## Hackathon 2023

### Vendors

AWS (OQC): Machine Learning, Simulation

AWS (Rigetti): Machine Learning, Optimisation

AWS (QuEra): Optimisation

D-Wave: Optimisation

ORCA Computing: Optimisation, Machine Learning

Emulation: D-Wave: Optimisation

Emulation: AWS: Machine Learning, Optimisation, Simulation

Quantinuum: Simulation

IBMQ: Simulation, Optimisation, Machine Learning

Emulation IBMQ: Simulation, Optimisation, Machine Learning

Classiq (IonQ): Simulation

AWS(IonQ): Optimisation

Emulation: Classiq (IonQ): Simulation

### Project 1:

Practical equivariant embeddings for DNA sequencing over the current NISQ devices

* Machine learning
* Zaiku Group Ltd.
* Healthcare
* AWS (OQC)
* AWS (Rigetti)
* Emulation: AWS

The team identified the reverse-complement symmetry of DNA sequences and defined an angle-based embedding with two features per qubit, in which the symmetry was represented on the Hilbert space via conjugation by a tensor product of Pauli-X gates. They created a synthetic dataset of eight sequences in length and defined via their representation an equivariant circuit to detect the string ‘AT’ within a sequence - an equivariant version of a small, typical variational ansatz. They compared this model’s performance to an analogous non-equivariant model on a simulated device. Time did not permit training on a real device but the angle-based solution would have been suitable for NISQ devices.

Outcome

The equivariant model appeared to reach a lower cost than the non-equivariant counterpart given the same amount of training, which was a nice conclusion.

Time limitations meant that the team could not practise a training/validation/test data split for benchmarking. Further, the hybrid nature of training loops meant that performing the task on a real device would have taken too long. These loose ends would be a natural starting point for further investigation, along with applications of the method to real-world datasets. It would also be interesting to compare the two quantum models to a classical approach to the same task.

References

J. J. Meyer, M. Mularski, E. Gil-Fuster, A. A. Mele, F. Arzani, A. Wilms, and J. Eisert, “Exploiting symmetry in variational quantum machine learning,” arXiv:2205.06217, 2022.

N. Innan and M. Al-Zafar Khan, “Classical-to-Quantum Sequence Encoding in Genomics,” arXiv:2304.10786, 2023.

### Project 2

The travelling salesman problem – route optimisation in healthcare

* Optimisation
* NHS
* Healthcare
* AWS (QuEra)
* D-Wave
* Emulation: AWS

Routing and scheduling optimisation is a constant exercise when seeking cost reduction and improvement of service times in the healthcare operational environment. In addition to the transportation of patients via ambulance, NHS services also require the transportation of time-sensitive material to multiple locations. The computation of optimal routes is of value and a common classical problem. An algorithm in computer science is said to be “efficient” if it executes in polynomial time or less. Owing to the nature of routing and scheduling problems as NP-complete, the application of quantum computing for problem solving in a healthcare setting is an exciting proposition. Routing and scheduling optimisation problems are commonplace, and innovative ways to tackle these problems may offer advantages not yet realised via classical methods.

Solution

To make the problem tractable, the team used a hybrid method that combines classical compute for clustering (k-means-constrained) with quantum annealing. The clustering technique focuses on finding non-overlapping clusters, constraining the search space, and making the solution viable for existing hardware and at scale.

The team used the D-Wave solver to implement the quantum annealing. Their objective function is defined in

Simulated annealing in a classical setting is prone to being trapped in local minima. Finding the global minimum using classical compute is computationally intensive and takes time. Quantum annealing takes advantage of low-energy states in quantum physics to find the low-energy states of a problem, and the solutions provided by D-wave are designed for these types of problems. Once the team encoded their problem onto an initial Hamiltonian, mapping eigenstates to energies, they sought an optimal solution.

Outcome

Several clustering methods were attempted, and k-means-constrained proved most useful. Hospitals were clustered before using D-Wave to provide solutions for the optimal route based on the shortest path between clusters. The team had a limited number of qubits to use but were still capable of finding solutions for up to five health centres.

References

S., Jain, 2021. Solving the traveling salesman problem on the d-wave quantum computer. Frontiers in Physics, p.646.

Quantum chemistry; Electronic structure problem

Dynamics can be simulated using Hamiltonian simulation (typically Trotter methods) [18] Clinton, L., Bausch, J., and Cubitt, T. “Hamiltonian simulation algorithms for near-term quantum hardware.” Nat. Commun. 12 (2021), 4989. arXiv:2003.06886.

and have been demonstrated for an 8 × 1 lattice on 16 superconducting qubits [46] Arute, F., Arya, K., Babbush, R., et al. “Observation of separated dynamics of charge and spin in the Fermi–Hubbard model.” arXiv:2010.07965 (2020).

### Project 3

Explore the use of quantum computing for unit commitment to balance the electricity grid

* Optimisation
* National Grid ESO
* Energy
* ORCA Computing
* D-Wave
* Emulation: D-Wave

The Unit Commitment (UC) problem is a fundamental problem in the electric power industry. The objective of UC is to determine an optimal schedule for each generating unit to meet the demand for power with the minimum cost.

Given its mixed-integer programming (MIP) nature, UC solution speeds could potentially be greatly enhanced with quantum computing.

Solution

We took a simplified version of UC to the hackathon that made it a combinatorial problem in which all the optimisation variables were binary, representing whether a generator was online or offline. Modelled constraints were positive margins and minimum up/down times for generators.

Objective function – minimise costs to operate the system:



Subject to:



We worked with ORCA Computing and D-Wave to solve this problem using quantum hardware and simulators.

Outcome

The team explored the possibility of formulating UC as a quadratic unconstrained binary optimisation (QUBO). Additionally, we used D-Wave’s hybrid solver to implement the constrained problem. This way, the team succeeded in solving a toy case of four generators and four time periods.

We found that:

* Formulating a QUBO problem for a simplified version of UC wasn’t straightforward and the limited time available made it more challenging. We can only speculate how much harder it would be for the full Security-Constrained UC problem
* Solving constrained optimisation problems using quantum platforms allows specifying ‘a-priory’ Langrage multipliers for the constraints. This means that it’s difficult to assess the optimality of a solution and the general applicability of an implementation without a reference solution
* While quantum computing shows great potential to solve complex optimisation problems in other sectors, there’s an outstanding gap before it can be directly applicable to power system operation. We’d be keen to continue exploring how to bridge this gap.

### Project 4

Quantum computing for nuclear fusion engineering and design with the finite element method

* Simulation
* UKAEA
* Energy
* AWS (OQC)
* Quantinuum
* Emulation: AWS

The design of nuclear fusion reactors requires accurate simulations of complex phenomena to optimise design parameters and reduce the need for costly physical prototypes. Unfortunately, the computational cost of some of these complex simulations limits their applicability and benefits. The improved computational performance promised by quantum computers will be of great benefit for shortening engineering design cycles and improving the accuracy of designs.

Solution

Finite Element Analysis (FEA) is a numerical simulation technique used for fusion reactor design that solves the complex dynamics of the fusion domain using a linearised set of equations. This is equivalent to solving the equation Ax=b, where the inverse of the matrix A is used to determine the vector x given a set of initialised values of b. The Harrow-Hassidim-Loyd (HHL) algorithm was deployed to solve the equation Ax=b, as an early demonstrator of a quantum solver for linear equations. The algorithm requires a large circuit depth, due to the initialisation operations required to store the value of the vector b. Gate operations were then conducted to apply the inverse matrix A on the qubit value of vector b to solve for vector x. To solve for a single qubit circuit line, seven qubits were used as control and scratch qubits. This was implemented on the eight-qubit Lucy quantum processor provided by Oxford Quantum Circuits.

Outcome

The experiment demonstrates that it is possible to solve linear matrix operations. However, these were based on classical binary input values, and representing large vectors with a large matrix space at the binary level will require large amounts of qubits and quantum information to represent the binary data and respective quantum transformation. Therefore, it is proposed to improve the encoding of classical high-level data to quantum qubit data to improve the computational expense and complex quantum algorithms with large depths of circuits and possible larger entanglement requirements. Also, to implement one logical qubit requires approximately seven physical qubits, which will significantly increase the number of physical qubits required if error corrects are to be considered.

FEA is a widely used numerical simulation technique, and therefore this algorithm has applicability in all complex engineering sectors such as aerospace, automotive, construction and so forth. In future exploration it would be interesting to merge quantum numerical simulation techniques in neural network loss functions with optimisations to improve the training for Physics Informed Neural Networks (PINNS).

### Project 5

1-D Monte Carlo particle transport

* Simulation
* Jacobs
* Energy
* Quantinuum
* IBMQ
* Emulation: IBMQ

Mathematical modelling of the transport of subatomic particles in matter, such as neutrons, plays an important role in the operation of radioactive facilities. The Monte Carlo (MC) approach to simulating radiation transport provides accurate results but is computationally expensive. Quantum computing may have a role in speeding up MC simulations.

Solution

The MC method consists of sampling from specified statistical distributions to determine the collision type, direction and distance travelled by the particle. Quantum sampling may be more advantageous to classical sampling due to accuracy and speed improvements. An alternative method involved encoding the probability distribution of particle penetration as a biased quantum coin, with the probability of penetration estimated via a circuit for quantum amplitude estimation (QAE). The H1 Emulator provided by Quantinuum was used to implement these circuits, with sampling of uniform distribution achieved via circuits with a depth of one and four gates. Sampling of exponential distribution was done via a state preparation box, leading to a deeper circuit. The circuit for QAE consists of more quantum gates, corresponding to greater depth if non-optimised, and is ideal for fault-tolerant systems, while quantum sampling may be better suited for NISQ devices.

Outcome

The solutions obtained via quantum sampling showed a large variation from the analytical solutions. Error correction techniques were not implemented, which could account for the difference in expected and obtained results. The use case was also a simple example with only two energy levels and limited neutron interactions (capture, scatter, leak). Future extensions could consider higher dimensions, more energy levels, as well as fission interactions. The alternative method using QAE was not fully explored, and could be better investigated in the future.

### Project 6

Near-term quantum linear solver algorithms

* Simulation
* Rolls Royce
* Aerospace
* Classiq (IonQ)
* IBMQ
* Emulation: Classiq (IonQ)

Rolls Royce designs and manufactures power systems for applications such as aviation and marine propulsion. For the simulation and modelling of modern power systems, engineers utilise Computational Fluid Dynamics and Finite Element Method algorithms. These reduce to the computationally intensive task of matrix inversion for solving systems of differential equations, which can bottleneck even supercomputing throughputs. Quantum computing’s dense matrix algebra indicates a potential advance for this use case.

Solution

Examples of the problem present as small instances of linear systems of equations, which may be formulated as Ax=b, presented with a matrix of equations A and a vector of constants b to solve for the vector x. Examples from sizes of 2x2 to 16x16 matrices were solved for this use case. Two different algorithms were studied to approach this, a Coherent Variational Quantum Linear Solver (CVQLS) that could be easily implemented on NISQ hardware and the Quantum Singular Value Transformation (QSVT), which is more intended for fault-tolerant systems. These were predominantly developed in Classiq to optimally compile a circuit with minimal execution time, with some bits being done in Pennylane and exported via QASM file to Classiq.



Outcome

The team performed runs of the CVQLS on the 16-qubit IBM Guadalupe device using Classiq and IBM Quantum, and on a 20-algorithmic qubit IonQ Aria device as well as the IonQ Harmony device using Classiq. Limited by the circuit size, the team was able to reach results for linear systems of up to 4x4 using the hardware devices, while simulation reached up to 16x16 matrix sizes. Besides normally not being a NISQ algorithm, the QSVT encountered issues in its hardware implementation as Pennylane’s newly introduced QSVT module did not compile to QASM code, and this will be a necessary step towards implementing the QSVT in hardware. Further implementations of quantum linear solvers are expected to be developed by Rolls Royce and partners for the future development of hydrogen-based systems.

References

<https://pennylane.ai/qml/demos/tutorial_coherent_vqls>

https://pennylane.ai/qml/demos/tutorial\_apply\_qsvt/

### Project 7

Skills constrained capacitated vehicle routing problem with time window (SC-CVRP-TW)

* Optimisation
* Unisys
* Logistics
* IBMQ
* D-Wave
* Emulation: IBMQ

A specific version of the Vehicle Routing Problem (VRP) is the Vehicle Routing Problem with Time Window and Skill Set constraints (VRP-TWSS). This involves calculating optimal routes for a fleet of vehicles where each vehicle has a particular set of skills, and each destination has a required skill and time window.

Providing a successful resolution to the optimisation of vehicle routing problems will:

Current classical solutions to these problems do not scale to the level faced by many of our customers, even with High Performance Computing (HPC).

Solution

No quantum circuit was developed as the solution was developed on D-Wave’s quantum annealer. Instead, the mathematical model was implemented directly using the Constrained Quadratic Model (CQM) in D-Wave’s Ocean Software Development Kit (SDK).

The solution was based on the mathematical model defined in L. Han’s paper.[1] In the time available, no consistent solution was returned by D-Wave. Different solutions were considered optimal by the quantum annealer on different runs of the code. This may very well result from the high density of constraints, a problem Unisys have encountered before and outlined in our whitepaper.[2]

Outcome

During the hackathon, the team very quickly (within a few hours) encountered hardware limitations using the IBMQ platform. The team’s use case was solving a small vehicle routing problem (32 nodes and 5 vehicles, or n32-k5) with time-window and skill-set constraints. Even without the time-window and skill constraints embedded in the model, the IBMQ platform did not yield results after running the model, even with a smaller model consisting of 8 nodes and 3 vehicles (n8-k3). By comparison, running the same model on the High-performance parallel linear optimisation Software (HIGHS) classical solver installed on an Orange Pi 5 returned results within minutes. Using a quantum annealer from D-Wave, results were also obtained within minutes.

References

[1] L. Han, “Metaheuristic Algorithms For The Vehicle Routing Problem With Time Window And Skill Set Constraints”, Dec 2016. Available: Han-Lu-MSc-IENG-Dec-2016.pdf (dal.ca)

[2] S. Sinno, T. Gross, A. Mott, A. Sahoo, D. Honnalli, S. Thruvavakkath, B. Bhalgamiya et al., “Performance of Commercial Quantum Annealing Solvers for the Capacitated Vehicle Routing Problem,”, Sep 2023. Available: 2309.05564.pdf (arxiv.org)

### Project 8

Price prediction over different time horizons

* Machine learning
* Nomura International Plc.
* Finance
* IBMQ
* ORCA Computing
* Emulation: IBMQ

The team was provided with a problem to predict the time series of financial data. The data are well known to be noisy with very low signal-to-noise ratio, and also have complex memory structure. The aim was to create a supervised machine-learning model that makes use of quantum resources.

Solution

The main approach to the problem was to utilise a quantum-enhanced Long Short-Term Memory (LSTM) model. The LSTM model is a neural network-based model that separately models the long and short memory, along with the process of forgetting the past. The process contains several parallel neural networks that are trained as part of the calibration. We replaced the neural networks with quantum equivalents and trained the model with IBMQ resources. The model showed promising results within the reach of currently available hardware (16-qubit processor), while we utilised a simulator for most of the analysis.

Outcome

The result was very encouraging and showed potential for how existing quantum hardware can be used to enhance existing algorithms or as a hybrid combination of classical and quantum parts. The project opened more questions to pursue in the future, such as the optimal combination, and which one dominates any existing setup, but it suggests that some form of enhancement is within reach in the short term.

### Project 9

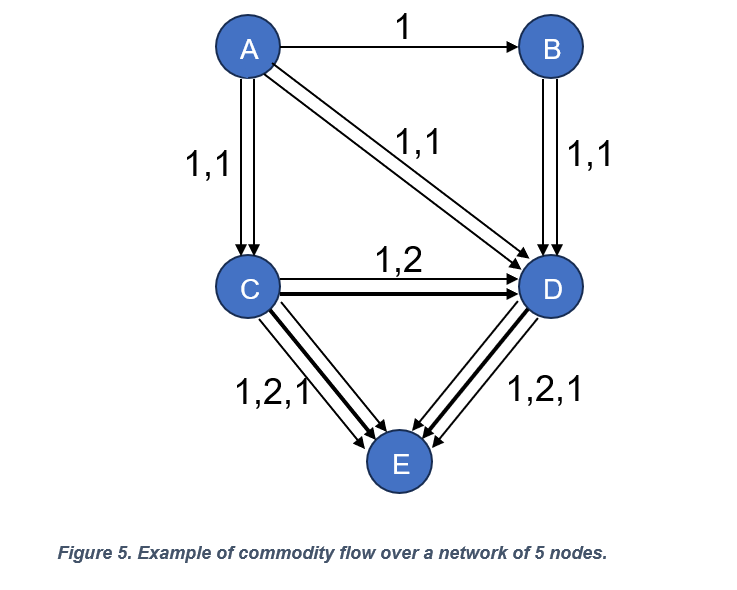
Generating efficient and resilient routes for unicast and multicast traffic

* Optimisation
* BT
* Telecommunications
* D-Wave
* AWS (Rigetti)
* Emulation: D-Wave

Allocating routes for transmitting data across telecommunication networks can become computationally hard under certain conditions, such as when the network is very congested, with different demands when the data traffic allocation on each channel is broken into a small number of options (integer flow). This can occur, for example, when assigning optical channels at layer 1, and where it is required to provide resilience by planning two independent routes for each demand that is completely independent of the first route. Further complications can include requirements for multicast (one source to many sinks) traffic and constraints such as the maximum latency of a route.

Solution

The team identified that data traffic flow is a variation of the standard commodity flow satisfaction problem. The team used a Quadratic Unconstrained Binary Optimisation (QUBO) representation of the routing and demand satisfaction model, and this was run on a D-Wave quantum annealing system with 5000 qubits. A number of approaches were investigated to represent this problem using the QUBO form. For single-commodity flow problems (both unicast routes and multicast trees), the team settled on a flow continuity model that used Kirchoff’s current laws at each node. Qubits represented the links of the network. The capacity of each link could be constrained using the choice of encoding the amount of flow in the link by using a binary expansion of the total capacity of each link (such that no configuration of weights could exceed the maximum capacity). This was efficient, creating a number of weighted qubits of order log C for each edge. The encoding of single-commodity flow required approximately E log2 C qubits (where E is the number of links in the original network, and C is the average capacity). However, for highly connected networks, additional qubits would be needed to overcome native Quantum Processing Unit (QPU) connectivity constraints. For the more realistic multicommodity flow problem (planning routes for different types of data), it was necessary to create multiple copies of the original qubit representation of the different commodity flows on the network coupled to represent the shared capacity constraints, with slack variables because the capacity represented an upper limit.



Outcome

The team showed that it was possible to encode a large number of different variations of the efficient routing problem. Smaller problems up to network sizes of 10 nodes were solved using the D-Wave native QPU (5000 qubits). Larger problems were run on the D-Wave Leap hybrid solver. Repeated anneals usually returned some correct solutions (with the lowest energy in the sample set), but the QUBO formulation had some technical issues that could affect the quality of the solution. Changing the weighting between the linearly independent terms (representing different constraints) often affected the probability of returning a valid solution in any given run. This was true both for the D-Wave native QPU and the hybrid solver.

The fact that very complex routing problems could be represented quite efficiently in numbers of qubits in a QUBO was very promising. However, the fact that the ground state optimum (and correct) solutions were not found a high percentage of the time was a limitation. This was not unexpected given that the annealer has some noise, and the encoding of the problem had not given any consideration to error mitigation, plus we expect larger problems to have smaller energy gaps between the ground state solution, and excited states. Whether an encoding on the current D-Wave annealer could mitigate errors (for example, by coupling qubits in redundant groups to represent single logical QUBO variables) and whether this would be efficient would be interesting to explore. The effect of varying the weightings of the different linearly independent constraint terms on the quality of the solution set was also something that needs to be explored and understood further. Ground state solutions were found, however, which are promising, especially in light of the potential to improve the annealing performance with emerging developments in quantum annealer hardware and methods. Continued increase in quantum annealer QPU size will also be helpful.

This use case was further developed by the hackathon team, leading to a paper entitled ‘Optical Routing with Binary Optimisation and Quantum Annealing’ where the details of this problem are explored in more depth and future work building from that conducted at the hackathon is reported.

References

E. Davies et al., “Optical Routing with Binary Optimisation and Quantum Annealing,” arXiv, 2402.07600v1, 2024

### Project 10

Vaccination centre location

* Optimisation
* Applied
* Quantum Computing
* Healthcare
* AWS (IonQ)
* D-Wave
* Emulation: AWS

Optimising the location of vaccination centres during a pandemic is an important means of reducing the pandemic’s impact. Optimal locations will help to maximise vaccination rates in the population, thereby reducing the chances of the vaccinated individuals catching the disease and also helping reduce the rate of spread, benefiting the whole population. The real-world problem objective is to maximise vaccination take-up by considering travel times (to be minimised), clinic capacity, population size and demographics, and the proportion of the population already vaccinated. Here, we consider the simplified problem of minimising population to vaccine centre travel times. At large scale, such problems cannot be optimally solved using classical means.

Solution

The problem can be considered as a weighted-set cover problem and formulated as a QUBO, which can be translated into Ising form for running on a quantum computer.[1] The objective function, which is to be minimised, encodes the total distance from the population centres, N, (nodes) to the vaccination centre(s), V, (also nodes) and includes certain problem constraints encoded using the Lagrange multiplier method, such as the requirement that each population centre is only served by one vaccination centre.

Two NISQ-era hybrid algorithms were used: QAOA (Quantum Approximate Optimisation Algorithm) and VQE (Variational Quantum Eigensolver), each of which seeks to optimise a parameterised circuit.[2, 3] An example circuit to run the VQE parameterised algorithm is shown above. Two simulators were used: Amazon Braket SDK for VQE and IBM Qiskit for QAOA.



Outcome

Two small-scale instances of the problem were considered: (A) N=4, V=2 and (B) N=3, V=2, with instance (A) run using QAOA and instance (B) using VQE. Each was run using up to 20 random initial choices for the circuit parameters and was successful in very significantly improving the probability of finding the optimal configuration (the ground state) by comparison with the initial problem starting state – for example, the VQE approach was able to achieve an 89% probability of finding the ground state with the best set of initial parameters.

An important unresolved question is how the approach scales with problem size and in particular, if the time to solution on a quantum processor for a real-world problem instance can be achieved (i.e. sufficiently short), which would make this approach practically valuable.

If improved versions of this method and associated hardware developments can be achieved, this approach could be applied to a wide variety of valuable applications in healthcare and other sectors.

References

A. Lucas, “Ising formulations of many NP problems,” arXiv, 1302.5843v3, 2013.

E. Farhi, J. Goldstone and S. Gutmann “A Quantum Approximate Optimization Algorithm,” arXiv, 1411.4028v1, 2014.

A. Peruzzo et al., “A variational eigenvalue solver on a photonic quantum processor,” Nature Communications, vol. 5, no. 4213, 2014.

Variational Quantum Algorithms (Delzel Page 258)

Factoring: Variational methods for factoring have been proposed which exploit a mapping between the factoring problem and that of finding the ground state of an Ising Hamiltonian [44] Anschuetz, E., Olson, J., Aspuru-Guzik, A., and Cao, Y. “Variational quantum factoring.” In: Quantum Technology and Optimization Problems (2019), 74–85. arXiv:1808.08927 . The authors use the QAOA ansatz and heuristically find that p = O(n) rounds of the ansatz can lead to a good solution overlap for small system sizes.

Combinatorial optimization: In the Quantum Approximate Optimization Algorithm (QAOA), combinatorial problems on bitstrings can be encoded in the Pauli-Z basis with Hamiltonian HP [30].Farhi, E., Goldstone, J., and Gutmann, S. “A Quantum Approximate Optimization Algorithm.” arXiv:1411.4028 (2014). By finding the state that minimizes ⟨ϕ(θ)|HP |ϕ(θ)⟩, where |ϕ(θ)⟩ = U(θ)|0⟩, the optimal bit-string can be extracted by sampling the optimized state in the computational basis. A widely studied ansatz for this problem is the Quantum Alternating Operator Ansatz (which bears the same acronym as the algorithm), inspired by Trotterized adiabatic evolution [31]. The ansatz takes the form U(γ, β) = Qp l=1 e−iβlHMe−iγlHP where HM is a specific “mixing” Hamiltonian. This ansatz is known to be computationally universal (when p → ∞) for certain classes of Hamiltonians [32, 33]. Moreover, under reasonable complexity-theoretic assumptions, it is known that sampling from the output of the QAOA at p = 1 is classically hard [34]. On the other hand, there is evidence that shallow (small p) QAOA does not perform well [35, 36, 37, 38], leading to intuition that p may need to grow with problem size to produce better approximate solutions than what can be easily found classically. Alternatively, there is some evidence that an exponential number of samples from shallow QAOA circuits may yield polynomial speedups over classical methods for finding exactly optimal solutions [39, 40], see the page on beyond-quadratic speedups for combinatorial optimization.