# Solving the Optimization Problem of Bidirectional EV-Charging on a Quantum Computer

Daniel Müssig[1], Marika Federer[2], Steve Lenk[2], Jörg Lässig[1,3]

1 Fraunhofer IOSB-AST, Wilhemsplatz 11, 02826 Görlitz
2 Fraunhofer IOSB-AST, Am Vogelherd 90, 98693 Ilmenau
3 Hochschule Zittau/Görlitz, Brückenstraße 1, 02826 Görlitz

**Abstract**

The optimization problem of bidirectional electrical vehicle charging (Vehicle-to-Home) becomes more and more important with rising energy prices and the required reduction of $CO_2$ emissions. We maximize the usage of local solar power generation, while minimizing the power grid usage. This is constrained by the energy demand of the household and the required state of charge at departure as well as the idle times of the car at home. The problem is formulated as a quadratic unconstrained binary optimization (QUBO) problem, which can be solved with the hybrid quantum-classical algorithm quantum alternating operator ansatz (QAOA). We show results for small problem instances executed on a real device and a simulator. Additionally, we approximate the requirements of the problem on quantum computers for problem instances which are more realistic.

## 1. Introduction

Quantum Computing promises an advantage over conventional computing in solving many NP-hard problems faster, without changing their complexity class. The current generation of quantum computers is small in the number of qubits and noisy, and thus called near-term intermediate scale quantum computer (NISQ). Further, to make use of current generation quantum computers most of the programs are written on a very low abstraction level i.e., using a quantum circuit consisting of quantum gates. A gate can act on one or more qubits, and is a matrix, which is matrix multiplied onto a quantum state i.e., a vector. Multiple qubits are combined by the tensor product to a quantum register. A qubit can be measured in the state $|0\rangle = \binom{1}{0}$ or $|1\rangle = \binom{0}{1}$, but could be without observation i.e., before measuring, in any state in between, which preserves the length of the vector. The most known gates are the Pauli gates $X, Y$ and $Z$, which implement the Pauli matrices $\sigma^x, \sigma^y$ and $\sigma^z$, the Hadamard gate $H$ and the controlled not gate $CNOT$. Pauli gates are rotations of $\pi$ radians around the Bloch sphere. The Hadamard gate transforms the state $|0\rangle$ into the equal superposition. Lastly, the $CNOT$ gate is a two-qubit gate, which leaves the targeted qubit in its state if the controlled qubit is in state $|0\rangle$ and does a bitflip on the targeted qubit if the controlled qubit is in state $|1\rangle$ (Nielsen & Chuang, 2010).

Hybrid algorithms are used to utilize current generations of quantum computers. They consist of a shallow parameterized quantum circuit, which is run on a quantum computer, while the parameters are optimized on a classical machine. This process is repeated until an optimal parameter set is found, as illustrated in Figure 1 (McClean et al.,

2016). One of those hybrid algorithms is the quantum alternating operator ansatz (QAOA), which we will use in this paper to solve the optimization problem of bidirectional EV-charging on a quantum computer.

The following of the paper is organized as follows: In Section 2 we present related work on the topic. Section 3 describes the problem, which we want to solve on the quantum computer. The 4th Section formulates the QUBO to solve the problem and describes the implementation in QAOA. In Section 5 we show our results of running the algorithm both on the simulator and on a real device. We conclude in Section 6 and give an outlook on further work.

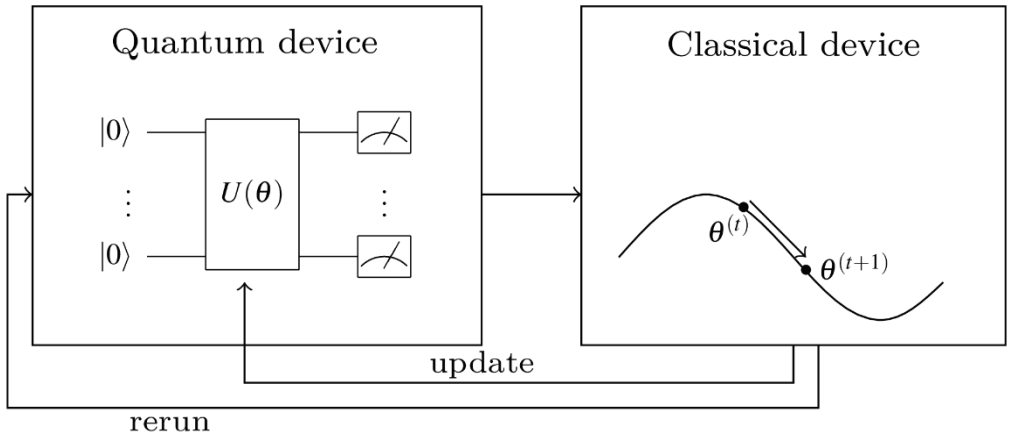


*Figure 1: Hybrid Algorithm adopted from (Schuld & Petruccione, 2018).*

## 2. Related Work

When we want to solve optimization problems on a current generation quantum computer, we have overall two options. On one hand we could use the variational quantum eigensolver (VQE) and on the other hand the quantum alternating operator ansatz (QAOA). However, VQE is more suited for optimization problems with quantum variables i.e., solving optimization problems in the quantum world, and QAOA is more suited towards classical optimization problems (Moll et al., 2018).

A good example for industrial usage of QAOA is the paper by Volkswagen, solving the binary paint shop problem (Streif et al., 2020). Here, the authors showed, how an industrial relevant problem could be mapped to an ising Hamiltonian, which was approximated by QAOA on a quantum computer. Further, they showed improvements against a classical heuristic i.e., the greedy algorithm.

In a previous work, we have optimized the charging schedule of battery electric service vehicles of the Erfurt airport on a quantum computer (Federer, Müssig, Klaiber, et al., 2022). We've modeled the problem as a QUBO and solved it using QAOA. However, in this industrial use case we haven't considered a bidirectional setting.

In a follow-up paper we have analyzed the results and investigated the complexity as well as the landscape of the cost function (Federer, Müssig, Lenk, et al., 2022). We have found that the introduction of constraints increases the complexity of the landscape. Further, we have seen symmetries in the landscape of the cost function.

The work by (Deller et al., 2022) also optimizes the charging schedule of electrical vehicles using QAOA. However, instead of a direct mapping of the problem to a QUBO, the authors decided to map the problem to a graph coloring problem. Further, ideas were discussed on implementing constraints as conditional gates on ancilla qubits and by using a dynamical decoupling strategy. Lastly, the QAOA circuits were implemented with qudits instead of qubits, which enabled the authors to use integer instead of binary variables.

## 3. Bidirectional EV-Charging

In this paper we solve a very specific problem for bidirectional EV charging. First, we consider a vehicle to home (V2H) use case, where the car discharges only into the home and not into the grid. Further, we want to maximize the local usage of the solar power generation. Therefore, we use the battery of the electrical vehicle also as a battery of the house. We always want to fulfill the minimum charging amount $e_{min}$ and not exceed the maximum charging amount $e_{max}$. For each timestep $t$ we have a maximum charging power $j_{max,t}$ and a maximum discharging power $j_{min,t}$. Further, we have the solar power generation $pv_t$ and the power demand of the house $d_t$ with their difference $p_t$. We minimize the cost function over the charging power $j_t^c$ and discharging power $j_t^d$.

To solve this problem, we use Equation (1), which is a quadratic constrained integer optimization (QCIO) problem.

$$
\begin{aligned}
&\boldsymbol{C\left(j_t^c, j_t^d\right) = min \sum_t \left(\left(j_t^c - j_t^d\right) - p_t\right)^2} \\
&\boldsymbol{subject\ to}: \\
&\boldsymbol{j_{min,t} \geq j_t^d} \\
&\boldsymbol{j_{max,t} \geq j_t^c} \\
&\boldsymbol{e_{min} \leq \sum_t j_t^c - j_t^d \leq e_{max}} \\
&\boldsymbol{j_t^c j_t^d = 0}
\end{aligned}
\tag{1}
$$

We have the following constraints from top to bottom: we are not allowed to discharge more than our maximum discharging power, we are not allowed to charge more than our maximum charging power, overall, we must fulfill the minimum charging amount and shall not exceed the maximum charging amount, and we are not allowed to charge and discharge at the same time step.

We decided to formulate the problem as a QCIO problem, because it can be translated into a quadratic unconstrained binary optimization problem (QUBO), which can be solved with QAOA.

## 4. Formulation of the QUBO and Implementation of QAOA

The quantum alternating operator ansatz (QAOA) formerly known as quantum approximate optimization algorithm approximates the optimal solution of combinatorial optimization problems by mapping them to find the ground state of an Ising Hamiltonian (Weidenfeller et al., 2022).

QUBOs have a close relation to Ising Hamiltonians. Thus, we start with a reformulation of our QCIO into a QUBO. In Equation (2) we introduce three variables, which hold the

required bits for $j_t^c$, $j_t^d$ and $q$ respectively. For each time step and each variable, we calculate the minimum required number of bits, to utilize the least number of qubits on a quantum computer. Further, if we have a constraint, where a number must be zero, we set it zero and do not use a qubit to store this information.

$$\begin{aligned} \boldsymbol{b_t} &= \lceil \boldsymbol{log_2}(\boldsymbol{j_{max_t}} + \boldsymbol{1}) \rceil \\ \boldsymbol{c_t} &= \lceil \boldsymbol{log_2}(\boldsymbol{j_{min,t}} + \boldsymbol{1}) \rceil \\ \boldsymbol{f} &= \lceil \boldsymbol{log_2}(\boldsymbol{e_{max}} - \boldsymbol{e_{min}} + \boldsymbol{1}) \rceil \end{aligned} \tag{2}$$

Equation (3) and (4) calculate the charging and discharging power for each time step. The second summand reflects the integer value represented by the binary variables $x_i$ at $j_{max}$ and $y_i$ at $j_{min}$ respectively. This assures constraint 1 and 2 in Equation (1).

$$\boldsymbol{j_t^c} = \sum_{\boldsymbol{i=0}}^{\boldsymbol{b_t-2}} \boldsymbol{2^i x_{i,t}} + \left(\boldsymbol{j_{max,t}} - \boldsymbol{2^{b_t-1}} + \boldsymbol{1}\right)\boldsymbol{x_{b_t-1,t}} \tag{3}$$

$$\boldsymbol{j_t^d} = \sum_{\boldsymbol{i=0}}^{\boldsymbol{c_t-2}} \boldsymbol{2^i y_{i,t}} + \left(\boldsymbol{j_{min,t}} - \boldsymbol{2^{c_t-1}} + \boldsymbol{1}\right)\boldsymbol{y_{c_t-1,t}} \tag{4}$$

Equation (5) introduces a slack variable, which is later used in the second penalty term.

$$\boldsymbol{q} = \sum_{\boldsymbol{i=0}}^{\boldsymbol{f-2}} \boldsymbol{2^i z_i} + (\boldsymbol{e_{max}} - \boldsymbol{e_{min}} + \boldsymbol{1}) \tag{5}$$

In the following Equation (6) introduces a QUBO version of Equation (1). The first part still looks the same as the sum in Equation (1). However, we have used the bionomic formula to prepare it already for the Ising formulation. $P_1$ and $P_2$ are penalty factors, where the corresponding penalty terms help us removing the constraints. In particular, the first penalty term is a reformulation of constraint 4 and the second is a reformulation of constraint 3.

$$\begin{aligned} \boldsymbol{C(x,y,z)} = \mathbf{min} \sum_{\boldsymbol{t}} \left[\boldsymbol{j_t^{c2}} + \boldsymbol{j_t^{d2}} - \boldsymbol{2j_t^c p_t} + \boldsymbol{2j_t^d p_t} + \boldsymbol{p_t^2}\right] + \boldsymbol{P_1}\left(\sum_{\boldsymbol{t}} \boldsymbol{j_t^c j_t^d}\right) \\ + \boldsymbol{P_2}\left(\sum_{\boldsymbol{t}} \left(\boldsymbol{j_t^c} - \boldsymbol{j_t^d}\right) - \boldsymbol{q} - \boldsymbol{e_{min}}\right)^2 \end{aligned} \tag{6}$$

To encode our QUBO into an Ising Hamiltonian we shift our binary variables $x_i$, $y_i$ and $z_i$ to $\frac{1-zx_i}{2}$, $\frac{1-zy_i}{2}$ and $\frac{1-zz_i}{2}$ with $zx_i, zy_i, zz_i \in \{-1,1\}$. Afterwards, we substitute these variables with $\sigma^z$ (Moll et al., 2018). The Pauli-$Z$ operator is used since it has the eigenvectors $|0\rangle$ and $|1\rangle$ with the corresponding eigenvalues $1$ and $-1$.

To find the ground energy of an Ising Hamiltonian $H_C$ with QAOA we also need a mixing Hamiltonian $H_B$. A typical mixing Hamiltonian is $H_B = \sum_{j=1}^{N} \sigma_j^x$ since we can easily prepare its ground state, which is the equal superposition of all states. Thus, the QAOA is initialized with Hadmard gates on each qubit. A $p$-level QAOA circuit consists of $H_C$ and $H_B$ alternating $p$ times to approximate adiabatic quantum annealing via Trotterization. Therefore, we parameterize each Hamiltonian. The convention is to use $\gamma$ for the Ising

or problem Hamiltonian ranging from 0 to $2\pi$ and $\beta$ for the mixing Hamiltonian ranging from 0 to $\pi$. A formal description of QAOA would be $U = U(H_B, \beta_p) \cdot U(H_C, \gamma_p) \cdot \ldots \cdot U(H_B, \beta_1) \cdot U(H_C, \gamma_1)$ and is shown in Figure 2 (Zhou et al., 2020).

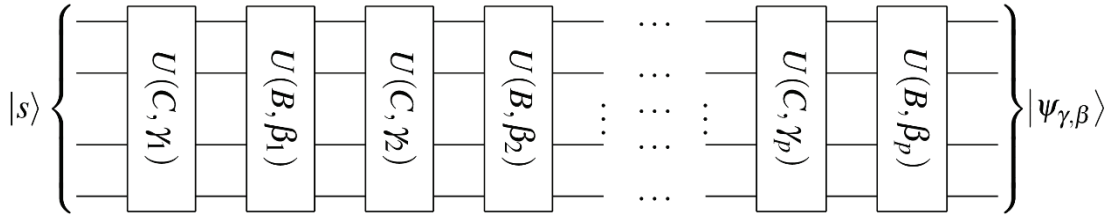


*Figure 2: A p-level QAOA with alternating $U_C$ und $U_B$.*

Since we have now constructed the QAOA circuit, the next step is to optimize the parameters to find the ground state of the problem Hamiltonian. Therefore, we calculate the expectation value $F_p(\gamma, \beta) = \langle \psi_p(\gamma, \beta) | H_C | \psi_p(\gamma, \beta) \rangle$ in each iteration for the current best parameters. In our classical optimization loop we use a classical nonlinear optimizer, which does not require the derivative of the cost function like *COBYLA* or *SPSA*, to find the best parameter set regarding the smallest expectation value.

## 5. Results

We have implemented our QAOA for bidirectional charging with the Qiskit Runtime program for QAOA with the classical optimizer *SPSA*, since it can be executed as one job within the IBM Quantum Experience and does not require a single job for each iteration (Weidenfeller et al., 2022). The advantage of having only one job, which must be executed, is, that we only need to be once in the queue for each experiment. Waiting times in queues for today's public cloud quantum computers range from a couple of minutes to sometimes even days. Thus, it is currently not viable to run a hybrid quantum algorithm without a reservation of the quantum computer.

Our experiments consist of five problem instances, which were solved with IBMs CPLEX solver, which gives an optimal result, with the IBM QASM Simulator, which is a noiseless simulator and on IBMQ Ehningen, which is a real 27-qubit machine deployed in Germany. On the two devices we used a two-layer QAOA algorithm. The results are shown in Table 1. We have listed in the first column the data of the experiments. The second column lists the results for the CPLEX solver, which is an exact solution and was always the same result in multiple runs. Columns three and four list the results on the IBM QASM Simulator and on IBMQ Ehningen respectively. We have done multiple runs on both devices and used 1024 shots, which means, that the circuit was run 1024 times to establish an expectation value. From the 1024 shots we have taken the best solution and calculated the average probability within the shots over all runs.

The first experiment requires two qubits. There is only one feasible solution, which was found on all three solvers. The achieved probability for the best state was higher than equal superposition, for both the simulator and the calculations on the real quantum device, which would have led to a probability of $\sim 25\%$. Experiment 2 requires four qubits. Here, the probability distribution for the equal superposition would have been $\sim 3.125\%$ and again we see better results on both devices. However, we also see, that the QASM Simulator has a lower probability distribution. Experiments three, four and five required seven qubits. We have chosen this as the maximum number of qubits for

our experiments, because we had intended to use a free 7-qubit machine. However, since the queue times were to long, we switched to the paid IBMQ Ehningen machine. In the last three experiments the probability distribution for an equal superposition would have been $\sim 0.781\%$. We see that the results are better or near that value.

| | optimal result | QASM Simulator | IBMQ Ehningen |
|---|---|---|---|
| $\boldsymbol{p} = [\mathbf{0}, \mathbf{1}]$ | $j^c = [0,0]$ | $j^c = [0,0]$ | $j^c = [0,0]$ |
| $\boldsymbol{j_{min}} = [\mathbf{0}, \mathbf{0}]$ | $j^d = [0,0]$ | $j^d = [0,0]$ | $j^d = [0,0]$ |
| $\boldsymbol{j_{max}} = [\mathbf{1}, \mathbf{0}]$ | $q = 0$ | $q = 0$ | $q = 0$ |
| $\boldsymbol{e_{min}} = \mathbf{0}$ | $C(x,y,z) = 1$ | $C(x,y,z) = 1$ | $C(x,y,z) = 1$ |
| $\boldsymbol{e_{max}} = \mathbf{1}$ | | $Prob. = \sim 51{,}85\%$ | $Prob. = \sim 44.49\%$ |
| $\boldsymbol{p} = [\mathbf{1}, -\mathbf{1}, \mathbf{1}]$ | $j^c = [1,0,0]$ | $j^c = [1,0,0]$ | $j^c = [1,0,0]$ |
| $\boldsymbol{j_{min}} = [\mathbf{0}, \mathbf{0}, \mathbf{0}]$ | $j^d = [0,0,0]$ | $j^d = [0,0,0]$ | $j^d = [0,0,0]$ |
| $\boldsymbol{j_{max}} = [\mathbf{1}, \mathbf{1}, \mathbf{1}]$ | $q = 0$ | $q = 0$ | $q = 0$ |
| $\boldsymbol{e_{min}} = \mathbf{1}$ | $C(x,y,z) = 2$ | $C(x,y,z) = 2$ | $C(x,y,z) = 2$ |
| $\boldsymbol{e_{max}} = \mathbf{1}$ | | $Prob. = \sim 7{,}12\%$ | $Prob. = \sim 13{,}60\%$ |
| $\boldsymbol{p} = [\mathbf{1}, -\mathbf{1}, \mathbf{1}]$ | $j^c = [1,0,1]$ | $j^c = [1,0,1]$ | $j^c = [1,0,1]$ |
| $\boldsymbol{j_{min}} = [\mathbf{1}, \mathbf{1}, \mathbf{0}]$ | $j^d = [0,1,0]$ | $j^d = [0,1,0]$ | $j^d = [0,1,0]$ |
| $\boldsymbol{j_{max}} = [\mathbf{1}, \mathbf{1}, \mathbf{1}]$ | $q = 0$ | $q = 0$ | $q = 0$ |
| $\boldsymbol{e_{min}} = \mathbf{1}$ | $C(x,y,z) = 0$ | $C(x,y,z) = 0$ | $C(x,y,z) = 0$ |
| $\boldsymbol{e_{max}} = \mathbf{2}$ | | $Prob. = \sim 1{,}07\%$ | $Prob. = \sim 2{,}08\%$ |
| $\boldsymbol{p} = [\mathbf{1}, -\mathbf{1}, \mathbf{1}, -\mathbf{1}]$ | $j^c = [1,0,1,0]$ | $j^c = [1,0,1,0]$ | $j^c = [1,0,1,0]$ |
| $\boldsymbol{j_{min}} = [\mathbf{0}, \mathbf{1}, \mathbf{0}, \mathbf{1}]$ | $j^d = [0,0,0,1]$ | $j^d = [0,0,0,1]$ | $j^d = [0,0,0,1]$ |
| $\boldsymbol{j_{max}} = [\mathbf{1}, \mathbf{1}, \mathbf{1}, \mathbf{0}]$ | $q = 0$ | $q = 0$ | $q = 0$ |
| $\boldsymbol{e_{min}} = \mathbf{1}$ | $C(x,y,z) = 1$ | $C(x,y,z) = 1$ | $C(x,y,z) = 1$ |
| $\boldsymbol{e_{max}} = \mathbf{2}$ | | $Prob. = \sim 1{,}07\%$ | $Prob. = \sim 1{,}59\%$ |
| $\boldsymbol{p} = [\mathbf{1}, -\mathbf{1}, \mathbf{3}]$ | $j^c = [1,0,3]$ | $j^c = [1,0,3]$ | $j^c = [1,0,3]$ |
| $\boldsymbol{j_{min}} = [\mathbf{1}, \mathbf{1}, \mathbf{0}]$ | $j^d = [0,1,0]$ | $j^d = [0,1,0]$ | $j^d = [0,1,0]$ |
| $\boldsymbol{j_{max}} = [\mathbf{1}, \mathbf{1}, \mathbf{3}]$ | $q = 1$ | $q = 1$ | $q = 1$ |
| $\boldsymbol{e_{min}} = \mathbf{2}$ | $C(x,y,z) = 0$ | $C(x,y,z) = 0$ | $C(x,y,z) = 0$ |
| $\boldsymbol{e_{max}} = \mathbf{3}$ | | $Prob. = \sim 1{,}07\%$ | $Prob. = \sim 0{,}65\%$ |

*Table 1: Results of our five experiments of the bidirectional EV charging QAOA algorithm.*

Overall, the results show, that we have found the optimal solution on both devices for all five experiments. However, we can also see, that the probabilities of finding the optimal solution within the 1024 shots in the last three experiments is very low. Further, only for the first experiment the optimal solution was also the most frequent. The low probabilities in the noiseless QASM Simulator suggests that the settings of the QAOA approach and its optimizer needs to be optimized and that our use case is hard-to-solve due to the global constraints involved.

## 6. Conclusion and Future Work

In this paper we have formulated a QUBO for the optimization problem of bidirectional EV charging. The QUBO was solved with a two-layer QAOA algorithm on a QASM Simulator and on the real device IBMQ Ehningen, where we found the optimal solution for all five experiments.

One of the next steps will be a different implementation of the penalty terms e.g., with dynamic decoupling approaches (Deller et al., 2022). From this we expect to reduce the complexity of the QUBO and to achieve higher probabilities for the optimal solution.

Further, a next step would be the usage of qudits instead of qubits. With this we would not need to convert the integers first into binaries. Also, an implementation of the QUBO and the problem instances in QBench (Müssig, Daniel & Lässig, Jörg, 2022) is planned. This will help us understand better the differences between the noiseless QASM Simulator and the real devices. Further, we could investigate device specific influences. Lastly, it would be also interesting to compare results from the superconducting hardware of IBM to other technologies and architectures like IonQs trapped ions or XANADUs photonics. Nevertheless, a comparison with D-Waves quantum annealer would be also from great interest.

**Acknowledgements**

Funded by the European Union under Horizon Europe Programme - Grant Agreement 101080086 — NeQST.

Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or European Climate, Infrastructure and Environment Executive Agency (CINEA). Neither the European Union nor the granting authority can be held responsible for them.

This work was financially supported by the Bundesministerium für Wirtschaft und Energie through the project “EnerQuant” (Project- ID 03EI1025C) and by Thüringer Aufbaubank through the project “Quantum Hub Thüringen”.

**Literature**

Deller, Y., Schmitt, S., Lewenstein, M., Lenk, S., Federer, M., Jendrzejewski, F., Hauke, P., & Kasper, V. (2022). Quantum approximate optimization algorithm for qudit systems with long-range interactions. *ArXiv:2204.00340 [Cond-Mat, Physics:Quant-Ph]*. http://arxiv.org/abs/2204.00340

Federer, M., Müssig, D., Klaiber, S., Lässig, J., Bretschneider, P., & Lenk, S. (2022). Application benchmark for quantum optimization on electromobility use case. *2022 IEEE Vehicle Power and Propulsion Conference (VPPC)*, 1–6. https://doi.org/10.1109/VPPC55846.2022.10003292

Federer, M., Müssig, D., Lenk, S., & Lässig, J. (2022). *Real-world application benchmark for QAOA algorithm for an electromobility use case*. Gesellschaft für Informatik, Bonn. https://doi.org/10.18420/inf2022_97

McClean, J. R., Romero, J., Babbush, R., & Aspuru-Guzik, A. (2016). The theory of variational hybrid quantum-classical algorithms. *New Journal of Physics*, *18*(2), 023023. https://doi.org/10.1088/1367-2630/18/2/023023

Moll, N., Barkoutsos, P., Bishop, L. S., Chow, J. M., Cross, A., Egger, D. J., Filipp, S., Fuhrer, A., Gambetta, J. M., Ganzhorn, M., Kandala, A., Mezzacapo, A., Müller, P., Riess, W., Salis, G., Smolin, J., Tavernelli, I., & Temme, K. (2018). Quantum optimization using variational algorithms on near-term quantum devices. *Quantum Science and Technology*, *3*(3), 030503. https://doi.org/10.1088/2058-9565/aab822

Müssig, Daniel, & Lässig, Jörg. (2022). *Quantum Computer and Quantum Algorithm Benchmarking*. https://doi.org/10.18420/INF2022_100

Nielsen, M. A., & Chuang, I. L. (2010). *Quantum computation and quantum information* (10th anniversary ed). Cambridge University Press.

Schuld, M., & Petruccione, F. (2018). *Supervised Learning with Quantum Computers*. Springer International Publishing. https://doi.org/10.1007/978-3-319-96424-9

Streif, M., Yarkoni, S., Skolik, A., Neukart, F., & Leib, M. (2020). *Beating classical heuristics for the binary paint shop problem with the quantum approximate optimization algorithm*. https://arxiv.org/pdf/2011.03403

Weidenfeller, J., Valor, L. C., Gacon, J., Tornow, C., Bello, L., Woerner, S., & Egger, D. J. (2022). Scaling of the quantum approximate optimization algorithm on superconducting qubit based hardware. *ArXiv:2202.03459 [Quant-Ph]*. http://arxiv.org/abs/2202.03459

Zhou, L., Wang, S.-T., Choi, S., Pichler, H., & Lukin, M. D. (2020). Quantum Approximate Optimization Algorithm: Performance, Mechanism, and Implementation on Near-Term Devices. *Physical Review X*, *10*(2), 021067. https://doi.org/10.1103/PhysRevX.10.021067