White Paper – Quantum Approximate Optimisation Algorithm

Mark Hodson

What is the Quantum Approximate Optimisation Algorithm and how can it be used for quantum algorithm development? This white paper provides a technical audience with the answers.

Introduction

This white paper provides a brief to a technical audience on the Quantum Approximate Optimisation Algorithm and its relevance to the development of quantum-accelerated applications atop emerging universal quantum computing hardware.

What is it?

The Quantum Approximate Optimisation Algorithm (QAOA) [3, 6] is a quantum algorithm that **approximately solves** the **Maximum Satisfiability Problem** (MAX-SAT). This is the problem of determining the maximum number of clauses of a given Boolean formula that can be made true by an assignment of truth values to the variables of the formula. By approximate, we mean that it "satisfies a high fraction of the maximum number of clauses that can be satisfied" [1].

Maximum satisfiability is particularly useful when there may not be a solution that satisfies all constraints. In its manifestation in QAOA, weights or values may additionally be applied to each clause satisfaction. This is a key algorithm control for engineering real-world applications.

The technique underpinning QAOA is one whose execution on a universal quantum computer may result in very shallow circuit depths. This makes the technique additionally attractive as a candidate for small-scale, near-term hardware, without the resource requirements for the manufacture of error correction codes.

Applications

QAOA has been successfully demonstrated to apply to graph partitioning problems such as MAX-CUT [5], number partitioning problems, and has the potential to apply to a range of NP-complete [5] and NP-hard problems.

The set of problems naturally able to be solved by QAOA includes decision problems, constraint satisfaction problems, and graph problems. These fundamental problem types have broad applicability to industry. Examples of problems that may be tractable using QAOA are:

- Automotive: Constraint problems in car configuration fit-out optimisation.
- Electronics: Constraint problems in circuit layout optimisation.
- Finance: Graph problems in arbitrage, customer behaviour, portfolio optimisation.
- Logistics: Graph problems in distribution optimisation, set packing for containers.
- Pharmaceutical: Graph problems in bioinformatics, gene matching, drug discovery.
- Software: Graph problems in formal verification of program control flow.

There exists an additional body of knowledge on the utility of current classical MAX-SAT solvers as an optimisation technique for some data analysis and machine learning problems [1]. These include techniques such as cost-optimal correlation clustering, and causal structure learning.

Community

QAOA is of interest to the quantum algorithm development community for two reasons.

First, a specific albeit contrived maximum satisfiability problem called MAX-E3LIN2 has been designed with the intent of executing it on early-stage quantum hardware and experimentally verifying quantum speed-up, and thus *quantum supremacy*. This problem has been built on top of QAOA and has been treated rigorously from a quantum information theory perspective. This effort is being led by Google, with initial results expected in 2018.

Second, this algorithm can act as a bridge between Adiabatic Quantum Optimisation (AQO) and Universal Quantum Computation (UQC). Specifically, AQO problems able to be described only in terms of constraints may be re-engineered to use QAOA. The most well-known existing quantum hardware for solving AQO problems is the quantum annealer from D-Wave Systems, whose potential for quantum speed-up remains a matter of conjecture. The key differentiator is the theoretical implication that QAOA's use of a non-stoquastic driver Hamiltonian leads to the potential for speed-up, where D-Wave's use of a stoquastic driver Hamiltonian does not [4].

Technique

Mathematically, QAOA can approximately solve any equation that can be reduced to the form

$$C(z) = \text{argmax}_z \sum_{\alpha=1}^{m} C_\alpha(z),\ \ z \in \{0,1\}^n$$

where

$$C_\alpha : \{0,1\}^n \rightarrow \{0,1\}$$

for Boolean constraints, where the value of 1 implies the constraint is met, or

$$C_\alpha : \{0,1\}^n \rightarrow \omega_\alpha,\ \ \omega_\alpha \in \Re$$

for weighted constraints, where the greater "ω_α" the more valuable the constraint.

The total number of solution variables "*n*" defines the problem size. The number of constraints is "*m*". Each constraint can involve any number of solution variables and this can affect the efficiency of implementation and depends on the problem formulation. The cost function "*C(z)*" is approximately maximised by QAOA.

The algorithm itself consists of two basic loops, depicted in Figure 1:

1. An inner loop that evolves a specially prepared quantum state towards a solution using *p* steps based on variational parameters *β* and *γ*. The value "*p*", which is an integer ≥ 1, can be used to trade off time (and total circuit depth) for solution accuracy. In this document, this is referred to as the "p-number".
2. An outer loop that seeks to optimise the quantum state to satisfy a high fraction of constraints by evaluation of the quantum expectation value for the defined problem. This is achieved by varying parameters *β* and *γ* using a hybrid classical/quantum approach known as the "variational quantum Eigen solver".

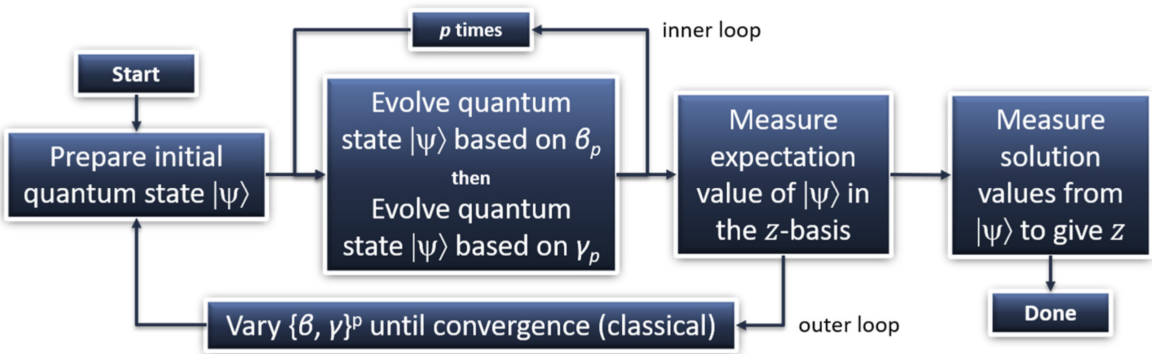

Figure 1 - Representation of the Quantum Approximate Optimisation Algorithm

At algorithm completion, a measurement is made to sample from the generated solution space.

Challenges

In terms of identifying feasible problems for mapping onto the QAOA, challenges reside in the conceptual mapping of the source problem onto the constraint satisfaction framework.

In terms of taking problems of interest onto the QAOA, challenges reside in minimising the number of constraint terms, and minimising the growth in constraint terms with problem size.

In terms of implementation, parameter setting strategies are known to create challenges as the algorithm's p-number increases, due to the curse of dimensionality of *2p* parameters.

References

1. Berg J, Hyttinen A, Jarvisalo M, "Applications of MaxSAT in Data Analysis" (2015), Pragmatics of SAT workshop 2015
2. Farhi E, Harrow A, "Quantum Supremacy through the Quantum Approximate Optimization Algorithm" (2016), https://arxiv.org/abs/1602.07674
3. Farhi E, Goldstone J, Gutmann S, "A Quantum Approximate Optimization Algorithm" (2014), https://arxiv.org/abs/1411.4028
4. Hormozi L, Brown E W, Carleo G, Troyer M, "Nonstoquastic Hamiltonians and Quantum Annealing of an Ising Spin Glass", https://arxiv.org/abs/1609.06558
5. Karp R M, "Reducibility Among Combinatorial Problems" (1972), in R. E. Miller and J. W. Thatcher (editors), Complexity of Computer Computations, New York, Plenum. pp. 85–103
6. Wang Z, Hadfield S, Jiang Z, Rieffel E G, "The Quantum Approximization Algorithm for MaxCut: A Fermionic View", https://arxiv.org/abs/1706.02998