian cost $U_C(\gamma) = e^{-i\gamma H_C}$ and mixer layers $U_M(\beta) = e^{-i\beta H_M}$. The subindexes in the unitary operations indicate the nodes on which the operators act upon.

In our case $n = 10$, there are two A–class edges and eleven B–class edges. This yields Equation (9), which is depicted in Figure 3 which shows the periodicity in both parameters and exhibits a highly nonlinear behaviour. Farhi et al. [20] proposed QAOA with the structure represented in Figure 4.

$$F_1(\beta, \gamma) = 2f_A(\beta, \gamma) + 11f_B(\beta, \gamma) = \left[\sin(4\gamma)\sin(4\beta) + \sin^2(2\beta)\sin^2(2\gamma) \right] +$$
$$+ \frac{11}{2} \left[1 - \sin^2(2\beta)\sin^2(2\gamma)\cos^2(4\gamma) - \frac{1}{4}\sin(4\beta)\cos(4\gamma)\left(1 + \cos^2(4\gamma)\right) \right] \tag{9}$$
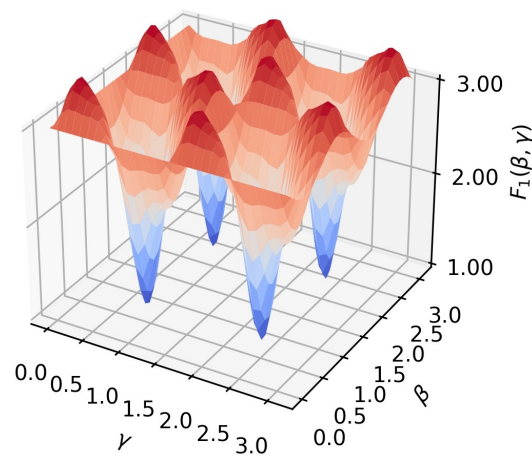
Figure 3. Analytic solution for $p = 1$ and value stream network configuration from Figure 2.
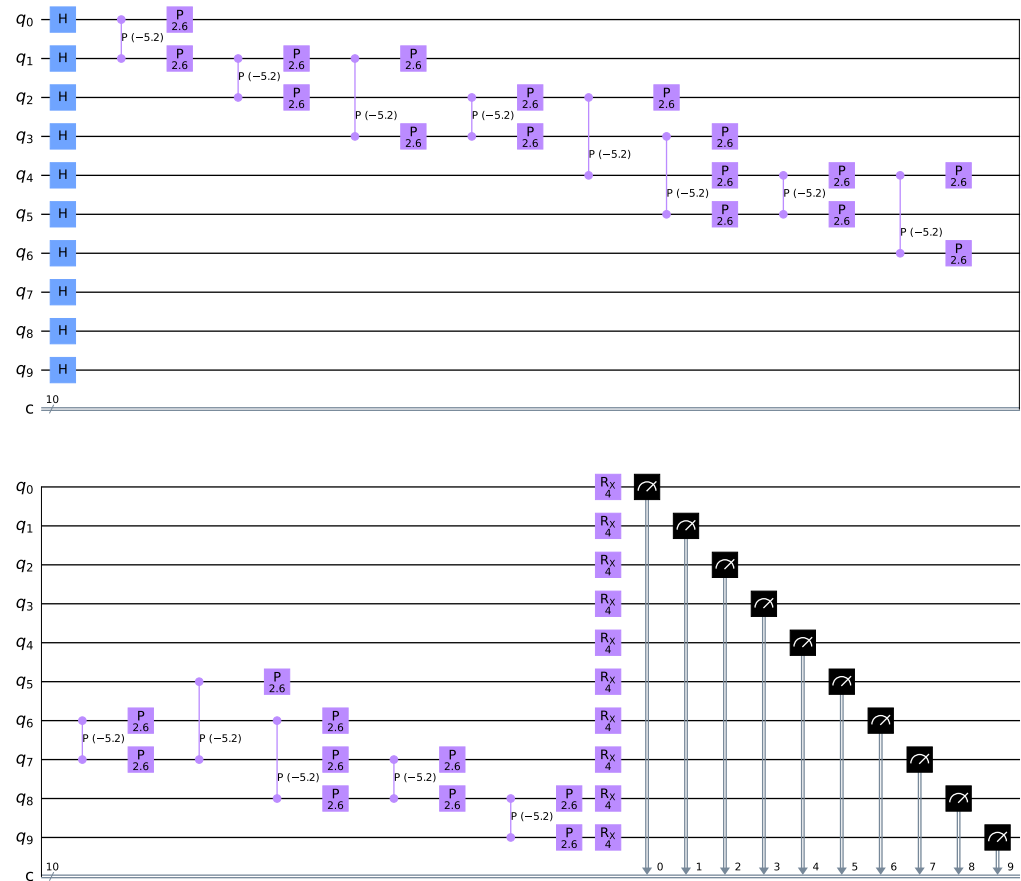

Figure 4. QAOA—Farhi et al. [20].

Such an algorithm starts by preparing the system in superposition with a Hadamard gate on all qubits. Next, a rotation of 2γ is applied to each of the edges if both are in state $|11\rangle$. This conditional rotation has the form given by

$$C_p(2\gamma) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & e^{-2i\gamma} \end{pmatrix}. \tag{10}$$

This allows the algorithm to be applied when both qubits are in state $|1\rangle$ simultaneously. A quantum phase correction of γ is then applied to each of the nodes joined by each edge. This rotation has the form given by

$$p(\gamma) = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\gamma} \end{pmatrix}. \tag{11}$$

Such configuration allows the previous rotation to be neutralized when the two qubits are in state $|11\rangle$. The result of these two operations allows a rotation of γ to be applied to all nodes as long as both communicating nodes are not simultaneously in state $|1\rangle$.

Finally, a rotation around the X-axis of 2β, perpendicular to the computing axis, is applied to all nodes. This rotation has the form given by

$$R_x(2\beta) = \begin{pmatrix} \cos\beta & -i\sin\beta \\ -i\sin(2\beta) & \cos(2\beta) \end{pmatrix}. \tag{12}$$

In summary, in [20] the QAOA algorithm applies, after a standard superposition, a quantum phase of γ to every node connected to each other, as long as both are not in state $|11\rangle$, and a rotation around the perpendicular to the computational axis of 2β to all the nodes.

On the other hand, this paper proposes a novel QAOA approach represented in Figure 5.

Analogous to the previous example, our algorithm starts by preparing the system in superposition with a Hadamard gate on all qubits. We then perform a conditional rotation of γ to each node connected to another if the second is in state $|1\rangle$ in both directions. This is done by concatenation of two $U3\left(\frac{\gamma}{2},0,0\right)$ and $U3\left(\frac{-\gamma}{2},0,0\right)$ gates given by

$$U3\left(\frac{\gamma}{2},0,0\right) = \begin{pmatrix} \cos\left(\frac{\gamma}{4}\right) & -\sin\left(\frac{\gamma}{4}\right) \\ \sin\left(\frac{\gamma}{4}\right) & \cos\left(\frac{\gamma}{4}\right) \end{pmatrix}, \tag{13}$$

and a conditional CX rotation applied to one of the nodes q_0 taking the other q_1 as control given by

$$CX_{q_0,q_1} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}. \tag{14}$$

This method works because when the control *qubit* $|\Psi_0\rangle$ is in state $|0\rangle$, all we have is $U3\left(\frac{\gamma}{2},0,0\right)$ followed by a $U3\left(\frac{-\gamma}{2},0,0\right)$ and the effect is trivial. On the other hand, when the control *qubit* $|\Psi_0\rangle$ is in state $|1\rangle$, the net effect is a controlled rotation $U3(\gamma,0,0)$ on the $|\Psi_1\rangle$ *qubit*. These rotations are taken in both directions because our network is not directed. This algorithm is expected to yield better results than the previous one because the transformations are differential as a function of node state.

Finally, as in the previous algorithm, a rotation around the X-axis of 2β, perpendicular to the computing axis, is applied to all nodes. This rotation has the form given by Equation (12).
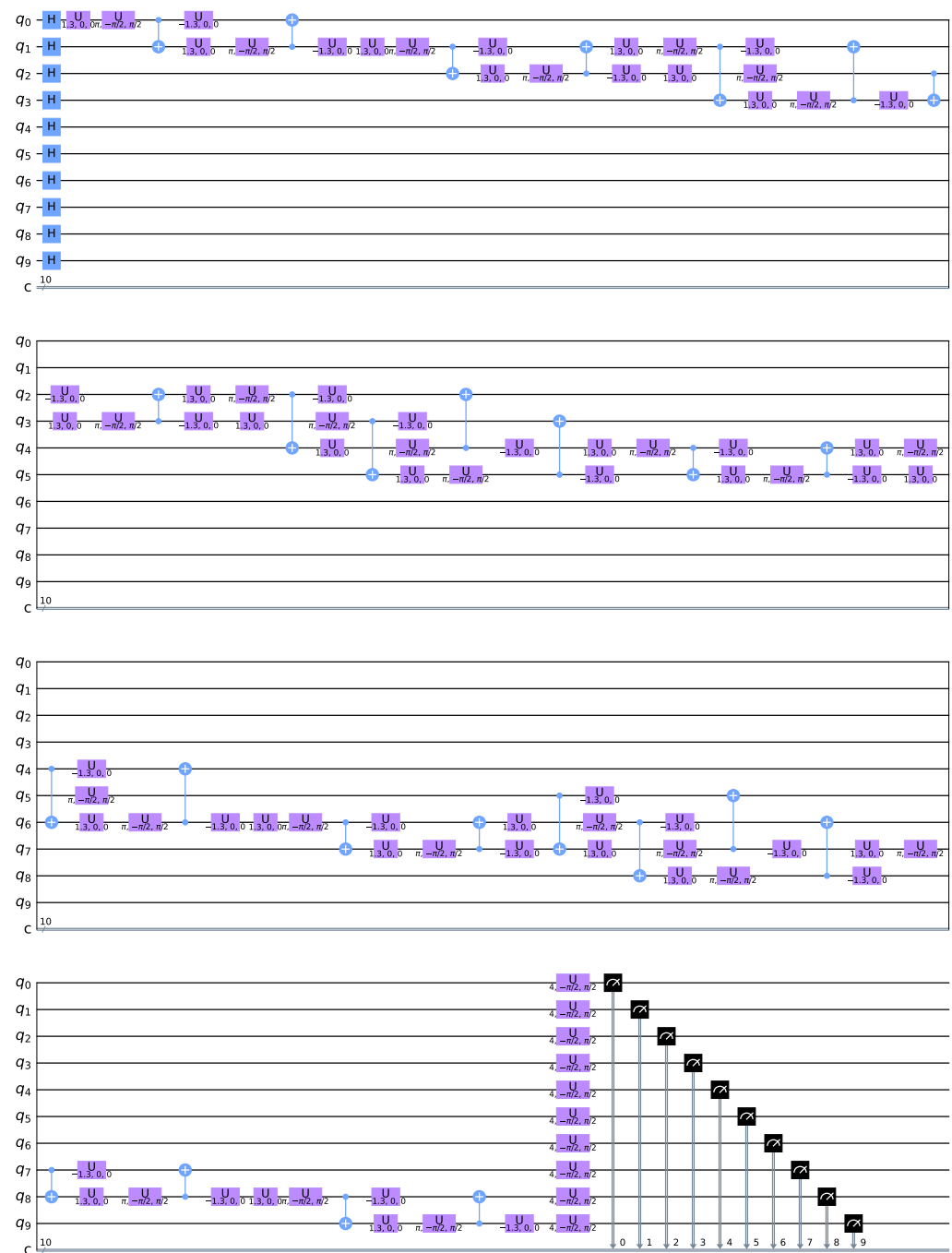
Figure 5. QAOA—Villalba–Diez et al.

3. Results

In this section, we present the results of the algorithm applied to a logical nondirected network of *n* = 10 nodes. The quantum simulations presented were simulated on *qiskit* tool, a Python–based quantum computing platform developed by IBM [24], and the code and additional results can be accessed in this Open Access Repository: [23].

The results confirm our expectations and our proposed QAOA algorithm predicts the analytical results better for a shallow quantum circuit with $p = 1$. A summary of the results for different numbers of nodes is shown in Figure 6. In Table 1 we represent the comparison of the analytical solution curve and the respective QAOA algorithms. Our solution shows better performance in all metrics.
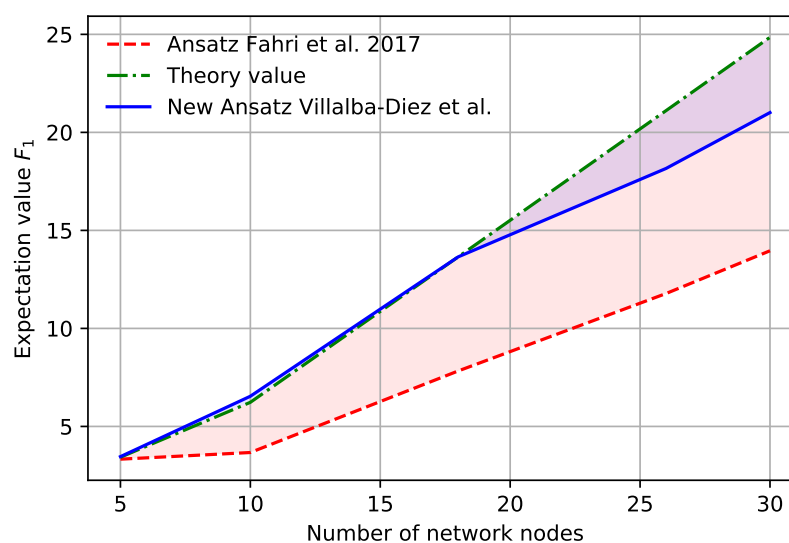
Figure 6. QAOA results comparison.

Table 1. Results comparison for different measures for identifying curve similarity [25].

	Analytic vs.	
	Farhi et al.	**Villalba et al.**
Directed Hausdorff distance	8.22	3.84
Discrete Fréchet distance	10.89	3.84
Dynamic Time Wrapping	28.70	7.13
Partial Curve Mapping	1.6893	0.3223
Area between two curves	1.2744	0.3642
Curve-Length distance metric	141.21	26.23

The bit string that delivers the optimal solution is $x = \{0110011010\}$, as shown in Figure 7. This graph clearly shows the configuration obtained by the QAOA algorithm presented with two types of nodes represented in two colors, green $\{0\}$ and blue $\{1\}$.
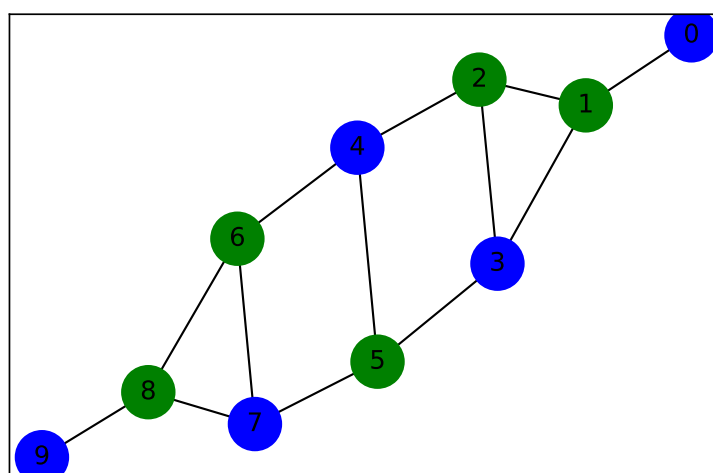

Figure 7. Value stream network clustering with QAOA.

4. Discussion, Future Lines of Research, and Limitations

The analysis of the solution presented in Table 1 shows to what end the quality of the solution found improves the previous one, which justifies the spent effort in considering smarter quantum circuits for the operation of the QAOA algorithm since there is not exist a universal strategy that works across a broad range of optimization problems. Based on

the proposal made, the benefit of the algorithm proposed is evident against other existing algorithms, at least in the case of p = 1. As a consequence, value–stream network design challenges can be better understood with the aid of this quantum optimization algorithm. More research is needed to analyze the evolution of potential benefits when the number of transformation blocks grows up. Indeed, resources and performance figures are also needed to get the whole perspective.

The network topology can be modified, however, if classes for the number of links per node are extended, then a new formulation for the cost function introduced in Equation (9) is required.

Future research lines will involve the implementation of this new proposal for quantum circuits in physical quantum computers, to analyze both the performance and the stability against noise, not only for p = 1 but also when the transformation blocks are increased. Moreover, since two-qubit gates (e.g., CNOT gates) are significantly more erroneous than single qubit gates, the proposal of smarter circuits with reduced number of two-qubit gates, such as the optimization proposed by [22], is an area of interest.

Because of the problem formulation, the network was defined and just the link of nodes with different managerial classes was the goal. However, it could be possible to reverse the problem and start from the node type distribution and look to connect those nodes with a number of edges optimizing the imputation rules between them.

Some limitations can be found regarding the applicability to real cases, because the existence of extra constraints applicable to nodes or edges. Therefore, additional aspects related to penalty terms when formulation of the $C(x)$ function could be a potential workaround. Following this line, another relevant research area is to extend the current formulation for the Max–Cut problem to the Max–k–Cut one, in line with the recent analysis provided by [26]. Although we have obtained satisfactory results with $p = 1$, we can expect a better approximation for a larger number of qubits if we increase the p parameter. This would, however, entail additional relative difficulties in factoring the Hamiltonian in the adiabatic hypothesis that may be problematic in practice.

Author Contributions: Conceptualization, J.V.-D.; methodology, J.V.-D.; software, J.V.-D.; validation, J.V.-D. and A.G.-M.; formal analysis, J.V.-D. and A.G.-M.; investigation, J.V.-D.; resources, J.B.O.-M.; data curation, J.V.-D.; writing—original draft preparation, J.V.-D.; writing—review and editing, J.V.-D.; visualization, J.V.-D.; supervision, J.B.O.-M. and A.G.-M.; project administration, J.V.-D.; funding acquisition, J.B.O.-M. All authors have read and agreed to the published version of the manuscript.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The code can be found at https://shorturl.at/hnpC3 (accessed on 1 December 2021) [23].

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

I4.0	Industry 4.0
IIoT	Industrial Internet of Things
MNO	Mobile Network Operator
NB–IoT	Narrowband Internet of Things
QAOA	Quantum Approximate Optimization Approach

References

1. Lee, J.; Bagheri, B.; Kao, H.A. A Cyber-Physical Systems architecture for Industry 4.0-based manufacturing systems. *Manuf. Lett.* **2015**, *3*, 18–23. [CrossRef]
2. Song, H.; Rawat, D.; Jeschke, S.; Brecher, C. (Eds.) *Cyber-Physical Systems*; Academic Press: Cambridge, MA, USA, 2017.
3. Schuh, G.; Zeller, V.; Stroh, M.F.; Harder, P. Finding the Right Way Towards a CPS—A Methodology for Individually Selecting Development Processes for Cyber-Physical Systems. In *Collaborative Networks and Digital Transformation*; Camarinha-Matos, L.M., Afsarmanesh, H., Antonelli, D., Eds.; Springer International Publishing: Cham, Switzerland, 2019; pp. 81–90.
4. Lodgaard, E.; Dransfeld, S. Organizational aspects for successful integration of human-machine interaction in the Industry 4.0 era. *Procedia Cirp* **2020**, *88*, 218–222. [CrossRef]
5. Nardo, M.; Forino, D.; Murino, T. The evolution of man–machine interaction: The role of human in Industry 4.0 paradigm. *Prod. Manuf. Res.* **2020**, *8*, 20–34. [CrossRef]
6. Ardanza, A.; Moreno, A.; Segura, Á.; de la Cruz, M.; Aguinaga, D. Sustainable and flexible industrial human machine interfaces to support adaptable applications in the Industry 4.0 paradigm. *Int. J. Prod. Res.* **2019**, *57*, 4045–4059. [CrossRef]
7. Grangel-González, I.; Halilaj, L.; Coskun, G.; Auer, S.; Collarana, D.; Hoffmeister, M. Towards a Semantic Administrative Shell for Industry 4.0 Components. In Proceedings of the 2016 IEEE Tenth International Conference on Semantic Computing (ICSC), Laguna Hills, CA, USA, 4–6 February 2016; pp. 230–237. [CrossRef]
8. Batty, M. Digital twins. *Environ. Plan. B Urban Anal. City Sci.* **2018**, *45*, 817–820. [CrossRef]
9. El Saddik, A. Digital Twins: The Convergence of Multimedia Technologies. *IEEE Multimedia* **2018**, *25*, 87–92. MMUL.2018.023121167. [CrossRef]
10. Qi, Q.; Tao, F. Digital Twin and Big Data Towards Smart Manufacturing and Industry 4.0: 360 Degree Comparison. *IEEE Access* **2018**, *6*, 3585–3593. [CrossRef]
11. Ma, H.; Govoni, M.; Galli, G. Quantum simulations of materials on near-term quantum computers. *Npj Comput. Mater.* **2020**, *6*, 1–8. [CrossRef]
12. Villalba-Diez, J.; Zheng, X. Quantum Strategic Organizational Design: Alignment in Industry 4.0 Complex-Networked Cyber-Physical Lean Management Systems. *Sensors* **2020**, *20*, 5856. [CrossRef] [PubMed]
13. Bortolini, M.; Ferrari, E.; Gamberi, M.; Pilati, F.; Faccio, M. Assembly system design in the Industry 4.0 era: A general framework. *IFAC-PapersOnLine* **2017**, *50*, 5700–5705. [CrossRef]
14. Cohen, Y.; Naseraldin, H.; Chaudhuri, A.; Pilati, F. Assembly systems in Industry 4.0 era: A road map to understand Assembly 4.0. *Int. J. Adv. Manuf. Technol.* **2019**, *105*, 4037–4054. [CrossRef]
15. Deza, M.; Laurent, M. *Geometry of Cuts and Metrics*, 1st ed.; Algorithms and Combinatorics; Springer: Berlin/Heidelberg, Germany, 1997; Volume 15.
16. Benlic, U.; Hao, J.K. Breakout Local Search for the Max-Cutproblem. *Eng. Appl. Artif. Intell.* **2013**, *26*, 1162–1173. [CrossRef]
17. Vesselinova, N.; Steinert, R.; Perez-Ramirez, D.F.; Boman, M. Learning combinatorial optimization on graphs: A survey with applications to networking. *IEEE Access* **2020**, *8*, 120388–120416. [CrossRef]
18. Beloborodov, D.; Ulanov, A.E.; Foerster, J.N.; Whiteson, S.; Lvovsky, A. Reinforcement learning enhanced quantum-inspired algorithm for combinatorial optimization. *Mach. Learn. Sci. Technol.* **2020**, *2*, 025009. [CrossRef]
19. Gu, S.; Yang, Y. A Deep Learning Algorithm for the Max-Cut Problem Based on Pointer Network Structure with Supervised Learning and Reinforcement Learning Strategies. *Mathematics* **2020**, *8*, 298. [CrossRef]
20. Farhi, E.; Goldstone, J.; Gutmann, S.; Neven, H. Quantum Algorithms for Fixed Qubit Architectures. *arXiv* **2017**, arXiv:1703.06199.
21. Farhi, E.; Goldstone, J.; Gutmann, S. A Quantum Approximate Optimization Algorithm. *arXiv* **2014**, arXiv:1411.4028.
22. Guerreschi, G.G.; Matsuura, A.Y. QAOA for Max-Cut requires hundreds of qubits for quantum speed-up. *Sci. Rep.* **2019**, *9*, 6903. [CrossRef] [PubMed]
23. Villalba-Diez, J. Value Stream Network Quantum Approximate Optimization Algorithm. 2021. Available online: https://shorturl.at/hnpC3 (accessed on 1 December 2021).
24. Wille, R.; Meter, R.; Naveh, Y. IBM's Qiskit Tool Chain: Working with and Developing for Real Quantum Computers. In Proceedings of the 2019 Design, Automation Test in Europe Conference Exhibition (DATE), Florence, Italy, 25–29 March 2019; pp. 1234–1240. [CrossRef]
25. Jekel, C.; Venter, G.; Venter, M.; Stander, N.; Haftka, R. Similarity measures for identifying material parameters from hysteresis loops using inverse analysis. *Int. J. Mater. Form.* **2019**, *12*, 355–378. [CrossRef]
26. Fuchs, F.G.; Kolden, H.Ø.; Aase, N.H.; Sartor, G. Efficient Encoding of the Weighted MAX-k-CUT on a Quantum Computer Using QAOA. *SN Comput. Sci.* **2021**, *2*, 1–14. [CrossRef]