

# Decoded Quantum Interferometry: Interference-Driven Quantum Speedup

*Optimization as Decoding for Fourier-Sparse Objectives*



**PHYS360 — Paper Draft**

Instructor: Dr. Chandra Turpen

Student: Jaehyun Kwak  
UID: 122274570 · Email: [apjh0703@terpmail.umd.edu](mailto:apjh0703@terpmail.umd.edu)

Affiliation: University of Maryland, College Park (Exchange from Korea University)

Word Count (text-only, excluding references/figures): \_\_\_\_\_  
Draft/Final: **Draft**

# Table of Contents

- I. Introduction
- II. Background
  - A. Quantum Computing Basics
    - 1. Superposition and Interference
    - 2. Interference for Global Structure Extraction
  - B. Prior Work
    - 1. Hamiltonian-Based Approaches
    - 2. Limitations of Hamiltonian-based Approaches
- III. Decoded Quantum Interferometry
  - A. Reformulating Optimization as Decoding
  - B. Fourier Sparsity and Structural Promise
  - C. DQI Analysis
- IV. Significance and Potential Impact
  - A. Broader Class of Applicable Problems
  - B. Potential Practical Relevance
- V. Limitations
  - A. Hardware Constraints
  - B. Possibility of Classical Catch-Up
- VI. Conclusion
- References

# I . Introduction

This review paper introduces a recent proposal called **Decoded Quantum Interferometry (DQI)** as an alternative approach to discrete optimization that combines short quantum circuits with classical decoders. Many quantum optimization algorithms encode the objective as an energy landscape and try to prepare low-energy states. DQI instead uses interference to highlight useful global structure in the objective and then decodes that structure with classical post-processing.

Optimization appears across science and engineering, and as problem size grows the search space typically expands combinatorially so that exhaustive search quickly becomes impossible. Classical methods such as convex relaxation and heuristics cope with this growth to some extent, but their performance often degrades as dimensionality, noise, or combinatorial interactions increase.

Quantum algorithms offer a different resource. By using superposition and interference, they can process many candidate solutions in parallel. Shor's factoring algorithm is a famous example. It uses the quantum Fourier transform (QFT) to reveal a hidden period and achieves an exponential speedup over the best-known classical methods [1]. This idea of using interference and Fourier analysis to expose structure also motivates DQI.

Most previous quantum methods are Hamiltonian-based [2]. They encode the cost function as a Hamiltonian and reach for low-energy states via quantum annealing or the Quantum Approximate Optimization Algorithm (QAOA). On near-term devices these methods require relatively deep circuits, are sensitive to calibration noise, and can suffer from barren plateaus, where gradients become exponentially small and parameter optimization stalls, during parameter training. DQI offers a different perspective. It reformulates optimization problems as decoding tasks for Fourier-sparse objectives. The quantum circuit imprints patterns of the objective into measurement statistics through interference, and a classical decoder then recovers informative features such as promising assignments.

This paper aims to give an accessible introduction to DQI. I explain how it converts optimization into decoding and summarize two central case studies from the original work: an algebraic optimization-by-interference problem with a provable quantum speedup and a sparse max-XORSAT instance where DQI competes with classical heuristics.

## II . Background

Before analyzing DQI, it is helpful to review how quantum algorithms use superposition and interference, and how previous approaches have treated optimization problems. This section recalls the basic quantum ingredients that DQI relies on and summarizes Hamiltonian-based methods that form the main comparison point.

### A. Quantum Computing Basics

#### 1. Superposition and Interference

A single qubit can be in a superposition of basis states,

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle, |\alpha|^2 + |\beta|^2 = 1.$$

Here  $\psi$  is the quantum state.  $\alpha$  and  $\beta$  are complex amplitudes, and  $|\alpha|^2$  and  $|\beta|^2$  give the probabilities of measuring 0 or 1 in the computational basis. For  $n$  qubits, the joint state is a superposition of  $2^n$  basis states.

Phases of the amplitudes matter as much as magnitudes. A global phase on all amplitudes has no physical effect, but relative phase between components can change later interference. Branches whose phases line up add constructively, while branches with opposite phase cancel. Many quantum algorithms are therefore designed to encode information about the problem into relative phases and then convert them into measurable probability differences.

#### 2. Interference for Global Structure Extraction

Interference is especially useful for extracting global information about a function or search space. A common template is: prepare a large superposition over candidate solutions, mark components by phases that encode which inputs are “good”, and apply an interference transform so that useful patterns appear in the measurement distribution.

Grover’s search algorithm is a standard example [3]. It prepares a uniform superposition over  $N$  basis states, uses an oracle – a black-box that can recognize the marked item - to flip its phase, and applies a fixed “diffusion” step that recombines all paths so that the marked state interferes constructively while the others interfere destructively. After  $O(\sqrt{N})$  iterations, the marked item has a much higher probability of being observed than under a random guess. DQI follows the same high-level pattern, but instead of one marked state it engineers phase markings so that families of assignments that satisfy many constraints interfere constructively, while assignments that violate many constraints interfere

destructively. The resulting interference pattern encodes global information about the objective that a classical decoder can then interpret.

## B. Prior Work

### 1. Hamiltonian-Based Approaches

Before DQI, most quantum approaches to combinatorial optimization were Hamiltonian-based. The idea is to encode the cost function as an energy landscape and use quantum dynamics to reach low-energy states that correspond to good solutions. One defines a problem-dependent cost Hamiltonian  $H_c$  whose ground state encodes an optimal assignment and a simpler “mixer” Hamiltonian  $H_M$  that spreads amplitude over the search space.

Adiabatic and quantum annealing methods work in continuous time. The system is initialized in the ground state of  $H_M$  and then evolved under an interpolating Hamiltonian

$$H(s) = (1 - s)H_m + sH_c, s \in [0, 1],$$

for a total runtime  $T$ . Here  $s$  is the dimensionless interpolation parameter that labels the schedule from the initial Hamiltonian  $H(0) = H_M$  to  $H(1) = H_c$ , and  $T$  is the total evolution time. As  $s$  changes, the smallest energy difference between the ground state and the first excited state of  $H(s)$  is the spectral gap. If  $H(s)$  is changed slowly enough compared to this gap, the adiabatic theorem says that the system stays near the ground state of  $H_c$ ; in that case, the final state approximates an optimal solution [4]. Quantum annealers such as those built by D-Wave implement a discretized, hardware-specific version of this idea for cost functions that can be written as an Ising or QUBO model, where the energy is a weighted sum of linear and pairwise terms in binary variables.

The Quantum Approximate Optimization Algorithm (QAOA) discretizes a related idea into a shallow, parametrized circuit. For a given depth  $p$ , QAOA alternates between cost and mixer unitaries,

$$U(\gamma, \beta) = \prod_{j=1}^p e^{-i\beta_j H_M} e^{-i\gamma_j H_c},$$

acting on an initial product state. The angles  $\gamma$  and  $\beta$  are lists of real parameters and each pair  $(\gamma_j, \beta_j)$  controls one layer. The parameters are optimized so that the expected cost  $\langle H_c \rangle$  is as low as possible. The cost unitary imprints phase shifts that depend on the objective value, the mixer unitary spreads amplitude to explore nearby configurations, and a classical optimizer adjusts the angles so that constructive interference builds up on low-cost bit strings.

## 2. Limitations of Hamiltonian-Based Approaches

Hamiltonian-based algorithms face several obstacles, especially on current hardware. To improve solution quality, adiabatic and annealing protocols usually need longer schedules, and QAOA typically requires higher depth, which exposes the computation to more gate noise and calibration errors. Small spectral gaps can force very slow evolution for adiabatic algorithms, and in hard optimization instances these gaps can shrink exponentially, making the required runtime enormous. QAOA also requires many circuit repetitions to estimate costs and a classical search over a high-dimensional parameter landscape that can exhibit barren plateaus where gradients vanish and optimization stalls.

Moreover, optimal parameters can be instance-dependent and sensitive to hardware drift, so parameters trained in one setting may not transfer reliably to another. Overall, Hamiltonian-based performance is highly instance-dependent and often lacks strong analytic guarantees. These methods largely exploit local moves in an energy landscape which may miss global algebraic or Fourier structure present in many combinatorial objectives. This motivates exploring a different paradigm in which quantum circuit is used to highlight and decode such structure. DQI, introduced by Jordan et al. [2], embodies this alternative approach and sets the stage for the next section.

## III. Decoded Quantum Interferometry

Decoded quantum interferometry (DQI) is an alternative approach to optimization that does not work with an explicit energy landscape. Instead, it operates on the Fourier representation of the objective function and treats optimization as a decoding task [2]. A short quantum circuit creates an interference pattern whose measurement statistics carry information about how well each assignment satisfies a family of constraints, and a classical decoder interprets this pattern. The quantum device supplies structured samples, and the heavy computation is offloaded to classical decoding algorithms.

### A. Fourier Sparsity and Structural Promise

DQI is designed for Boolean objective functions whose Fourier spectra are dominated by a small number of low-degree terms. Consider a function

$$f: \{-1, +1\}^n \rightarrow \mathbb{R},$$

where  $n$  is the number of variables and each input is a vector  $x = (x_1, \dots, x_n)$ , with  $x_i \in \{-1, +1\}$ . Using this  $\{-1, +1\}$  encoding, we can expand  $f$  in terms of parity features:

$$f(x) = \sum_{S \subseteq [n]} \hat{f}(S) \chi_S(x).$$

Here  $[n] = 1, 2, \dots, n$  is the index set,  $S$  is any subset of  $[n]$ ,  $\hat{f}(S)$  is the Fourier coefficient on  $S$ , and  $\chi_S(x)$  is the parity over indices in  $S$  defined by

$$\chi_S(x) = \prod_{i \in S} x_i.$$

The size  $|S|$  is the degree of the term, that is, the number of variables that interact in that component.

Many combinatorial objectives have low-degree, sparse Fourier spectra: only a few coefficients  $\hat{f}(S)$  are large, and most of the weight lies on subsets  $S$  of small size. The Max-Cut problem provides a concrete example. Let  $G = (V, E)$  be an unweighted graph with vertex set  $V$  and edge set  $E$ . We assign a label  $x_i \in -1, +1$  to each vertex  $i$ , indicating which side of the cut it lies on. An edge  $(i, j)$  is cut when  $x_i \neq x_j$ , or equivalently when  $x_i x_j = -1$ . A standard form of the Max-Cut objective is

$$\text{cut}(x) = \frac{1}{2} \sum_{(i,j) \in E} (1 - x_i x_j).$$

Here  $|E|/2$  is the expected cut size for a random assignment. The remaining terms are pairwise parities  $x_i x_j$  with coefficient  $-1/2$  on each edge  $(i, j)$ . There are no single-variable terms and no higher-order terms, so the Fourier spectrum is supported only on degree-0 and degree-2 terms.

DQI gains the most in regimes like this, where a small set of low-degree parity features already captures most of the structure. When the spectrum is dense and dominated by high-degree terms, no short list of parity patterns captures most of the structure, and the advantage of DQI becomes less clear.

## B. Reformulating Optimization as Decoding

DQI replaces the Hamiltonian-based slogan, “minimize an energy landscape” with “decode hidden structure from interference”. Instead of directly searching for an  $x$  that maximizes  $f(x)$ , it asks which parity structures distinguish high-scoring assignments from typical ones and designs a quantum experiment whose measurement statistics reveal those structures.

The implementation in Jordan et al. focuses on constraint families such as max-LINSAT and sparse max-XORSAT, where the objective counts how many linear constraints modulo 2 are satisfied [2]. These constraints can be encoded in a binary matrix  $B$  and a right-hand

side vector  $v$ . If  $B$  has  $m$  rows and  $n$  columns, and  $x \subseteq \{0,1\}^n$  is a candidate assignment in the  $\{0, 1\}$  encoding, then the constraints can be written compactly as

$$Bx = v \pmod{2}.$$

This is the same type of linear system that appears in error-correcting codes, where  $B$  acts as a parity-check matrix and recovering information about  $x$  from values derived from  $Bx$  is a decoding problem.

Operationally, the circuit proceeds in three conceptual stages. First, the  $x$ -register is initialized in the uniform superposition over all  $n$ -bit assignments,

$$\frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle,$$

so that every candidate assignment appears with equal amplitude. In parallel, a  $y$ -register is prepared in a superposition over low Hamming-weight bit strings, meaning bit strings with only a few ones, and each  $y$  encodes a small subset of constraints. Restricting the Hamming weight of  $y$  to be at most  $l$  imposes a degree budget  $l$  on how many constraints are combined at once.

Second, the circuit uses  $B$  and  $y$  to compute a syndrome register  $s$  that summarizes which constraints are touched and applies phase factors that depend on how many of those constraints are satisfied by  $x$ . A final layer of Hadamards on the  $x$ -register converts these relative phases into amplitude differences, producing an interference pattern in which assignments with larger objective value tend to have larger amplitudes.

Third, the syndrome  $s$  and the known matrix  $B$  are fed into a coherent version of a classical decoder. Implemented reversibly, this decoding procedure “uncomputes” auxiliary registers and leaves the  $x$ -register in a state whose amplitude on  $|x\rangle$  depends on the scalar value  $f(x)$  through a low-degree polynomial  $P(f(x))$ :

$$|\Psi_{\text{out}}\rangle \propto \sum_x P(f(x)) |x\rangle.$$

Here  $P$  is a polynomial of degree at most  $l$  determined by the protocol, and the proportionality symbol  $\propto$  indicates that the state is unnormalized. Measuring the  $x$ -register then produces samples with probability proportional to  $|P(f(x))|^2$ . Because the amplitude depends only on the value  $f(x)$  and not on  $x$  itself, all assignments with the same objective value are sampled with equal probability. This “level-set fairness” is a distinctive feature of DQI.

## C. DQI Analysis

DQI can be viewed as a controlled way to shape amplitudes. Phase marking, interference, and decoding together tilt the final measurement distribution toward higher objective values in a controlled way. The degree budget  $l$ , which bounds the Hamming weight of  $y$ , controls how sharply the biasing polynomial  $P(f)$  can grow. Larger  $l$  allows stronger bias but also requires a decoder strong enough to handle combinations of up to  $l$  constraints.

For a broad family of problems called max-LINSAT, the original paper analyzes this trade-off in the limit of many constraints [2]. Suppose there are  $m$  constraints and let  $s(x)$  denote the number of constraints satisfied by assignment  $x$ . The satisfaction fraction is

$$p = \frac{s(x)}{m}.$$

Under idealized assumptions, the distribution of  $p$  over DQI measurement outcomes takes a semicircle-like shape between a value near  $p_0$  and a higher, biased value. Here  $p_0$  denotes the baseline satisfaction fraction for a uniformly random assignment. The probability density  $\rho(p)$  has the form

$$\rho(p) = \frac{2}{\pi R^2} \sqrt{R^2 - (p - \mu)^2}$$

for  $|p - \mu| \leq R$  and  $\rho(p) = 0$  otherwise, where  $\mu$  is the center and  $R$  is the radius of the semicircle. The parameters  $\mu$  and  $R$  are determined by the normalized degree ratio  $r = l/m$ . As  $r$  increases from 0, the peak of the semicircle moves away from the random baseline toward larger satisfaction fractions and eventually comes back down once decoding becomes difficult.

Case studies such as Optimal Polynomial Interpolation (OPI) and sparse max-XORSAT in [2] illustrate this picture. In OPI, each instance specifies a collection of points together with small sets of allowed function values, and the goal is to find a low-degree polynomial over a finite field that agrees with as many of these allowed values as possible; DQI encodes these consistency constraints as a sparse system of parity checks on the polynomial's coefficients and uses an algebraic decoder to interpret the interference pattern, and in this setting the authors prove a super-polynomial quantum speedup over classical algorithms under a natural black-box model. Sparse max-XORSAT asks for an assignment to binary variables that satisfies as many XOR clauses as possible, where each clause constrains the parity (modulo 2) of a small subset of variables; here DQI is benchmarked against classical heuristics such as simulated annealing and can outperform these generic methods on certain structured instances, while carefully tuned classical solvers exploiting the same structure can still surpass DQI.

## IV. Significance and Potential Impact

Decoded quantum interferometry is interesting not only as a new algorithm but as a different way to use quantum circuits for optimization. Instead of requiring the quantum device to output the optimal bit string directly, DQI treats it as a source of structured samples whose statistics are decoded classically.

### A. Broader Class of Applicable Problems

One conceptual strength of DQI is that it is not tied to a single constraint family. The quantum part only needs to prepare a superposition over candidate assignments, range over small subsets of constraints, and compute a short “syndrome” that summarizes which constraints are touched. The classical part is a decoder chosen to match the algebraic or combinatorial structure of those constraints.

In practice this means that any optimization problem that can be expressed as a sparse set of parity-like checks, or more general low-degree interactions, is a potential target. Examples include parity-based satisfiability such as max-XORSAT, modular linear constraint problems, and Ising or QUBO models with relatively low graph degree, including Max-Cut on sparse graphs. Weighted or filtered versions can usually be handled by simple preprocessing or by reweighting constraints, without changing the basic quantum circuit.

### B. Potential Practical Relevance

On noisy intermediate-scale quantum (NISQ) devices, DQI’s main practical appeal is that it keeps the quantum part shallow and offloads most of the heavy work to classical decoding routines. Unlike adiabatic methods that require long, coherent interpolation schedules, or high-depth QAOA circuits that must be trained in a large parameter space, DQI uses a small number of interference blocks with fixed structure. This reduces exposure to decoherence, control errors, and barren plateaus in parameter optimization.

The output distribution also has a useful structural property. For a given objective value  $f(x)$ , all assignments with that value are sampled with equal probability. This “level-set fairness” makes DQI attractive as a biasing sampler. It can concentrate probability on high-quality assignments while still preserving diversity within each value level, which is useful for approximate counting, benchmarking classical heuristics, and seeding classical local search or refinement. As decoders improve and more problem families are recast into

sparse parity-check form, the same DQI template could therefore gain practical relevance without requiring deeper or more complex quantum circuits.

## V. Limitations

DQI's advantages come with several conditions. Its quantum part must produce stable interference patterns, its classical part must have access to a strong decoder that matches the problem's structure, and the objective must have enough low-degree, Fourier-sparse structure for the semicircle-style analysis to be meaningful. If any of these pieces fail, DQI can lose its edge over more conventional methods.

### A. Hardware Constraints

Even though DQI uses shallow prepare-interfere blocks, it still needs phase-coherent operation over many circuit repetitions (“shots”). Small differences in calibration, crosstalk between qubits, or slow drift in control parameters can distort the interference pattern across shots and make the estimated semicircle distribution unreliable. Resolving small differences between level sets may require many samples, which amplifies the impact of slow drift. In addition, coherent implementations of powerful decoders can require extra ancilla qubits and reversible arithmetic. Current NISQ devices may struggle to provide enough clean qubits and stable gates to run both the interference and decoding stages at the desired scale.

### B. Possibility of Classical Catch-Up

DQI's performance also depends critically on the structure of the objective. If the target problem does not exhibit low-degree, sparse parity structure, the biasing polynomial effectively becomes weak, and DQI offers little improvement over random sampling. Even when such structure exists, classical algorithms can sometimes exploit it just as well. In sparse max-XORSAT, for example, DQI can outperform generic heuristics like off-the-shelf simulated annealing on some structured instances, but carefully engineered classical solvers that use the same parity constraints and graph structure can still surpass DQI. Increasing the degree budget  $l$  strengthens the bias but makes decoding harder. If decoding becomes the bottleneck, any end-to-end quantum advantage shrinks, and there is room for classical methods to catch up or overtake DQI.

## VI. Conclusion

Decoded Quantum Interferometry offers a different way to think about quantum optimization. Rather than trying to prepare an optimal state in a cost landscape, it uses short quantum circuits to imprint information about an objective into interference patterns and then hands those patterns to classical decoders. In problems where the objective has a low-degree, Fourier-sparse structure and a good decoder is available, this approach can tilt the sampling distribution toward high-quality assignments in a mathematically controlled way, as captured by the semicircle analysis for max-LINSAT and the super-polynomial speedup result for OPI.

At the same time, DQI is not a universal solution. Its benefits depend on hardware that can maintain stable interference across many shots and on objective functions that match the parity-check viewpoint. Classical algorithms remain competitive, and in some cases superior, when they exploit the same structure. For near-term devices, DQI is best viewed as a structured biasing sampler that complements classical local search and specialized solvers, rather than replacing them. In that role it highlights an important conceptual shift: quantum circuits can be used not only to search energy landscapes, but also to generate decodable interference patterns that reveal hidden algebraic structure in optimization problems.

## References

- [1] P. W. Shor, Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer, SIAM Journal on Computing, 1997.
- [2] S. P. Jordan, N. Shutty, M. Wootters, A. Zalcman, A. Schmidhuber, R. King, S. V. Isakov, T. Khattar and R. Babbush, Optimization by Decoded Quantum Interferometry, Nature, 2025.
- [3] "Fundamentals of quantum algorithms," IBM Quantum Platform, [Online]. Available: <https://quantum.cloud.ibm.com/learning/en/courses/fundamentals-of-quantum-algorithms/grover-algorithm/introduction>. [Accessed 05 11 2025].
- [4] D-Wave Quantum Inc., "What is Quantum Annealing?," D-Wave Quantum Inc., 17 10 2025. [Online]. Available: <https://support.dwavesys.com/hc/en-us/articles/360003680954-What-Is-Quantum-Annealing>. [Accessed 28 11 2025].
- [5] T. G. Wong, Introduction to Classical and Quantum Computing, Omaha, Nebraska, 2022.
- [6] "Explore quantum Interference," Microsoft, [Online]. Available: <https://quantum.microsoft.com/en-us/insights/education/concepts/interference>. [Accessed 08 11 2025].
- [7] Qiskit Algorithms Development Team, "Quantum Approximate Optimization Algorithm," IBM Quantum, 9 9 2025. [Online]. Available: <https://quantum.cloud.ibm.com/docs/en/tutorials/quantum-approximate-optimization-algorithm>. [Accessed 28 11 2025].