Variational Quantum Algorithms

The phrase variational quantum algorithms names a family of hybrid methods that combine a quantum computer with a classical computer. In the parent document, this phrase appears at the beginning of the book’s main topic: optimization methods that use quantum hardware without assuming that today’s quantum devices are perfect. That placement is important. A variational quantum algorithm, or VQA, is not simply “a quantum algorithm” in the older sense of a fixed sequence of quantum gates that runs once and produces an answer. It is usually an iterative loop: prepare a quantum state, measure something about it, use a classical computer to adjust parameters, and try again.

The word variational is the key. It means that we choose a flexible family of possible candidates, described by adjustable parameters, and then vary those parameters to improve a cost or objective value. This idea is older than quantum computing. In ordinary curve fitting, for example, we might choose all lines of the form

$$y = ax + b,$$

then vary the numbers a and b until the line fits the data well. In a variational quantum algorithm, the candidate is not a line. It is a quantum state prepared by a circuit whose gates contain adjustable numbers, often rotation angles. The algorithm varies those numbers in search of a better state or a better solution.

The basic loop

A VQA usually begins with a parameterized quantum circuit. This is a quantum circuit whose behavior depends on a list of classical parameters, often written as

$$\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_m).$$

Here, boldsymbolθ is just a vector of adjustable numbers. When the quantum computer runs the circuit with these parameters, it prepares a quantum state, often written

$$|\psi(\boldsymbol{\theta})\rangle.$$

This notation means “the quantum state prepared when the circuit uses parameters boldsymbolθ.” The state depends on the parameters, just as the line y=ax+b depends on a and b.

The algorithm then measures the quantum state to estimate a quantity of interest. In optimization, this quantity is usually a cost function. If the cost is written as

$$C(\text{boldsymbol}\theta),$$

then the goal might be to find parameters that make C(boldsymbolθ) as small as possible. A classical optimizer receives the estimated cost and proposes new parameters. The quantum computer tries those new parameters, measurements produce a new estimate, and the loop continues.

So the heart of a VQA is not one machine replacing the other. It is cooperation. The quantum processor prepares and samples from quantum states. The classical processor stores parameters, performs arithmetic, chooses updates, and decides when to stop.

Why measurement makes the loop statistical

One subtle point is that a quantum computer usually does not reveal the full quantum state directly. When we measure, we get samples. If a circuit is measured once, it gives one outcome, such as a bit string 01011. To estimate an average cost reliably, the same circuit must usually be run many times. These repeated runs are often called shots.

For example, suppose a quantum circuit is being used to search over binary choices, such as selecting or not selecting projects. A measurement may return one candidate choice. After many measurements, we can estimate whether the current parameters tend to produce good choices. The estimate has statistical uncertainty: with too few shots, the measured cost may be noisy simply because of sampling variation. This is one reason VQAs are practical but also difficult. The classical optimizer is not given perfect information; it is given estimates affected by both measurement statistics and hardware noise.

This point supports the parent document’s careful tone. VQAs are promising, but they are not magic. Their performance depends on circuit design, number of measurements, optimizer choice, noise level, and the quality of the problem encoding.

Why VQAs matter in the NISQ era

VQAs became especially prominent because they appear suitable for noisy intermediate-scale quantum devices, often called NISQ devices. NISQ machines have enough qubits to be scientifically interesting, but they are noisy and not fully error-corrected [Preskill 2018]. Long quantum circuits usually accumulate too much error on such hardware. VQAs try to work with shorter circuits and let the classical computer handle much of the adaptive decision-making.

This does not guarantee a practical advantage. A short circuit may be easier to run, but it may also be too limited to represent the best solution. A classical optimizer may improve parameters, but it can become stuck in poor regions of the search landscape. Noise can blur the cost function and make good parameters harder to identify. Reviews of the field emphasize both the potential of VQAs and the major challenges of trainability, noise, and benchmarking [Cerezo et al. 2021].

The parent document's description is therefore supported by the standard understanding of VQAs: they are hybrid algorithms, they use parameterized quantum circuits, and they rely on classical optimization. What remains open, and must be checked problem by problem, is whether a given VQA provides better solution quality, speed, robustness, or insight than strong classical methods.

VQE and QAOA as examples

Two famous examples help make the phrase concrete.

The Variational Quantum Eigensolver, or VQE, was developed to estimate low-energy states of quantum systems, especially in quantum chemistry. In simplified terms, VQE prepares a trial quantum state and estimates its energy. The classical optimizer changes the parameters to reduce that energy. The method is "variational" because, according to the variational principle in quantum mechanics, the energy expectation value of a trial state gives an upper bound on the true ground-state energy, assuming the Hamiltonian and state preparation are treated correctly. The early demonstration by Peruzzo and collaborators was influential because it showed a hybrid quantum-classical workflow on photonic hardware [Peruzzo et al. 2014].

The Quantum Approximate Optimization Algorithm, or QAOA, is more directly tied to combinatorial optimization. It was proposed for problems such as MaxCut, where the goal is to divide a graph's vertices into two groups while maximizing the number of edges crossing between the groups [Farhi et al. 2014]. QAOA uses a structured parameterized circuit built from alternating operations: some related to the problem's objective and others that mix possible solutions. The parameters control how strongly and for how long these operations are applied. After measurement, the bit strings are interpreted as candidate solutions.

Both VQE and QAOA fit the same broad pattern: choose a parameterized quantum circuit, measure a cost-related quantity, and let a classical optimizer update the parameters.

The optimization view

From the perspective of the parent document, a VQA can be seen as an optimization method nested inside another optimization problem. The original task might be "find the best route," "choose the best portfolio," or "minimize molecular energy." To use a VQA, we create a parameterized quantum procedure and then solve a new optimization problem over the parameters:

min_boldsymbolθ C(boldsymbolθ).

Here, C(boldsymbolθ) is not usually available as a simple formula. Instead, it is estimated by running quantum circuits and measuring outcomes. This makes VQA optimization different from ordinary textbook optimization, where the objective function may be exactly computable. In a VQA, each evaluation of the objective can be expensive, noisy, and uncertain.

This also explains why classical computation remains central. The quantum device does not usually "find the minimum" by itself. It provides information about a parameterized quantum state. The classical optimizer decides how to use that information. Different optimizers may behave very differently, especially when the cost estimates are noisy.

What should be verified when reading about VQAs

When the parent document introduces "variational quantum algorithms," the basic definition is sound: VQAs are hybrid quantum-classical methods using adjustable quantum circuits and classical parameter updates. But later claims about their usefulness require more careful verification.

A responsible reader should ask what problem is being encoded, how many qubits and gates are required, how the cost is estimated, how many measurements are needed, and which classical baseline is used. If a study compares a VQA only against a weak classical method, the result may not show meaningful quantum advantage. If a circuit works only for tiny examples, it may still be valuable as a proof of concept, but not as evidence of commercial usefulness. If the optimization landscape contains barren plateaus—regions where gradients become extremely small—training may become difficult for large systems [McClean et al. 2018].

Thus, “variational quantum algorithms” should be understood as a flexible research framework rather than a single guaranteed speedup. Their strength is that they give a practical way to explore quantum computation on imperfect hardware. Their weakness is that every part of the loop—encoding, circuit ansatz, measurement, noise, and classical optimization—can become a bottleneck.

References

Cerezo, M., Arrasmith, A., Babbush, R., Benjamin, S. C., Endo, S., Fujii, K., McClean, J. R., Mitarai, K., Yuan, X., Cincio, L., & Coles, P. J. (2021). Variational quantum algorithms. Nature Reviews Physics, 3, 625–644. https://doi.org/10.1038/s42254-021-00348-9

Farhi, E., Goldstone, J., & Gutmann, S. (2014). A quantum approximate optimization algorithm. arXiv:1411.4028. https://arxiv.org/abs/1411.4028

McClean, J. R., Boixo, S., Smelyanskiy, V. N., Babbush, R., & Neven, H. (2018). Barren plateaus in quantum neural network training landscapes. Nature Communications, 9, Article 4812. https://doi.org/10.1038/s41467-018-07090-4

Peruzzo, A., McClean, J., Shadbolt, P., Yung, M.-H., Zhou, X.-Q., Love, P. J., Aspuru-Guzik, A., & O’Brien, J. L. (2014). A variational eigenvalue solver on a photonic quantum processor. Nature Communications, 5, Article 4213. https://doi.org/10.1038/ncomms5213

Preskill, J. (2018). Quantum computing in the NISQ era and beyond. Quantum, 2, Article 79. https://doi.org/10.22331/q-2018-08-06-79

Document information

variational quantum algorithms

Project	Variational Quantum Algorithms for Optimization
Document	Document 1.4.1
Author	phone
Verifier	Not verified
Downloaded	July 03, 2026 17:56 KST
Status	Working
Document link	https://theorytrace.com/projects/variational-quantum-algorithms-for-optimization/documents/variational-quantum-algorithms/